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

Journal:	*IEEE Transactions on Wireless Communications*
Manuscript ID:	Paper-TW-Sep-12-1400
Manuscript Type:	Original Transactions Paper
Date Submitted by the Author:	13-Sep-2012
Complete List of Authors:	El Soussi, Mohieddine; Université catholique de Louvain, ICTEAM Zaidi, Abdellatif; Université Paris-Est Marne la Vallée, Institut Gaspard Monge Vandendorpe, Luc; Université catholique de Louvain, ICTEAM
Keyword:	Compute-and-forward, network coding, relay channel, optimization

SCHOLARONE™
Manuscripts

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

Mohieddine El Soussi   Abdellatif Zaidi   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 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 sum-rate. Next, we address the power allocation and the selection of the integer-valued coefficients to maximize the offered sum-rate; an iterative coordinate descent method is proposed. The analysis shows that our coding schemes can offer substantial gains over standard relaying techniques. The results are illustrated through some numerical examples.

Index Terms

Mohieddine El Soussi is with ICTEAM, Université catholique de Louvain, Place du Levant.2, 1348 Louvain la Neuve, Belgium. Email: mohieddine.elsoussi@uclouvain.be

Abdellatif Zaidi is with Université Paris-Est Marne La Vallée, 77454 Marne la Vallée Cedex 2, France. Email: abdellatif.zaidi@univ-mlv.fr

Luc Vandendorpe is with ICTEAM, Université catholique de Louvain, Place du Levant.2, 1348 Louvain la Neuve, Belgium. Email: luc.vandendorpe@uclouvain.be

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] are 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 interference-relay channel [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 be 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. 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 separate nodes, is less easy to achieve. By opposition to 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. 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 the destination are chosen appropriately, the destination can solve for the transmitted codewords. We consider individual power constraints at the transmitters and the relay, and analyse the sum rate offered by this coding scheme [14] [15].

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 Wyner-Ziv compression [16]. 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. The destination determines the two required linear combinations, as follows. It utilizes an appropriate combination of the output from 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 sum 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 alternatively. More specifically, we show that the problem of finding appropriate integer coefficients for a given set of powers has the same solution as an approximated 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 the 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 both schemes outperform 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, decode-and-forward or compress-and-forward relaying schemes. 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 sum 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 sum-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 $\mathbf{X}$; and, if $m = n$, $\det(\mathbf{X})$ to designate the determinant 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, we use $\text{Var}(\mathbf{x})$ to denote the power of $\mathbf{x}$ i.e. $\mathbb{E}[\|\mathbf{x}\|^2]$. 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 $(\boldsymbol{\lambda}_1 - \boldsymbol{\lambda}_2) \in \Lambda$. For an n-dimensional lattice $\Lambda \in \mathbb{R}^n$, there exists (at least one) matrix $\mathbf{G} \in \mathbb{R}^{n \times n}$ such that any lattice point $\boldsymbol{\lambda} \in \Lambda$ can be written as an integer combination of the columns of $\mathbf{G}$. The matrix $\mathbf{G}$ is called generator matrix of Λ, and satisfies

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

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.,

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

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

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

A lattice Λ is said to be nested into another lattice Λ_{FINE} if $\Lambda \subset \Lambda_{\text{FINE}}$, 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 $\Lambda \subset \Lambda_{\text{FINE}}$, 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

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

The rate of this codebook is

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

Finally, the mod operation satisfies the following properties:

$$(\text{P1}) \qquad [[\mathbf{x}] \bmod \Lambda + \mathbf{y}] \bmod \Lambda = [\mathbf{x} + \mathbf{y}] \bmod \Lambda, \qquad \forall \, \mathbf{x} \in \mathbb{R}^n, \, \mathbf{y} \in \mathbb{R}^n$$

$$(\text{P2}) \qquad [k([\mathbf{x}] \bmod \Lambda)] \bmod \Lambda = [k\mathbf{x}] \bmod \Lambda, \qquad \forall \, k \in \mathbb{Z}, \, \mathbf{x} \in \mathbb{R}^n$$

$$(\text{P3}) \qquad \gamma([\mathbf{x}] \bmod \Lambda) = [\gamma\mathbf{x}] \bmod \gamma\Lambda, \qquad \forall \, \gamma \in \mathbb{R}, \, \mathbf{x} \in \mathbb{R}^n. \tag{9}$$

B. System Model

We consider the communication system shown in Figure 1. Two transmitters A and B communicate with a destination D with the help of a common relay R. Transmitter A, and Transmitter 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 sum rate $R_{\text{sum}} = R_a + R_b = 2R$. 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.
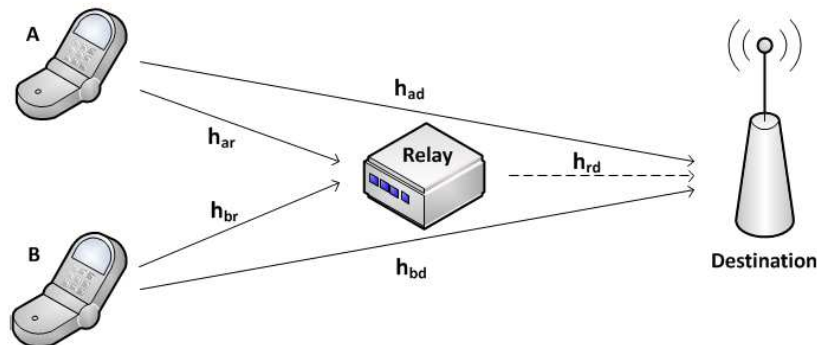

Fig. 1. Multiple-access channel with a half-duplex relay

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 at the 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 the 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, 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 the 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),

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

where $P_a \geq 0$, $P_b \geq 0$ and $P_r \geq 0$ are some constraints imposed by the system; $P \geq 0$ 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 $0 \leq |\beta_a| \leq \sqrt{P_a/P}$, $0 \leq |\beta_b| \leq \sqrt{P_b/P}$ and $0 \leq |\beta_r| \leq \sqrt{P_r/P}$. 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 $\mathbf{H} = [\mathbf{h}_d^T; \mathbf{h}_r^T] \in \mathbb{R}^{2\times 2}$. 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 snr $= P/N$ in the linear scale, and by SNR $= 10\log_{10}(\text{snr})$ in decibels in the logarithmic scale.

C. Sum 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 rather straightforwardly [13] that the following sum rate is achievable with the relay operating on the amplify-and-forward mode,

$$R_{\text{sum}}^{\text{AF}} = \max\ \frac{1}{2}\min\ \bigg\{ \log\Big(\det\Big(\mathbf{I}_2 + \beta_a^2\text{snr}(\mathbf{h}_a\mathbf{h}_a^T)\Big)\Big),$$

$$\log\Big(\det\Big(\mathbf{I}_2 + \beta_b^2\text{snr}(\mathbf{h}_b\mathbf{h}_b^T)\Big)\Big),\ \frac{1}{2}\log\Big(\det\Big(\mathbf{I}_2 + \beta_a^2\text{snr}(\mathbf{h}_a\mathbf{h}_a^T) + \beta_b^2\text{snr}(\mathbf{h}_b\mathbf{h}_b^T)\Big)\Big)\bigg\}. \quad (14)$$

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]

$$R_{\text{sum}}^{\text{DF}} = \max\ \frac{1}{4}\min\ \bigg\{ \log\Big(1 + \text{snr}\,\|\boldsymbol{\beta_s}\circ\mathbf{h}_r\|^2\Big),\ \log\Big(1 + \text{snr}\,\|\boldsymbol{\beta_s}\circ\mathbf{h}_d\|^2\Big) + \log\Big(1 + \text{snr}\,|h_{rd}|^2\beta_r^2\Big)\bigg\}, \quad (15)$$

where the maximization is over $\boldsymbol{\beta}$. Under the constraint of *symmetric-rate*, it can be shown rather straightforwardly [13] that the following sum rate is achievable with the relay operating on the decode-and-forward mode,

$$R_{\text{sum}}^{\text{DF}} = \max\ \frac{1}{2}\min\ \bigg\{ R(\mathbf{h}_r),\ R(\mathbf{h}_d) + \frac{1}{2}\log\Big(1 + \text{snr}|h_{rd}|^2\beta_r^2\Big)\bigg\}, \quad (16)$$

where

$$R(\mathbf{h}_i) = \min\ \bigg\{ \log\Big(1 + \text{snr}|h_{ai}|^2\beta_a^2\Big),\ \log\Big(1 + \text{snr}|h_{bi}|^2\beta_b^2\Big),\ \frac{1}{2}\log\Big(1 + \text{snr}\,\|\boldsymbol{\beta_s}\circ\mathbf{h}_i\|^2\Big)\bigg\}. \quad (17)$$

3) Compress-and-Forward: At the end of the first transmission period, the relay quantizes the received $\mathbf{y}_r$ using Wyner-Ziv compression [14], 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], [20],

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

where

$$R^{\text{CF}} = \max\ \log\left(\frac{\big(1 + \text{snr}\|\boldsymbol{\beta_s}\circ\mathbf{h}_d\|^2\big)\big(1 + D/N + \text{snr}\|\boldsymbol{\beta_s}\circ\mathbf{h}_r\|^2\big) - \text{snr}^2\big((\boldsymbol{\beta_s}\circ\mathbf{h}_r)^T(\boldsymbol{\beta_s}\circ\mathbf{h}_d)\big)^2}{(1 + D/N)}\right), \quad (19)$$

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,

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

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

$$R_{\text{sum}}^{\text{CF}} = \max\ \frac{1}{2}\min\ \bigg\{ \log\Big(1 + \text{snr}|h_{ad}|^2\beta_a^2 + \frac{\text{snr}|h_{ar}|^2\beta_a^2}{1 + D}\Big),\ \log\Big(1 + \text{snr}|h_{bd}|^2\beta_b^2 + \frac{\text{snr}|h_{br}|^2\beta_b^2}{1 + D}\Big),\ \frac{1}{2}R^{\text{CF}}\bigg\}. \quad (21)$$

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 Wyner-Ziv compression [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 sum 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 sum rate is achievable for the multiaccess relay model that we study:

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

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 $|\det(\mathbf{k}, \mathbf{t})| \geq 1$.

As we already indicated, in the coding scheme that we use for the proof 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 obtain two equations that can be solved for the transmitted codewords.

Proof: In what follows, we first describe the transmission scheme, the encoding procedures at the transmitters and the relay, and the decoding procedures at the relay and the destination. Then, we analyze the allowed sum rate.

Let Λ be an n-dimensional lattice of second moment $\sigma_\Lambda^2 = P$ and normalized second moment $G(\Lambda)$. Let $\mathcal{V}$ be its fundamental Voronoi region. Also, let $\Lambda_{\text{FINE}} \supseteq \Lambda$ be chosen such that the codebook $\mathcal{C} = \Lambda_{\text{FINE}} \cap \mathcal{V}$ be of cardinality 2^{2nR} [21]. Also, let $\mathbf{k} = [k_a, k_b] \in \mathbb{Z}^2$ and $\mathbf{t} = [t_a, t_b] \in \mathbb{Z}^2$ be given such that $|\det(\mathbf{k}, \mathbf{t})| = |k_a t_b - k_b t_a| \geq 1$.

The encoding and transmission scheme is as follows.

Encoding: 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

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

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

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

where $0 \le |\beta_b| \le \sqrt{P_b/P}$. The scalars β_a and β_b are chosen so as to adjust the transmitters' powers during this period.

As will be shown shortly the relay decodes correctly an integer combination $\mathbf{e}_2 = k_a\mathbf{v}_a + k_b\mathbf{v}_b$ from what it receives during the first transmission period. It then sends

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

during the second transmission period, where the scalar β_r is chosen so as to adjust its transmitted power during this period.

Decoding: During the first transmission period, the relay receives

$$\mathbf{y}_r \quad = \quad h_{ar}\mathbf{x}_a + h_{br}\mathbf{x}_b + \mathbf{z}_r. \tag{26}$$

Next, the relay performs the following modulo-reduction operation on the received signal:

$$
\begin{aligned}
\mathbf{y}_r' &= \left[\alpha_r\mathbf{y}_r + k_a\mathbf{u}_a + k_b\mathbf{u}_b\right] \mod \Lambda \\
&= \left[\alpha_r(h_{ar}\mathbf{x}_a + h_{br}\mathbf{x}_b + \mathbf{z}_r) + \frac{k_a}{\beta_a}\mathbf{x}_a - \frac{k_a}{\beta_a}\mathbf{x}_a + \frac{k_b}{\beta_b}\mathbf{x}_b - \frac{k_b}{\beta_b}\mathbf{x}_b + k_a\mathbf{u}_a + k_b\mathbf{u}_b\right] \mod \Lambda \\
&\overset{(a)}{=} \left[(\alpha_r h_{ar} - \frac{k_a}{\beta_a})\mathbf{x}_a + (\alpha_r h_{br} - \frac{k_b}{\beta_b})\mathbf{x}_b + k_a[\mathbf{v}_a - \mathbf{u}_a] \mod \Lambda + k_b[\mathbf{v}_b - \mathbf{u}_b] \mod \Lambda + \alpha_r\mathbf{z}_r + k_a\mathbf{u}_a + k_b\mathbf{u}_b\right] \mod \Lambda \\
&\overset{(b)}{=} \left[k_a\mathbf{v}_a + k_b\mathbf{v}_b + (\alpha_r h_{ar} - \frac{k_a}{\beta_a})\mathbf{x}_a + (\alpha_r h_{br} - \frac{k_b}{\beta_b})\mathbf{x}_b + \alpha_r\mathbf{z}_r\right] \mod \Lambda \\
&\overset{(c)}{=} \left[k_a\mathbf{v}_a + k_b\mathbf{v}_b + \mathbf{z}_r'\right] \mod \Lambda
\end{aligned}
\tag{27}
$$

where (a) follows by substituting $\mathbf{x}_a$ and $\mathbf{x}_b$ using (23) and (24); (b) follows by using the properties (P1) and (P2) in (9); and (c) follows by substituting,

$$\mathbf{z}_r' \triangleq \left[\alpha_r\mathbf{z}_r + (\alpha_r h_{ar} - \frac{k_a}{\beta_a})\mathbf{x}_a + (\alpha_r h_{br} - \frac{k_b}{\beta_b})\mathbf{x}_b\right] \mod \Lambda. \tag{28}$$

The parameter $\alpha_r \in \mathbb{R}$ is some inflation factor whose optimal value will be specified below and $\mathbf{z}_r'$ is the *effective* noise at the relay. Since an integer combination of lattice points is a lattice point, $\mathbf{e}_2 = [k_a\mathbf{v}_a + k_b\mathbf{v}_b] \in \Lambda$ and, thus, the equivalent channel $\mathbf{y}_r'$ is a modulo-lattice additive noise (MLAN) channel [22] with noise equal to $\mathbf{z}_r'$. By decoding the lattice point $\mathbf{e}_2 \in \Lambda$ from the MLAN channel $\mathbf{y}_r'$, the relay obtains an integer combination of the users' codewords, $\mathbf{e}_2 = [k_a\mathbf{v}_a + k_b\mathbf{v}_b]$.

During the first transmission period, the destination receives

$$\mathbf{y}_d = h_{ad}\mathbf{x}_a + h_{bd}\mathbf{x}_b + \mathbf{z}_d. \tag{29}$$

Similar to the relay, the destination computes a linear combination with integer coefficients of the transmitters' codewords by performing the modulo-reduction operation:

$$
\begin{aligned}
\mathbf{y}_d' &= \left[\alpha_d\mathbf{y}_d + t_a\mathbf{u}_a + t_b\mathbf{u}_b\right] \mod \Lambda \\
&= \left[t_a\mathbf{v}_a + t_b\mathbf{v}_b + \mathbf{z}_d'\right] \mod \Lambda
\end{aligned}
\tag{30}
$$

where $\alpha_d \in \mathbb{R}$ is some inflation factor and $\mathbf{z}_d'$ is the effective noise at the destination, given by

$$\mathbf{z}_d' \triangleq \left[\alpha_d\mathbf{z}_d + (\alpha_d h_{ad} - \frac{t_a}{\beta_a})\mathbf{x}_a + (\alpha_d h_{bd} - \frac{t_b}{\beta_b})\mathbf{x}_b\right] \mod \Lambda. \tag{31}$$

Thus, during the first transmission period, by using the MLAN channel $\mathbf{y}_d'$, the destination obtains a first integer combination $\mathbf{e}_1 = [t_a\mathbf{v}_a + t_b\mathbf{v}_b] \in \Lambda$ of the users' codewords.

During the second transmission period, the destination receives from the relay the signal,

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

$$= h_{rd}\beta_r \left(\left[k_a\mathbf{v}_a + k_b\mathbf{v}_b - \mathbf{u}_r \right] \bmod \Lambda \right) + \tilde{\mathbf{z}}_d. \tag{32}$$

Again, by performing a modulo-reduction operation on the obtained signal, the destination gets

$$\tilde{\mathbf{y}}'_d = \left[\tilde{\alpha}_d\tilde{\mathbf{y}}_d + \mathbf{u}_r \right] \bmod \Lambda$$

$$= \left[k_a\mathbf{v}_a + k_b\mathbf{v}_b + \tilde{\mathbf{z}}'_d \right] \bmod \Lambda, \tag{33}$$

where $\tilde{\alpha}_d \in \mathbb{R}$ is some inflation factor whose optimal value will be specified below and $\tilde{\mathbf{z}}'_d$ is the effective noise at the destination, given by

$$\tilde{\mathbf{z}}'_d \triangleq \left[\tilde{\alpha}_d\tilde{\mathbf{z}}_d + (\tilde{\alpha}_d h_{rd} - \frac{1}{\beta_r})\tilde{\mathbf{x}}_r \right] \bmod \Lambda. \tag{34}$$

Thus, during the second transmission period, the destination obtains a second linear combination $\mathbf{e}_2 = \left[k_a\mathbf{v}_a + t_b\mathbf{v}_b \right] \in \Lambda$ that relates the users' codewords. This combination has been computed and forwarded by the relay.

Summarizing, over the entire transmission time, the destination collects two linear combinations with integer coefficients that relate the users' codewords, as

$$\begin{bmatrix} \mathbf{e}_1 \\ \mathbf{e}_2 \end{bmatrix} = \begin{bmatrix} t_a & t_b \\ k_a & k_b \end{bmatrix} \begin{bmatrix} \mathbf{v}_a \\ \mathbf{v}_b \end{bmatrix}. \tag{35}$$

Now, since the integer-valued matrix in (35) 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 (35).

The destination is able to recover the messages $\hat{W}_a$ and $\hat{W}_b$ reliably only if the message rate is less than or equal to the computational rate [13].

Rate Analysis:

<u>Step 1)</u> First, we find the rate at which the relay decodes reliably the combination $\mathbf{e}_2 = \left[k_a\mathbf{v}_a + t_b\mathbf{v}_b \right] \in \Lambda$. Consider the channel from $\mathbf{e}_2$ to $\mathbf{y}'_r$ as given by (27). Due to the dither $\mathbf{u}_a$, the input $\mathbf{x}_a$ of Transmitter A is independent of $\mathbf{v}_a = \phi_a(W_a) \in \mathcal{C}$, and is uniformly distributed over $\mathcal{V}$ (see, e.g., [23], [24]). Similarly, due to the dither $\mathbf{u}_b$, the input $\mathbf{x}_b$ of Transmitter B is independent of $\mathbf{v}_b = \phi_b(W_b) \in \mathcal{C}$, and is uniformly distributed over $\mathcal{V}$. The effective noise $\mathbf{z}'_r$ is then independent of $\mathbf{v}_a$ and $\mathbf{v}_b$, and so of $\mathbf{e}_2 = k_a\mathbf{v}_a + k_b\mathbf{v}_b$. It is composed of a "self noise" component and a Gaussian noise component; and its average power satisfies

$$\mathrm{Var}\left(\mathbf{z}'_r\right) \leq \mathrm{Var}\left(\alpha_r\mathbf{z}_r + (\alpha_r h_{ar} - \frac{k_a}{\beta_a})\mathbf{x}_a + (\alpha_r h_{br} - \frac{k_b}{\beta_b})\mathbf{x}_b \right)$$

$$= \alpha_r^2 N + P(\alpha_r h_{ar}\beta_a - k_a)^2 + P(\alpha_r h_{br}\beta_b - k_b)^2. \tag{36}$$

where the inequality follows since the modulo-reduction operation can only reduce the entropy.

The rate (per channel use) at which the relay decodes the linear combination with integer coefficients $\mathbf{e}_2 = k_a \mathbf{v}_a + k_b \mathbf{v}_b$ is given by [25],

$$
\begin{aligned}
\frac{1}{2n} I(\mathbf{e}_2; \mathbf{y}_r') &= \frac{1}{2n} h(\mathbf{y}_r') - \frac{1}{2n} h(\mathbf{y}_r' | \mathbf{e}_2) \\
&\overset{(d)}{=} \frac{1}{2n} h(\mathbf{y}_r') - \frac{1}{2n} h(\mathbf{z}_r') \\
&\overset{(e)}{=} \frac{1}{2n} \log^+ \mathrm{Vol}\,(\mathcal{V}) - \frac{1}{2n} h(\mathbf{z}_r') \\
&\overset{(f)}{\geq} \frac{1}{4} \log^+ \left(\frac{P}{\alpha_r^2 N + P(\alpha_r h_{ar}\beta_a - k_a)^2 + P(\alpha_r h_{br}\beta_b - k_b)^2} \right) - \frac{1}{4} \log\left(2\pi e G(\Lambda)\right) \\
&\overset{(g)}{\geq} \frac{1}{4} \log^+ \left(k_a^2 + k_b^2 - \frac{P(h_{ar}\beta_a k_a + h_{br}\beta_b k_b)^2}{N + P(|h_{ar}|^2 \beta_a^2 + |h_{br}|^2 \beta_b^2)} \right)^{-1} - \frac{1}{4} \log\left(2\pi e G(\Lambda)\right) \\
&= \frac{1}{4} \log^+ \left(\left(\|\mathbf{k}\|^2 - \frac{P((\boldsymbol{\beta}_s \circ \mathbf{h}_r)^T \mathbf{k})^2}{N + P\|\boldsymbol{\beta}_s \circ \mathbf{h}_r\|^2} \right)^{-1} \right) - \frac{1}{4} \log\left(2\pi e G(\Lambda)\right)
\end{aligned}
\tag{37}
$$

where (d) follows from the fact that $\mathbf{z}_r'$ is independent of $\mathbf{e}_2$; (e) follows from the fact that $\mathbf{y}_r'$ is uniformly distributed over $\mathcal{V}$; (f) follows from the fact that the entropy of $\mathbf{z}_r'$ is upper-bounded by that of an i.i.d Gaussian vector that has the same second moments and by using (36); and (g) follows through straightforward algebra and by noticing that the right hand side (RHS) of (36) is maximized with the value of α_r set to

$$
\alpha_r^\star = \frac{P(\boldsymbol{\beta}_s \circ \mathbf{h}_r)^T \mathbf{k}}{N + P\|\boldsymbol{\beta}_s \circ \mathbf{h}_r\|^2}.
\tag{38}
$$

Step 2) Using analysis and algebra that are essentially similar to those in Step 1) above, it can be shown that the destination decodes the linear combination $\mathbf{e}_1 = t_a \mathbf{v}_a + t_b \mathbf{v}_b$ from what it gets directly from the direct links at rate

$$
\frac{1}{2n} I(\mathbf{e}_1; \mathbf{y}_d') \geq \frac{1}{4} \log^+ \left(\left(\|\mathbf{t}\|^2 - \frac{P((\boldsymbol{\beta}_s \circ \mathbf{h}_r)^T \mathbf{t})^2}{N + P\|\boldsymbol{\beta}_s \circ \mathbf{h}_r\|^2} \right)^{-1} \right) - \frac{1}{4} \log\left(2\pi e G(\Lambda)\right).
\tag{39}
$$

Similarly, it can be easily shown that the destination decodes the linear combination that is computed at the relay and forwarded to the destination during the second transmission period at rate

$$
\frac{1}{2n} I(\mathbf{e}_2; \tilde{\mathbf{y}}_d') \geq \frac{1}{4} \log \left(1 + \frac{P|h_{rd}|^2 \beta_r^2}{N} \right) - \frac{1}{4} \log\left(2\pi e G(\Lambda)\right).
\tag{40}
$$

Let us define $R_{sr}(\Lambda)$ as the RHS of (37), $R_{sd}(\Lambda)$ as the RHS of (39), and $R_{rd}(\Lambda)$ as the RHS of (40).

The above means that using the coding scheme that we described, which employs the n-dimensional lattice Λ, the destination can decode the transmitters' codewords correctly at a transmission rate equal to the minimum among $R_{sr}(\Lambda)$, $R_{sd}(\Lambda)$ and $R_{rd}(\Lambda)$, i.e., $R(\Lambda) = \min\{R_{sr}(\Lambda), R_{sd}(\Lambda), R_{rd}(\Lambda)\}$. The allowed sum rate is given by $R_{\mathrm{sum}}^{\mathrm{CoF}}(\Lambda) = 2R(\Lambda) = 2\min\{R_{sr}(\Lambda), R_{sd}(\Lambda), R_{rd}(\Lambda)\}$. Noticing that $2\pi e G(\Lambda) \to 1$ when $n \to \infty$ [10], the desired sum rate (22) is obtained by taking the limit of $R_{\mathrm{sum}}^{\mathrm{CoF}}(\Lambda)$ as n goes to infinity; and this completes the proof of Proposition 1.   □

Although the achievable sum 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 sum 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 sum rate is achievable:

$$R_{\text{sum}}^{\text{CoD}} = \max \frac{1}{2} \min \left\{ \log^+ \left(\frac{\text{snr}}{\text{snr}\|\boldsymbol{\beta}_s \circ \mathbf{H}^T \boldsymbol{\alpha}_t - \mathbf{t}\|^2 + (\boldsymbol{\alpha}_t \circ \boldsymbol{\alpha}_t)^T \mathbf{n}_d} \right), \right.$$

$$\left. \log^+ \left(\frac{\text{snr}}{\text{snr}\|\boldsymbol{\beta}_s \circ \mathbf{H}^T \boldsymbol{\alpha}_k - \mathbf{k}\|^2 + (\boldsymbol{\alpha}_k \circ \boldsymbol{\alpha}_k)^T \mathbf{n}_d} \right) \right\}, \tag{41}$$

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, $\mathbf{n}_d = [1, \, 1 + D/N]^T \in \mathbb{R}^2$, and D is given by

$$D = \frac{N^2 \left(1 + \text{snr}\|\boldsymbol{\beta}_s \circ \mathbf{h}_r\|^2 \right)}{|h_{rd}|^2 P_r} - \frac{N^2 \left(\text{snr}(\boldsymbol{\beta}_s \circ \mathbf{h}_r)^T (\boldsymbol{\beta}_s \circ \mathbf{h}_d) \right)^2}{|h_{rd}|^2 P_r \left(1 + \text{snr}\|\boldsymbol{\beta}_s \circ \mathbf{h}_d\|^2 \right)}, \tag{42}$$

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 $|\det(\mathbf{k}, \mathbf{t})| \geq 1$.

As we indicated previously, 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 relates the users' codewords.

Proof: The transmission scheme, and the encoding procedures at the transmitters are similar to those in the proof of Proposition 1. Therefore they will be outlined only, for brevity. We will insist more on aspects of the coding scheme that are inherently different from those of the coding scheme of Proposition 1.

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).

The relay quantizes what it receives during the first transmission period using Wyner-Ziv compression [16], accounting for the available side information $\mathbf{y}_d$ at the destination. Let $\hat{\mathbf{y}}_r$ be the compressed version of $\mathbf{y}_r$ given by

$$\hat{\mathbf{y}}_r = \mathbf{y}_r + \mathbf{d} \tag{43}$$

where $\mathbf{d}$ is a Gaussian random vector whose elements are i.i.d with zero mean and variance D; and is independent of all other signals. Also let $\hat{R}_{\text{WZ}}$ be the resulting compression rate. During the second transmission period, the relay conveys the description $\hat{\mathbf{y}}_r$ of $\mathbf{y}_r$ to the destination. To this end, it sends an independent Gaussian input $\tilde{\mathbf{x}}_r$ with $\beta_r^2 P$ and carries the Wyner-Ziv compression index of $\hat{\mathbf{y}}_r$.

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

$$\mathbf{y}_d = h_{ad}\mathbf{x}_a + h_{bd}\mathbf{x}_b + \mathbf{z}_d$$

$$\tilde{\mathbf{y}}_d = h_{rd}\tilde{\mathbf{x}}_r + \tilde{\mathbf{z}}_d. \tag{44}$$

The destination computes two linearly independent combinations with integer coefficients that relate the users' codewords, as follows.

Step 1) 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 $\hat{\mathbf{y}}_r$ of $\mathbf{y}_r$ if the constraint (54) below is satisfied (see the "Rate Analysis" section).

Next, the destination combines $\mathbf{y}_d$ and $\hat{\mathbf{y}}_r$ and uses the obtained signal to compute a linear combination with integer coefficients of the users' codewords. More specifically, let

$$
\begin{aligned}
\mathbf{y}_t &= \alpha_{1t}\mathbf{y}_d + \alpha_{2t}\hat{\mathbf{y}}_r \\
&= \left(\alpha_{1t}h_{ad} + \alpha_{2t}h_{ar}\right)\mathbf{x}_a + \left(\alpha_{1t}h_{bd} + \alpha_{2t}h_{br}\right)\mathbf{x}_b + \alpha_{1t}\mathbf{z}_d + \alpha_{2t}\mathbf{z}_r + \alpha_{2t}\mathbf{d},
\end{aligned}
\tag{45}
$$

for some $\boldsymbol{\alpha}_t = \left[\alpha_{1t}, \alpha_{2t}\right]^T \in \mathbb{R}^2$. The destination uses the obtained signal $\mathbf{y}_t$ to compute a linear combination with integer coefficients of the transmitters' codewords by performing the modulo reduction operation

$$
\begin{aligned}
\mathbf{y}'_t &= \left[\mathbf{y}_t + t_a\mathbf{u}_a + t_b\mathbf{u}_b\right] \bmod \Lambda \\
&= \left[t_a\mathbf{v}_a + t_b\mathbf{v}_b + \mathbf{z}'_t\right] \bmod \Lambda
\end{aligned}
\tag{46}
$$

where the algebra follows (27) and $\mathbf{z}'_t$ is the effective noise given by

$$
\mathbf{z}'_t \triangleq \left[\alpha_{1t}\mathbf{z}_d + \alpha_{2t}\mathbf{z}_r + \alpha_{2t}\mathbf{d} + (\alpha_{1t}h_{ad} + \alpha_{2t}h_{ar} - \frac{t_a}{\beta_a})\mathbf{x}_a + (\alpha_{1t}h_{bd} + \alpha_{2t}h_{br} - \frac{t_b}{\beta_b})\mathbf{x}_b\right] \bmod \Lambda.
\tag{47}
$$

Finally, by decoding the lattice point $\mathbf{e}_1 = \left[t_a\mathbf{v}_a + t_b\mathbf{v}_b\right] \in \Lambda$ using the MLAN channel $\mathbf{y}'_t$, the destination obtains a first linear combination with integer coefficients of the users' codewords.

Step 2 Second, the destination again combines $\mathbf{y}_d$ and $\hat{\mathbf{y}}_r$ and uses the obtained signal to compute a second linear combination with integer coefficients of the users' codewords which is different from the one decoded in *Step 1*. More specifically, let

$$
\begin{aligned}
\mathbf{y}_k &= \alpha_{1k}\mathbf{y}_d + \alpha_{2k}\hat{\mathbf{y}}_r \\
&= \left(\alpha_{1k}h_{ad} + \alpha_{2k}h_{ar}\right)\mathbf{x}_a + \left(\alpha_{1k}h_{bd} + \alpha_{2k}h_{br}\right)\mathbf{x}_b + \alpha_{1k}\mathbf{z}_d + \alpha_{2k}\mathbf{z}_r + \alpha_{2k}\mathbf{d},
\end{aligned}
\tag{48}
$$

for some $\boldsymbol{\alpha}_k = \left[\alpha_{1k}, \alpha_{2k}\right]^T \in \mathbb{R}^2$. The destination uses the obtained signal $\mathbf{y}_k$ to compute a linear combination with integer coefficients of the transmitters' codewords by performing the modulo reduction operation

$$
\begin{aligned}
\mathbf{y}'_k &= \left[\mathbf{y}_k + k_a\mathbf{u}_a + k_b\mathbf{u}_b\right] \bmod \Lambda \\
&= \left[k_a\mathbf{v}_a + k_b\mathbf{v}_b + \mathbf{z}'_k\right] \bmod \Lambda
\end{aligned}
\tag{49}
$$

where the algebra follows (27) and $\mathbf{z}'_k$ is the effective noise given by

$$
\mathbf{z}'_k \triangleq \left[\alpha_{1k}\mathbf{z}_d + \alpha_{2k}\mathbf{z}_r + \alpha_{2k}\mathbf{d} + (\alpha_{1k}h_{ad} + \alpha_{2k}h_{ar} - \frac{k_a}{\beta_a})\mathbf{x}_a + (\alpha_{1k}h_{bd} + \alpha_{2k}h_{br} - \frac{k_b}{\beta_b})\mathbf{x}_b\right] \bmod \Lambda.
\tag{50}
$$

Finally, by decoding the lattice point $\mathbf{e}_2 = \left[k_a\mathbf{v}_a + k_b\mathbf{v}_b\right] \in \Lambda$ using the MLAN channel $\mathbf{y}'_k$, the destination obtains a second linear combination with integer coefficients of the users' codewords.

Rate Analysis:

The relay compresses its output $\mathbf{y}_r$ at the per-channel use Wyner-Ziv quantization rate

$$
\begin{aligned}
\hat{R}_{\text{WZ}} &= \frac{1}{2n} I(\mathbf{y}_r; \hat{\mathbf{y}}_r | \mathbf{y}_d) \\
&= \frac{1}{2n} h(\hat{\mathbf{y}}_r | \mathbf{y}_d) - \frac{1}{2n} h(\hat{\mathbf{y}}_r | \mathbf{y}_r, \mathbf{y}_d) \\
&\overset{(a)}{=} \frac{1}{2n} h(\hat{\mathbf{y}}_r | \mathbf{y}_d) - \frac{1}{2n} h(\mathbf{d}) \\
&\overset{(b)}{=} \frac{1}{4n} \log(2\pi e)^n \left| \mathbb{E}[\hat{\mathbf{y}}_r \hat{\mathbf{y}}_r^T] - \mathbb{E}[\hat{\mathbf{y}}_r \mathbb{E}[\hat{\mathbf{y}}_r | \mathbf{y}_d]^T] \right| - \frac{1}{4} \log(2\pi e D) \\
&\overset{(c)}{=} \frac{1}{4} \log\left(1 + \frac{N + P\|\boldsymbol{\beta}_s \circ \mathbf{h}_r\|^2}{D} - \frac{[P(\boldsymbol{\beta}_s \circ \mathbf{h}_r)^T(\boldsymbol{\beta}_s \circ \mathbf{h}_d)]^2}{D(N + P\|\boldsymbol{\beta}_s \circ \mathbf{h}_d\|^2)}\right),
\end{aligned}
\tag{51}
$$

where (a) follows since $\mathbf{d}$ is independent of $(\mathbf{y}_r, \mathbf{y}_d)$; (b) follows since $(\hat{\mathbf{y}}_r, \mathbf{y}_d)$ are jointly Gaussian and $\mathbf{d}$ is a Gaussian vector; (c) follows through by straightforward algebra, and by noticing that the minimum mean square error (MMSE) estimator of $\hat{\mathbf{y}}_r$ given $\mathbf{y}_d$ is given by

$$
\mathbb{E}[\hat{\mathbf{y}}_r | \mathbf{y}_d] = \frac{P(\boldsymbol{\beta}_s \circ \mathbf{h}_r)^T(\boldsymbol{\beta}_s \circ \mathbf{h}_d)}{N + P\|\boldsymbol{\beta}_s \circ \mathbf{h}_d\|^2} \mathbf{y}_d.
\tag{52}
$$

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

$$
\hat{R}_{\text{WZ}} \leq \frac{1}{4} \log\left(1 + \frac{P|h_{rd}|^2 \beta_r^2}{N}\right).
\tag{53}
$$

From (51) and (53), we get the following constraint on the distortion

$$
D \geq \frac{N^2\left(1 + \text{snr}\|\boldsymbol{\beta_s} \circ \mathbf{h}_r\|^2\right)}{|h_{rd}|^2 P_r} \frac{N^2\left(\text{snr}(\boldsymbol{\beta_s} \circ \mathbf{h}_r)^T(\boldsymbol{\beta_s} \circ \mathbf{h}_d)\right)^2}{|h_{rd}|^2 P_r\left(1 + \text{snr}\|\boldsymbol{\beta_s} \circ \mathbf{h}_d\|^2\right)}.
\tag{54}
$$

The above implies that, under the constraint (54), the destination recovers the lossy version $\hat{\mathbf{y}}_r$ of what it was sent by the relay during the second transmission period.

The destination processes $\mathbf{y}_t$ and $\mathbf{y}_k$ to obtain the linear combinations $\mathbf{e}_1 = [t_a\mathbf{v}_a + t_b\mathbf{v}_b]$, and $\mathbf{e}_2 = [k_a\mathbf{v}_a + k_b\mathbf{v}_b]$ of the users' codewords. Using the MLAN channel $\mathbf{y}'_t$ given by (46) and proceeding similar to above, we get that the destination obtains the desired combination $\mathbf{t}$ at mutual information satisfying

$$
\frac{1}{2n} I(\mathbf{e}_1; \mathbf{y}'_t) \geq \frac{1}{4} \log^+\left(\frac{\text{snr}}{\text{snr}\|\boldsymbol{\beta}_s \circ \mathbf{H}^T \boldsymbol{\alpha}_t - \mathbf{t}\|^2 + (\boldsymbol{\alpha}_t \circ \boldsymbol{\alpha}_t)^T \mathbf{n}_d}\right) - \frac{1}{4} \log\left(2\pi e G(\Lambda)\right),
\tag{55}
$$

where the distortion D satisfies the constraint (54) and $\boldsymbol{\alpha}_t$ should be chosen to minimize the effective noise $\mathbf{z}'_t$ in (47), i.e., such that

$$
\boldsymbol{\alpha}_t^\star = \left(\mathbf{G}\mathbf{G}^T + \mathbf{N}_d\right)^{-1} \mathbf{G}\mathbf{t},
\tag{56}
$$

where $\mathbf{G} = [(\boldsymbol{\beta}_s \circ \mathbf{h}_d)^T; (\boldsymbol{\beta}_s \circ \mathbf{h}_r)^T] \in \mathbb{R}^{2 \times 2}$ and $\mathbf{N}_d = [1/\text{snr}, 0; 0, 1/\text{snr} + D/P] \in \mathbb{R}^{2 \times 2}$. In a similar way, we get that the destination obtains the desired combination $\mathbf{k}$ at mutual information satisfying

$$
\frac{1}{2n} I(\mathbf{e}_2; \mathbf{y}'_k) \geq \frac{1}{4} \log^+\left(\frac{\text{snr}}{\text{snr}\|\boldsymbol{\beta}_s \circ \mathbf{H}^T \boldsymbol{\alpha}_k - \mathbf{k}\|^2 + (\boldsymbol{\alpha}_k \circ \boldsymbol{\alpha}_k)^T \mathbf{n}_d}\right) - \frac{1}{4} \log\left(2\pi e G(\Lambda)\right),
\tag{57}
$$

where the distortion D satisfies the constraint (54), and $\boldsymbol{\alpha}_k$ should be chosen to minimize the effective noise $\mathbf{z}'_k$ in (50), i.e., such that

$$
\boldsymbol{\alpha}_k^\star = \left(\mathbf{G}\mathbf{G}^T + \mathbf{N}_d\right)^{-1} \mathbf{G}\mathbf{k}.
\tag{58}
$$

Let us define $R_1(\Lambda)$ as the RHS of (55) and $R_2(\Lambda)$ as the RHS of (57). The above means that using the lattice-based coding scheme that we described, the destination can decode the transmitters' codewords correctly at the transmission sum rate $R_{\text{sum}}^{\text{CoD}}(\Lambda) = 2\min\{R_1(\Lambda), R_2(\Lambda)\}$ provided that the condition (54) is satisfied. Furthermore, investigating the expression of $R_1(\Lambda)$, it can easily be seen that it decreases with increasing D. Also, observing that the RHS of (54) decreases if β_r increases, the largest rate $R_1(\Lambda)$ is then obtained by taking the equality in the distortion constraint (54) with $\beta_r^2 = P_r/P$. Finally, observing that $2\pi eG(\Lambda) \to 1$ when $n \to \infty$ [10], the desired sum rate (41) is obtained by taking the limit of $R_{\text{sum}}^{\text{CoD}}(\Lambda)$ as n goes to infinity; and this completes the proof of Proposition 2. □

Remark 1: There are some high level similarities among the coding strategies of proposition 1 and proposition 2. In particular, they both consist essentially in decoding two linearly independent equations. However, as we mentioned previously, 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. A direct consequence of all the processing being performed locally at the destination with the latter coding scheme is that both computations of the required equations utilize *all* the output available at the destination, i.e., the output received during the first transmission period as well as the output received during the second transmission period, in a joint manner. Recall that, 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, and 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 joint processing at the destination somehow gives some advantage to the coding strategy of proposition 2. The reader may refer to Section V where this aspect will be illustrated through some numerical examples and discussed further.

IV. SUM RATES OPTIMIZATION

Section IV-A is devoted to find optimal powers and integer-coefficients that maximize the sum 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 sum rate $R_{\text{sum}}^{\text{CoF}}$ as given by (22) in Proposition 1. The optimization problem can be stated as:

$$\text{(A)} : \quad \max \frac{1}{2}\min\left\{\log^+\left(\left(\|\mathbf{t}\|^2 - \frac{P((\boldsymbol{\beta}_s\circ\mathbf{h}_d)^T\mathbf{t})^2}{N+P\|\boldsymbol{\beta}_s\circ\mathbf{h}_d\|^2}\right)^{-1}\right), \ \log^+\left(\left(\|\mathbf{k}\|^2 - \frac{P((\boldsymbol{\beta}_s\circ\mathbf{h}_r)^T\mathbf{k})^2}{N+P\|\boldsymbol{\beta}_s\circ\mathbf{h}_r\|^2}\right)^{-1}\right), \ \log\left(1 + \frac{P|h_{rd}|^2\beta_r^2}{N}\right)\right\} \quad (59)$$

where the maximization is over $\boldsymbol{\beta}$ such that $0 \le |\beta_a| \le \sqrt{P_a/P}$, $0 \le |\beta_b| \le \sqrt{P_b/P}$, $0 \le |\beta_r| \le \sqrt{P_r/P}$ and over the integer coefficients $\mathbf{k}$ and $\mathbf{t}$ such that $|\det(\mathbf{k}, \mathbf{t})| \ge 1$.

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 $\boldsymbol{\beta}$ and integer coefficients $\mathbf{t}$ and $\mathbf{k}$ alternatively. We note that the allocation of the vector $\boldsymbol{\beta}$ determines the power that each of the transmitters and the relay should use for the transmission. For this reason, we will sometimes refer loosely to the process of selecting the vector $\boldsymbol{\beta}$ appropriately as the power allocation process.

Let, with a slight abuse of notation, denote by $R_{\text{sum}}^{\text{CoF}}[\iota]$ the value of the sum rate at some iteration $\iota \geq 0$. To compute $R_{\text{sum}}^{\text{CoF}}$ as given by (59) 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 $R_{\text{sum}}^{\text{CoF}}$ as given by (59)

1: Initialization: set $\iota = 1$ and $\boldsymbol{\beta} = \boldsymbol{\beta}^{(0)}$.

2: Set $\boldsymbol{\beta} = \boldsymbol{\beta}^{(\iota-1)}$ in (59), 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 (59), and solve the obtained problem using Algorithm A-2 given below. Denote by $\boldsymbol{\beta}^{(\iota)}$ the found $\boldsymbol{\beta}$.

4: Increment the iteration index as $\iota = \iota + 1$, and go back to Step 2.

5: Terminate if $\|\boldsymbol{\beta}^{(\iota)} - \boldsymbol{\beta}^{(\iota-1)}\| \leq \epsilon$, $|R_{\text{sum}}^{\text{CoF}}[\iota] - R_{\text{sum}}^{\text{CoF}}[\iota-1]| \leq \epsilon$ or $R_{\text{sum}}^{\text{CoF}}[\iota] \leq R_{\text{sum}}^{\text{CoF}}[\iota-1]$.

As described in Algorithm A, we compute the appropriate preprocessing vector $\boldsymbol{\beta}$ and integer coefficients $\mathbf{k}$ and $\mathbf{t}$, alternatively. More specifically, at iteration $\iota \geq 1$, 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 (59) computed with the choice of the preprocessing vector $\boldsymbol{\beta}$ set to its value obtained from the previous iteration, i.e., $\boldsymbol{\beta} = \boldsymbol{\beta}^{(\iota-1)}$ (for the initialization, set $\boldsymbol{\beta}^{(0)}$ to a default value). As we will show, this sub-problem is a MIQP problem with quadratic constraints; and we solve it iteratively using Algorithm A-1. Next, for the found integer coefficients, the algorithm computes adequate preprocessing vector $\boldsymbol{\beta}^{(\iota)}$ that corresponds to a maximum of (59) 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 geometric programming and successive convex optimization approach (see Algorithm A-2 below). The iterative process in Algorithm A terminates if one of the following two conditions holds: 1) either of $\|\boldsymbol{\beta}^{(\iota)} - \boldsymbol{\beta}^{(\iota-1)}\|$ and $|R_{\text{sum}}^{\text{CoF}}[\iota] - R_{\text{sum}}^{\text{CoF}}[\iota-1]|$ is smaller than a prescribed small strictly positive constant ϵ — in this case, the optimized value of the sum rate is $R_{\text{sum}}^{\text{CoF}}[\iota]$, and is attained using the preprocessing power vector $\boldsymbol{\beta}^{\star} = \boldsymbol{\beta}^{(\iota)}$ and integer vectors $\mathbf{k}^{\star} = \mathbf{k}^{(\iota)}$ and $\mathbf{t}^{\star} = \mathbf{t}^{(\iota)}$; or 2) the sum rate stops increasing, i.e., $R_{\text{sum}}^{\text{CoF}}[\iota] \leq R_{\text{sum}}^{\text{CoF}}[\iota-1]$ — in this case, the optimized value of the sum rate is $R_{\text{sum}}^{\text{CoF}}[\iota-1]$, and is attained using the preprocessing power vector $\boldsymbol{\beta}^{\star} = \boldsymbol{\beta}^{(\iota-1)}$ and integer vectors $\mathbf{k}^{\star} = \mathbf{k}^{(\iota-1)}$ and $\mathbf{t}^{\star} = \mathbf{t}^{(\iota-1)}$.

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 $\boldsymbol{\beta}$. Investigating the objective function in (59), it can

easily be seen that this problem can be equivalently stated as

$$\min_{\mathbf{k},\,\mathbf{t},\,\Delta_1} \quad \Delta_1 \tag{60a}$$

$$\text{s. t.} \quad \Delta_1 \ge \|\mathbf{t}\|^2 - \frac{P((\boldsymbol{\beta}_s \circ \mathbf{h}_d)^T \mathbf{t})^2}{N + P\|\boldsymbol{\beta}_s \circ \mathbf{h}_d\|^2} \tag{60b}$$

$$\Delta_1 \ge \|\mathbf{k}\|^2 - \frac{P((\boldsymbol{\beta}_s \circ \mathbf{h}_r)^T \mathbf{k})^2}{N + P\|\boldsymbol{\beta}_s \circ \mathbf{h}_r\|^2} \tag{60c}$$

$$\Delta_1 \ge \frac{N}{N + \beta_r^2 |h_{rd}|^2 P}, \tag{60d}$$

$$|\det(\mathbf{k}, \mathbf{t})| \ge 1 \tag{60e}$$

$$\mathbf{k} \in \mathbb{Z}^2, \ \mathbf{t} \in \mathbb{Z}^2, \ \Delta_1 \in \mathbb{R}. \tag{60f}$$

Note that Δ_1 is simultaneously an extra optimization variable and the objective function in (60). 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 (59).

In order to reformulate problem (60) in a manner that will be convenient for solving it, we 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 $c_1 = c_2 = 0$, $c_3 = N/(N + \beta_r^2 |h_{rd}|^2 P)$ 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

$$\mathbf{F}_1 = \begin{bmatrix} 2(\mathbf{I}_2 - \boldsymbol{\Omega}_1) & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}, \quad \mathbf{F}_2 = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & 2(\mathbf{I}_2 - \boldsymbol{\Omega}_2) & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}, \tag{61}$$

with

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

$$\boldsymbol{\Omega}_2 := \frac{P}{N + P\|\mathbf{h}_r\|^2} (\boldsymbol{\beta}_s \circ \mathbf{h}_r)(\boldsymbol{\beta}_s \circ \mathbf{h}_r)^T, \tag{62}$$

$$\mathbf{F}_3 = \mathbf{0}, \text{ and } \mathbf{F}_4 = \begin{bmatrix} \mathbf{0} & 0 & -2 & 0 \\ \mathbf{0} & 2 & 0 & 0 \\ \mathbf{0} & 0 & 0 & 0 \end{bmatrix}. \tag{63}$$

The optimization problem (60) can now be reformulated equivalently as

$$\min_{\mathbf{b}} \quad \mathbf{a}_0^T \mathbf{b}$$

$$\text{s. t.} \quad \frac{1}{2}\mathbf{b}^T \mathbf{F}_i \mathbf{b} + \mathbf{a}_i^T \mathbf{b} \le c_i \quad i = 1, \cdots, 4$$

$$\mathbf{k} \in \mathbb{Z}^2, \ \mathbf{t} \in \mathbb{Z}^2, \ \Delta_1 \in \mathbb{R} \tag{64}$$

The equivalent optimization problem (64) is a MIQP problem with quadratic constraints [26]. 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 [26]. 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 $\mathbf{t}$.

In order to transform the optimization problem (64) into one that is MIQP-compatible (i.e., in which all the quadratic constraints are associated with semi-definite matrices), we replace the quadratic constraint (60e) with one that is linear, as follows. We introduce the following two real-valued vectors $\tilde{\mathbf{k}} = [\tilde{k}_a, \tilde{k}_b]^T \in \mathbb{R}^2$ and $\tilde{\mathbf{t}} = [\tilde{t}_a, \tilde{t}_b]^T \in \mathbb{R}^2$ defined such that they satisfy

$$\mathbf{k} = \boldsymbol{\kappa} \circ \exp(\tilde{\mathbf{k}}), \quad \mathbf{t} = \boldsymbol{\tau} \circ \exp(\tilde{\mathbf{t}}), \tag{65}$$

where $\boldsymbol{\kappa} = [\kappa_a, \kappa_b]^T \in \mathbb{R}^2$ and $\boldsymbol{\tau} = [\tau_a, \tau_b]^T \in \mathbb{R}^2$ are constant vectors to be chosen appropriately. Thus, the constraint (60e) can now be rewritten equivalently as

$$|\det(\mathbf{k}, \mathbf{t})| \geq 1 \quad \text{iff} \quad |\kappa_a\tau_b\exp(\tilde{k}_a + \tilde{t}_b) - \kappa_b\tau_a\exp(\tilde{k}_b + \tilde{t}_a)| \geq 1. \tag{66}$$

Now, we linearize the constraint (66) by selecting the constant vectors $\boldsymbol{\kappa}$ and $\boldsymbol{\tau}$ such that the first order Taylor series approximations $\exp(\tilde{\mathbf{k}}) \approx \mathbf{1} + \tilde{\mathbf{k}}$ and $\exp(\tilde{\mathbf{t}}) \approx \mathbf{1} + \tilde{\mathbf{t}}$ hold. Hence, the constraint (60e) can be rewritten as

$$|\det(\mathbf{k}, \mathbf{t})| \geq 1 \quad \text{iff} \quad |\kappa_a\tau_b(1 + \tilde{k}_a + \tilde{t}_b) - \kappa_b\tau_a(1 + \tilde{k}_b + \tilde{t}_a)| \gtrsim 1. \tag{67}$$

Note that the constraint (67) is now linear. The optimization problem (64) has the same solution as the following problem which is MIQP-compatible,

$$\min_{\mathbf{k},\,\mathbf{t},\,\Delta_1} \quad \Delta_1 \tag{68a}$$

$$\text{s. t.} \quad \|\mathbf{t}\|^2 - \frac{P((\boldsymbol{\beta}_s \circ \mathbf{h}_d)^T\mathbf{t})^2}{N + P\|\boldsymbol{\beta}_s \circ \mathbf{h}_d\|^2} - \Delta_1 \leq 0 \tag{68b}$$

$$\|\mathbf{k}\|^2 - \frac{P((\boldsymbol{\beta}_s \circ \mathbf{h}_r)^T\mathbf{k})^2}{N + P\|\boldsymbol{\beta}_s \circ \mathbf{h}_r\|^2} - \Delta_1 \leq 0 \tag{68c}$$

$$\frac{N}{N + \beta_r^2|h_{rd}|^2P} - \Delta_1 \leq 0 \tag{68d}$$

$$-|\kappa_a\tau_b(1 + \tilde{k}_a + \tilde{t}_b) - \kappa_b\tau_a(1 + \tilde{k}_b + \tilde{t}_a)| \lesssim -1 \tag{68e}$$

$$\frac{k_i}{\kappa_i} - 1 - \tilde{k}_i \leq 0, \quad -\frac{k_i}{\kappa_i} + 1 + \tilde{k}_i \leq 0, \quad i = a, b \tag{68f}$$

$$\frac{t_i}{\tau_i} - 1 - \tilde{t}_i \leq 0, \quad -\frac{t_i}{\tau_i} + 1 + \tilde{t}_i \leq 0, \quad i = a, b \tag{68g}$$

$$\mathbf{k},\,\mathbf{t} \in \mathbb{Z}^2,\ \tilde{\mathbf{k}},\,\tilde{\mathbf{t}},\,\boldsymbol{\kappa},\,\boldsymbol{\tau} \in \mathbb{R}^2, \Delta_1 \in \mathbb{R}. \tag{68h}$$

The optimization problem (68) can be solved iteratively using Algorithm A-1 hereinafter.

3) Power Allocation Policy: Let us now focus on the problem of finding an appropriate preprocessing vector $\boldsymbol{\beta}$ for given integer vectors $\mathbf{k} \in \mathbb{Z}^2$ and $\mathbf{t} \in \mathbb{Z}^2$. Again, investigating the objective function in (59), it can easily be seen that this

---

Algorithm A-1 Integer coefficients selection for $R_{\text{sum}}^{\text{CoF}}$ as given by (59)

1: Initialization: set $\iota_1 = 1$.

2: Use the branch-and-bound algorithm of [27], [28] to solve for $\Delta_1^{(\iota_1)}$, $\mathbf{k}^{(\iota_1)}$ and $\mathbf{t}^{(\iota_1)}$ with the constraint (68e) substituted with $-\kappa_a \tau_b (1 + \tilde{k}_a + \tilde{t}_b) + \kappa_b \tau_a (1 + \tilde{k}_b + \tilde{t}_a) \leq -1$.

3: Update the values of $\boldsymbol{\kappa}$ and $\boldsymbol{\tau}$ in a way to satisfy (65); and increment the iteration index as $\iota_1 = \iota_1 + 1$.

4: Terminate if $\exp(\tilde{\mathbf{k}}^{(\iota_1)}) \approx \mathbf{1} + \tilde{\mathbf{k}}^{(\iota_1)}$ and $\exp(\tilde{\mathbf{t}}^{(\iota_1)}) \approx \mathbf{1} + \tilde{\mathbf{t}}^{(\iota_1)}$. Denote the found solution as $\Delta_1^{\min,1}$.

5: Redo steps 1 to 4 with in Step 2 the constraint (68e) substituted with $\kappa_a \tau_b (1 + \tilde{k}_a + \tilde{t}_b) - \kappa_b \tau_a (1 + \tilde{k}_b + \tilde{t}_a) \leq -1$. In Step 4, denote the found solution as $\Delta_1^{\min,2}$.

6: Select the integer coefficients corresponding to the minimum among $\Delta_1^{\min,1}$ and $\Delta_1^{\min,2}$.

---

problem can be equivalently stated as

$$\min_{\boldsymbol{\beta}, \Delta_2} \quad \Delta_2 \tag{69a}$$

$$\text{s. t.} \quad \Delta_2 \geq \|\mathbf{t}\|^2 - \frac{P((\boldsymbol{\beta}_s \circ \mathbf{h}_d)^T \mathbf{t})^2}{N + P\|\boldsymbol{\beta}_s \circ \mathbf{h}_d\|^2} \tag{69b}$$

$$\Delta_2 \geq \|\mathbf{k}\|^2 - \frac{P((\boldsymbol{\beta}_s \circ \mathbf{h}_r)^T \mathbf{k})^2}{N + P\|\boldsymbol{\beta}_s \circ \mathbf{h}_r\|^2} \tag{69c}$$

$$\Delta_2 \geq \frac{N}{N + P|h_{rd}|^2 \beta_r^2}, \tag{69d}$$

$$-\sqrt{\frac{P_i}{P}} \leq \beta_i \leq \sqrt{\frac{P_i}{P}}, \quad i = a, b, r \tag{69e}$$

$$\boldsymbol{\beta}_s \in \mathbb{R}^2, \; \boldsymbol{\beta} \in \mathbb{R}^3, \; \Delta_2 \in \mathbb{R}. \tag{69f}$$

Here, similarly to the previous section, Δ_2 is simultaneously an extra optimization variable and the objective function in (69). Also, it is easy to see that the value of $\boldsymbol{\beta}$ that achieves the minimum value of Δ_2 also achieves a maximum value of the objective function in (59).

The optimization problem in (69) is non-linear and non-convex. We use geometric programming [17] for solving it. Geometric programming is a special form of convex optimization for which efficient algorithms have been developed and are known in the related literature [29]. 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 [29]

$$\text{minimize} \quad f_0(\boldsymbol{\beta}, \Delta_2) \tag{70a}$$

$$\text{subject to} \quad f_j(\boldsymbol{\beta}, \Delta_2) \leq 1, \quad j = 1, \cdots, J, \tag{70b}$$

$$g_l(\boldsymbol{\beta}, \Delta_2) = 1, \quad l = 1, \cdots, L, \tag{70c}$$

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 $\boldsymbol{\beta}$ and Δ_2. In its standard form, (70) 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 (70) into one that is equivalent and convex. That is, (70) is a GP nonlinear, nonconvex optimization problem that can be transformed into a nonlinear, convex optimization problem.

In our case, in the problem (69), the constraints (69b) and (69c) contain functions that are non posynomial. Also, the variables in (69) are not all positive, thus preventing a direct application of logarithmic transformation. In what follows, we first transform the problem (69) into one equivalent in which the constraints involve functions that are all posynomial and the variables are all positive; and then we develop an algorithm for solving the equivalent problem.

Let $\mathbf{c} = [c_a, c_b, c_r]^T \in \mathbb{R}^3$ and $\boldsymbol{\delta} = [\delta_a, \delta_b, \delta_r]^T \in \mathbb{R}^3$, such that $c_i > \sqrt{P_i/P}$ and $\delta_i = \beta_i + c_i$ for $i = a, b, r$. Note that the elements of $\boldsymbol{\delta}$ are all strictly positive. Also, for convenience, we define the following functions, for $\mathbf{z} = [z_a, z_b] \in \mathbb{Z}^2$,

$$\psi_1^i(\boldsymbol{\delta}, \Delta_2, \mathbf{z}) = 2\Delta_2 P\left(|h_{ai}|^2 \delta_a c_a + |h_{bi}|^2 \delta_b c_b\right) + P\left(z_a^2 + z_b^2\right)\left(|h_{ai}|^2(\delta_a^2 + c_a^2) + |h_{bi}|^2(\delta_b^2 + c_b^2)\right)$$

$$+ 2P\left(|h_{ai}|^2 z_a^2 \delta_a c_a + |h_{bi}|^2 z_b^2 \delta_b c_b + h_{ai} h_{bi} z_a z_b(\delta_a c_b + \delta_b c_a)\right) + N\left(z_a^2 + z_b^2\right)$$

$$\psi_2^i(\boldsymbol{\delta}, \Delta_2, \mathbf{z}) = \Delta_2\left(N + P|h_{ai}|^2(\delta_a^2 + c_a^2) + P|h_{bi}|^2(\delta_b^2 + c_b^2)\right) + 2P\left(z_a^2 + z_b^2\right)\left(|h_{ai}|^2 \delta_a c_a + |h_{bi}|^2 \delta_b c_b\right)$$

$$+ P\left(|h_{ai}|^2 z_a^2(\delta_a^2 + c_a^2) + |h_{bi}|^2 z_b^2(\delta_b^2 + c_b^2) + 2h_{ai} h_{bi} z_a z_b(\delta_a \delta_b + c_a c_b)\right). \tag{71}$$

Let us now define the following functions, $f_1(\boldsymbol{\delta}, \Delta_2) = \psi_1^d(\boldsymbol{\delta}, \Delta_2, \mathbf{t})$, $f_2(\boldsymbol{\delta}, \Delta_2) = \psi_1^r(\boldsymbol{\delta}, \Delta_2, \mathbf{k})$, $g_1(\boldsymbol{\delta}, \Delta_2) = \psi_2^d(\boldsymbol{\delta}, \Delta_2, \mathbf{t})$, $g_2(\boldsymbol{\delta}, \Delta_2) = \psi_2^r(\boldsymbol{\delta}, \Delta_2, \mathbf{k})$, $f_3(\boldsymbol{\delta}, \Delta_2) = N + 2\Delta_2 P|h_{rd}|^2 \delta_r c_r$, and $g_3(\boldsymbol{\delta}, \Delta_2) = \Delta_2\left(N + P|h_{rd}|^2(\delta_r^2 + c_r^2)\right)$.

It is now easy to see that the optimization problem (69) can be stated in the following form.

$$\min_{\boldsymbol{\delta}, \Delta_2} \quad \Delta_2 \tag{72a}$$

$$\text{s. t.} \quad \frac{f_1(\boldsymbol{\delta}, \Delta_2)}{g_1(\boldsymbol{\delta}, \Delta_2)} \leq 1, \quad \frac{f_2(\boldsymbol{\delta}, \Delta_2)}{g_2(\boldsymbol{\delta}, \Delta_2)} \leq 1, \quad \frac{f_3(\boldsymbol{\delta}, \Delta_2)}{g_3(\boldsymbol{\delta}, \Delta_2)} \leq 1 \tag{72b}$$

$$-\sqrt{\frac{P_i}{P}} + c_i \leq \delta_i \leq \sqrt{\frac{P_i}{P}} + c_i, \quad i = a, b, r \tag{72c}$$

$$\boldsymbol{\delta} \in \mathbb{R}^3, \quad \mathbf{c} \in \mathbb{R}^3, \quad \Delta_2 \in \mathbb{R}. \tag{72d}$$

The constraints (72b) 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 [29]. One can transform a Complementary GP problem into a GP problem using series of approximations. In order to get posynomials, we approximate the functions $g_1(\boldsymbol{\delta}, \Delta_2)$, $g_2(\boldsymbol{\delta}, \Delta_2)$ and $g_3(\boldsymbol{\delta}, \Delta_2)$ with monomials, by using the following lemma [17].

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

$$g(\boldsymbol{\delta}, \Delta_2) \geq \tilde{g}(\boldsymbol{\delta}, \Delta_2) = \prod_j \left(\frac{u_j(\boldsymbol{\delta}, \Delta_2)}{\gamma_j}\right)^{\gamma_j}. \tag{73}$$

Here, $\gamma_j = u_j(\boldsymbol{\delta}^{(0)}, \Delta_2^{(0)})/g(\boldsymbol{\delta}^{(0)}, \Delta_2^{(0)})$, $\forall j$, for any fixed positive $\boldsymbol{\delta}^{(0)}$ and $\Delta_2^{(0)}$ then $\tilde{g}(\boldsymbol{\delta}^{(0)}, \Delta_2^{(0)}) = g(\delta^{(0)}, \Delta_2^{(0)})$, and $\tilde{g}(\boldsymbol{\delta}^{(0)}, \Delta_2^{(0)})$ is the best local monomial approximation to $g(\boldsymbol{\delta}^{(0)}, \Delta_2^{(0)})$ near $\boldsymbol{\delta}^{(0)}$ and $\Delta_2^{(0)}$.

Let $\tilde{g}_1(\boldsymbol{\delta}, \Delta_2)$, $\tilde{g}_2(\boldsymbol{\delta}, \Delta_2)$ and $\tilde{g}_3(\boldsymbol{\delta}, \Delta_2)$ be the monomial approximations of the functions $g_1(\boldsymbol{\delta}, \Delta_2)$, $g_2(\boldsymbol{\delta}, \Delta_2)$ and $g_3(\boldsymbol{\delta}, \Delta_2)$ obtained using Lemma 1. Using these monomial approximations, the ratios of posynomials involved in the constraint (72b) can be upper bounded by posynomials. The optimal solution of the problem obtained using the convex approximations is also optimal for the original problem (69), i.e., satisfies the Karush-Kuhn-Tucker (KKT) conditions of the original problem (69), if the applied approximations satisfy the following three properties [30], [17]:

1) $g_j(\boldsymbol{\delta}, \Delta_2) \leq \tilde{g}_j(\boldsymbol{\delta}, \Delta_2)$ for all $\boldsymbol{\delta}$ and Δ_2 where $\tilde{g}_j(\boldsymbol{\delta}, \Delta_2)$ is the approximation of $g_j(\boldsymbol{\delta}, \Delta_2)$.

2) $g_j(\boldsymbol{\delta}^{(0)}, \Delta_2^{(0)}) = \tilde{g}_j(\boldsymbol{\delta}^{(0)}, \Delta_2^{(0)})$ where $\boldsymbol{\delta}^{(0)}$ and $\Delta_2^{(0)}$ are the optimal solution of the approximated problem in the previous iteration.

3) $\triangledown g_j(\boldsymbol{\delta}^{(0)}, \Delta_2^{(0)}) = \triangledown \tilde{g}_j(\boldsymbol{\delta}^{(0)}, \Delta_2^{(0)})$, where $\triangledown g_j(\cdot)$ stands for the gradient of function $g_j(\cdot)$.

Summarizing, applying the aforementioned transformations, we transformed the original optimization problem (69) first into a Complementary GP problem (72) and then into a GP problem by applying the convex approximations (73). Finally, the obtained GP problem can be solved easily using, e.g., an interior point approach. More specifically, the problem of finding the appropriate preprocessing vector $\boldsymbol{\delta}$ for given integer vectors $\mathbf{k}$ and $\mathbf{t}$ can be solved using Algorithm A-2 hereinafter.

Algorithm A-2 Power allocation policy for $R_{\text{sum}}^{\text{CoF}}$ as given by (59)

1: Set $\boldsymbol{\delta}^{(0)}$ to some initial value. Compute $\Delta_2^{(0)}$ using $\boldsymbol{\delta}^{(0)}$ and set $\iota_2 = 1$.
2: Approximate $g(\boldsymbol{\delta}^{(\iota_2)}, \Delta_2^{(\iota_2)})$ with $\tilde{g}(\boldsymbol{\delta}^{(\iota_2)}, \Delta_2^{(\iota_2)})$ around $\boldsymbol{\delta}^{(\iota_2-1)}$ and $\Delta_2^{(\iota_2-1)}$ using (73).
3: Solve the resulting approximated GP problem using an interior point approach. Denote the found solutions as $\boldsymbol{\delta}^{(\iota_2)}$ and $\Delta_2^{(\iota_2)}$.
4: Increment the iteration index as $\iota_2 = \iota_2 + 1$ and go back to Step 2 using $\boldsymbol{\delta}$ and Δ_2 of step 3.
5: Terminate if $\|\boldsymbol{\delta}^{(\iota_2)} - \boldsymbol{\delta}^{(\iota_2-1)}\| \leq \epsilon$ or if $\Delta_2^{(\iota_2)} \geq \Delta_2^{(\iota_2-1)}$.

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

The algorithms that we develop in this section to solve the optimization problem involved in the maximization of the sum rate given in 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 $R_{\text{sum}}^{\text{CoD}}$ as given by (41) in Proposition 2. The optimization problem can be stated as:

$$(\text{B}) \; : \; \max \frac{1}{2} \min \left\{ \log^+ \left(\frac{\text{snr}}{\text{snr}\|\boldsymbol{\beta}_s \circ \mathbf{H}^T \boldsymbol{\alpha}_t - \mathbf{t}\|^2 + (\boldsymbol{\alpha}_t \circ \boldsymbol{\alpha}_t)^T \mathbf{n}_d} \right), \right.$$
$$\left. \log^+ \left(\frac{\text{snr}}{\text{snr}\|\boldsymbol{\beta}_s \circ \mathbf{H}^T \boldsymbol{\alpha}_k - \mathbf{k}\|^2 + (\boldsymbol{\alpha}_k \circ \boldsymbol{\alpha}_k)^T \mathbf{n}_d} \right) \right\}, \tag{74}$$

where the distortion D is given by

$$D = \frac{N^2 \left(1 + \text{snr}\|\boldsymbol{\beta}_s \circ \mathbf{h}_r\|^2 \right)}{|h_{rd}|^2 P_r} - \frac{N^2 \left(\text{snr}(\boldsymbol{\beta}_s \circ \mathbf{h}_r)^T (\boldsymbol{\beta}_s \circ \mathbf{h}_d) \right)^2}{|h_{rd}|^2 P_r \left(1 + \text{snr}\|\boldsymbol{\beta}_s \circ \mathbf{h}_d\|^2 \right)}, \tag{75}$$

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 $|\det(\mathbf{k}, \mathbf{t})| \geq 1$.

In order to compute $R_{\text{sum}}^{\text{CoD}}$ as given by (74), we develop the following iterative algorithm which optimizes the integer coefficients and the powers alternatively, and to which we refer to as "Algorithm B" in reference to the optimization problem (B).

Algorithm B Iterative algorithm for computing $R_{\text{sum}}^{\text{CoD}}$ as given by (74)

1: Choose an initial feasible vector $\boldsymbol{\beta}_{\text{s}}^{(0)}$ and set $\iota = 1$.

2: Solve (74) with $\boldsymbol{\beta}_{\text{s}} = \boldsymbol{\beta}_{\text{s}}^{(\iota-1)}$ 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 (74) with $\mathbf{k} = \mathbf{k}^{(\iota)}$ and $\mathbf{t} = \mathbf{t}^{(\iota)}$ for the optimal $\boldsymbol{\beta}_{\text{s}}$ using Algorithm B-2 and assign it to $\boldsymbol{\beta}_{\text{s}}^{(\iota)}$.

4: Increment the iteration index as $\iota = \iota + 1$ and go back to Step 2.

5: Terminate if $\|\boldsymbol{\beta}_{\text{s}}^{(\iota)} - \boldsymbol{\beta}_{\text{s}}^{(\iota-1)}\| \le \epsilon$, $|R_{\text{sum}}^{\text{CoD}}[\iota] - R_{\text{sum}}^{\text{CoD}}[\iota-1]| \le \epsilon$ or if $R_{\text{sum}}^{\text{CoD}}[\iota] \le R_{\text{sum}}^{\text{CoD}}[\iota-1]$.

---

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

$$\min_{\mathbf{k}, \mathbf{t}, \Theta_1} \quad \Theta_1 \tag{76a}$$

$$\text{s. t.} \quad \Theta_1 \ge \mathbf{t}^T \boldsymbol{\Omega} \mathbf{t} \tag{76b}$$

$$\Theta_1 \ge \mathbf{k}^T \boldsymbol{\Omega} \mathbf{k} \tag{76c}$$

$$\det(\mathbf{k}, \mathbf{t}) = |k_a t_b - k_b t_a| \ge 1 \tag{76d}$$

$$\mathbf{k}, \mathbf{t} \in \mathbb{Z}^2, \ \Theta_1 \in \mathbb{R}, \tag{76e}$$

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 (76) into a MIQP problem with quadratic constraints [26] as before. We 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

$$\mathbf{F}_1 = \begin{bmatrix} 2\boldsymbol{\Omega} & \mathbf{0} \\ \mathbf{0} & 0 \end{bmatrix}, \qquad \mathbf{F}_2 = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & 2\boldsymbol{\Omega} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & 0 \end{bmatrix}, \text{ and } \mathbf{F}_3 = \begin{bmatrix} 0 & 0 & -2 & 0 \\ 0 & 2 & 0 & 0 \\ \mathbf{0} & 0 & 0 & 0 \end{bmatrix}. \tag{77}$$

The optimization problem (76) can now be reformulated equivalently as

$$\min_{\mathbf{b}} \quad \mathbf{a}_0^T \mathbf{b}$$

$$\text{s. t.} \quad \frac{1}{2}\mathbf{b}^T \mathbf{F}_i \mathbf{b} + \mathbf{a}_i^T \mathbf{b} \le c_i \ \ i = 1, 2, 3$$

$$\mathbf{k} \in \mathbb{Z}^2, \ \mathbf{t} \in \mathbb{Z}^2, \ \Theta_1 \in \mathbb{R} \tag{78}$$

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 $\mathbf{t}$. In order to transform the optimization problem (78) into one that is MIQP-compatible (i.e., in which all the quadratic constraints are associated with semi-definite matrices), we replace the quadratic constraint (76d) with one that is linear, as performed in the previous section.

Finally, the MIQP optimization problem is given by,

$$\min_{\mathbf{k},\, \mathbf{t},\, \Theta_1} \quad \Theta_1 \tag{79a}$$

$$\text{s. t.} \quad \Theta_1 \geq \mathbf{t}^T \mathbf{\Omega} \mathbf{t} \tag{79b}$$

$$\Theta_1 \geq \mathbf{k}^T \mathbf{\Omega} \mathbf{k} \tag{79c}$$

$$-\left|\kappa_a \tau_b \left(1 + \tilde{k}_a + \tilde{t}_b\right) - \kappa_b \tau_a \left(1 + \tilde{k}_b + \tilde{t}_a\right)\right| \lesssim -1 \tag{79d}$$

$$\frac{k_i}{\kappa_i} - 1 - \tilde{k}_i \leq 0, \quad -\frac{k_i}{\kappa_i} + 1 + \tilde{k}_i \leq 0, \quad i = a, b \tag{79e}$$

$$\frac{t_i}{\tau_i} - 1 - \tilde{t}_i \leq 0, \quad -\frac{t_i}{\tau_i} + 1 + \tilde{t}_i \leq 0, \quad i = a, b \tag{79f}$$

$$\mathbf{k},\, \mathbf{t} \in \mathbb{Z}^2,\; \tilde{\mathbf{k}},\, \tilde{\mathbf{t}},\, \boldsymbol{\kappa},\, \boldsymbol{\tau} \in \mathbb{R}^2, \Theta_1 \in \mathbb{R}. \tag{79g}$$

The optimization problem (79) can be solved iteratively using Algorithm B-1 hereinafter.

Algorithm B-1 Integer coefficients selection for $R_{\text{sum}}^{\text{CoD}}$ as given by (74)

1: Initialization: set $\iota_1 = 1$.

2: Use the branch-and-bound algorithm of [27], [28] to solve for $\Theta_1^{(\iota_1)}$, $\mathbf{k}^{(\iota_1)}$ and $\mathbf{t}^{(\iota_1)}$ with the constraint (79d) substituted with $-\kappa_a \tau_b \left(1 + \tilde{k}_a + \tilde{t}_b\right) + \kappa_b \tau_a \left(1 + \tilde{k}_b + \tilde{t}_a\right) \leq -1$.

3: Update the values of $\boldsymbol{\kappa}$ and $\boldsymbol{\tau}$ in a way to satisfy (65); and increment the iteration index as $\iota_1 = \iota_1 + 1$.

4: Terminate if $\exp(\tilde{\mathbf{k}}^{(\iota_1)}) \approx \mathbf{1} + \tilde{\mathbf{k}}^{(\iota_1)}$ and $\exp(\tilde{\mathbf{t}}^{(\iota_1)}) \approx \mathbf{1} + \tilde{\mathbf{t}}^{(\iota_1)}$. Denote the found solution as $\Theta_1^{\min,2}$.

5: Redo steps 1 to 4 with in Step 2 the constraint (79d) substituted with $\kappa_a \tau_b \left(1 + \tilde{k}_a + \tilde{t}_b\right) - \kappa_b \tau_a \left(1 + \tilde{k}_b + \tilde{t}_a\right) \leq -1$. In Step 4, denote the found solution as $\Theta_1^{\min,2}$.

6: 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 β_s for a fixed integer coefficients $\mathbf{k}$, and $\mathbf{t}$, can be written as,

$$\min_{\boldsymbol{\beta}_s,\, \boldsymbol{\alpha}_t,\, \boldsymbol{\alpha}_k,\, \Theta_2} \quad \Theta_2 \tag{80a}$$

$$\text{s. t.} \quad \Theta_2 \geq \frac{\text{snr}\|\boldsymbol{\beta}_s \circ \mathbf{H}^T \boldsymbol{\alpha}_t - \mathbf{t}\|^2 + (\boldsymbol{\alpha}_t \circ \boldsymbol{\alpha}_t)^T \mathbf{n}_d}{\text{snr}}, \tag{80b}$$

$$\Theta_2 \geq \frac{\text{snr}\|\boldsymbol{\beta}_s \circ \mathbf{H}^T \boldsymbol{\alpha}_k - \mathbf{k}\|^2 + (\boldsymbol{\alpha}_k \circ \boldsymbol{\alpha}_k)^T \mathbf{n}_d}{\text{snr}}, \tag{80c}$$

$$D \geq \frac{N^2 \left(1 + \text{snr}\|\boldsymbol{\beta_s} \circ \mathbf{h}_r\|^2\right)}{|h_{rd}|^2 P_r} - \frac{N^2 \left(\text{snr}(\boldsymbol{\beta_s} \circ \mathbf{h}_r)^T (\boldsymbol{\beta_s} \circ \mathbf{h}_d)\right)^2}{|h_{rd}|^2 P_r \left(1 + \text{snr}\|\boldsymbol{\beta_s} \circ \mathbf{h}_d\|^2\right)} \tag{80d}$$

$$-\sqrt{\frac{P_i}{P}} \leq \beta_i \leq \sqrt{\frac{P_i}{P}}, \quad i = a, b \tag{80e}$$

$$\boldsymbol{\beta}_s \in \mathbb{R}^2, \Theta_2 \in \mathbb{R}. \tag{80f}$$

Algorithm B-2 Power allocation policy for $R_{\text{sum}}^{\text{CoD}}$ as given by (74)

1: Set $\boldsymbol{\delta_s}^{(0,0)}$ 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 $\boldsymbol{\delta_s}^{(\iota_2-1,\iota_3)}$.

3: Approximate $g(\boldsymbol{\delta}^{(\iota_2,\iota_3)}, \Theta_2^{(\iota_2,\iota_3)})$ with $\tilde{g}(\boldsymbol{\delta_s}^{(\iota_2,\iota_3)}, \Theta_2^{(\iota_2,\iota_3)})$ around $\boldsymbol{\delta}^{(\iota_2-1,\iota_3)}$ and $\Theta_2^{(\iota_2-1,\iota_3)}$ using (73).

4: Solve the resulting approximated GP problem using an interior point approach. Denote the found solutions as $\boldsymbol{\delta_s}^{(\iota_2,\iota_3)}$ 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 $\boldsymbol{\delta_s}$ and Θ_2 of step 4.

6: Terminate if $\|\boldsymbol{\delta_s}^{(\iota_2,\iota_3)} - \boldsymbol{\delta_s}^{(\iota_2-1,\iota_3)}\| \le \epsilon$ or if $\Theta_2^{(\iota_2,\iota_3)} \ge \Theta_2^{(\iota_2-1,\iota_3)}$ and denote by $\boldsymbol{\delta}$ the final value.

7: Increment the iteration index as $\iota_3 = \iota_3 + 1$, set $\iota_2 = 1$, and $\boldsymbol{\delta_s}^{(\iota_2-1,\iota_3)} = \boldsymbol{\delta}$ and then go back to Step 2.

8: Terminate if $|R_{\text{sum}}^{\text{CoD}}[\iota_3] - R_{\text{sum}}^{\text{CoD}}[\iota_3 - 1]| \le \epsilon$ or if $R_{\text{sum}}^{\text{CoD}}[\iota_3] \le R_{\text{sum}}^{\text{CoD}}[\iota_3 - 1]$.

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 $i = a, b$. Note that the elements of $\boldsymbol{\delta}$ are all strictly positive. We can reformulate the optimization problem as,

$$\min_{\boldsymbol{\delta_s}, \boldsymbol{\alpha}_t, \boldsymbol{\alpha}_k, \Theta_2} \quad \Theta_2 \tag{81a}$$

$$\text{s. t.} \quad \frac{f_1(\boldsymbol{\delta}, \Theta_2, \boldsymbol{\alpha}_t, \boldsymbol{\alpha}_k)}{g_1(\boldsymbol{\delta}, \Theta_2, \boldsymbol{\alpha}_t, \boldsymbol{\alpha}_k)} \le 1, \quad \frac{f_2(\boldsymbol{\delta}, \Theta_2, \boldsymbol{\alpha}_t, \boldsymbol{\alpha}_k)}{g_2(\boldsymbol{\delta}, \Theta_2, \boldsymbol{\alpha}_t, \boldsymbol{\alpha}_k)} \le 1, \quad \frac{f_3(\boldsymbol{\delta}, \Theta_2)}{g_3(\boldsymbol{\delta}, \Theta_2)} \le 1 \tag{81b}$$

$$-\sqrt{\frac{P_i}{P}} + c_i \le \delta_i \le \sqrt{\frac{P_i}{P}} + c_i \ , \ i = a, b \tag{81c}$$

$$\boldsymbol{\delta_s} \in \mathbb{R}^2, \ \mathbf{c} \in \mathbb{R}^2, \ \Theta_2 \in \mathbb{R}. \tag{81d}$$

The constraints (81b) correspond to the constraints (80b), (80c), and (80d). These functions consist of ratios of posynomials, i.e., are not posynomial — recall that a ratio of posynomials is in general non posynomial. As before, we transform the Complementary GP problem into a GP problem using series of approximations.

As there are several variables to optimize over simultaneously, the optimization is carried out in two steps. First, we optimize the power value $\boldsymbol{\delta_s}$ using geometric programming with successive convex approximation in a way that is essentially similar to the above, for a fixed value of $\boldsymbol{\alpha}_t$ and $\boldsymbol{\alpha}_k$. Next, we optimize the value of $\boldsymbol{\alpha}_t$ and $\boldsymbol{\alpha}_k$ for a fixed value of $\boldsymbol{\delta}_s$. This process is repeated until convergence. More specifically, the problem of finding the appropriate preprocessing vector $\boldsymbol{\delta}_s$ for a given integer vectors $\mathbf{k}$ and $\mathbf{t}$ can be solved using Algorithm B-2.

V. Numerical Examples

In this section, we provide some numerical examples. We measure the performance with the sum rate obtained under the constraint of symmetric rate, i.e., two times the maximum rate at which both users can communicate simultaneously with the destination, in a reliable fashion. 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; that from Transmitter B to the relay is modeled with a zero-mean Gaussian distribution with variance σ_{br}^2; and that from the relay to the destination is modeled with a zero-mean

Gaussian distribution with variance σ_{rd}^2. Similar assumptions and notations are used for the direct links from the transmitters to the destination. 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.
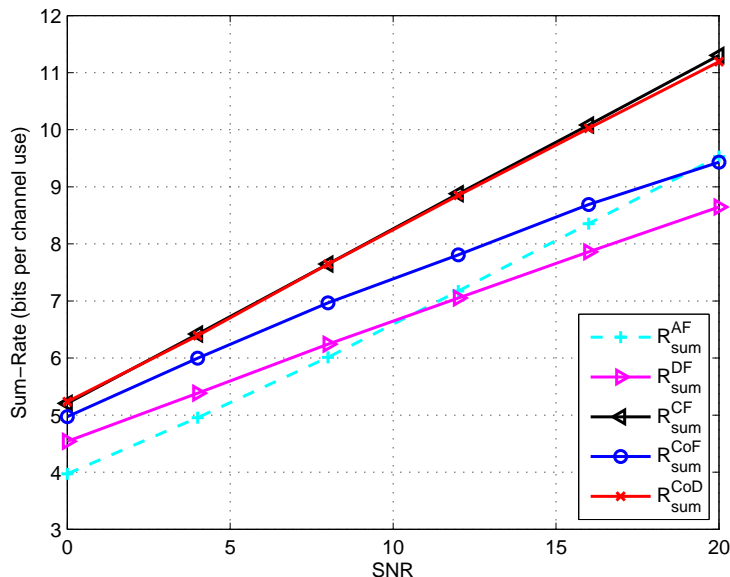

Fig. 2. Achievable sum rates under symmetric rate constraint. Numerical values are $P = 20$ dBW, $\sigma_{\text{ar}}^2 = \sigma_{\text{br}}^2 = \sigma_{\text{rd}}^2 = 20$ dBW, and $\sigma_{\text{ad}}^2 = \sigma_{\text{bd}}^2 = 0$ dBW.

Figure 2 depicts the evolution of the sum rate obtained using the so-called *compute-and-forward at the relay* approach, i.e., the sum rate $R_{\text{sum}}^{\text{CoF}}$ of proposition 1 as given by (22); and the sum rate obtained using the so-called *compute-and-forward at the relay and compute at the destination* approach, i.e., the sum rate $R_{\text{sum}}^{\text{CoD}}$ of proposition 2 as given by (41), as functions of the signal-to-noise ratio SNR $= 10 \log(P/N)$ (in decibels). Note that the curves correspond to numerical values of channel coefficients chosen such that the transmitters-to-relay links are of comparable quality to that of relay-destination link and are better than transmitters-to-destination links. For comparison reasons, the figure also shows the sum rates obtained using the classical strategies of Section II-C, i.e., the sum rate $R_{\text{sum}}^{\text{AF}}$ allowed by standard amplify-and-forward as given by (14), the sum rate $R_{\text{sum}}^{\text{DF}}$ allowed by standard decode-and-forward as given by (16), and the sum rate $R_{\text{sum}}^{\text{CF}}$ allowed by standard compress-and-forward as given by (21). All the sum rates are evaluated under the constraint of symmetric rate by the users.

For the example shown in Figure 2, we observe that both strategies of proposition 1 and proposition 2 perform well. The strategy of proposition 2 achieves a sum rate that is larger than what is obtained using classical relaying techniques, for all ranges of SNR. Also, the strategy of proposition 1 outperforms standard AF and DF for all ranges of SNR. Figure 3 depicts the same curves for other combinations of channel coefficients, chosen such that the transmitters-to-relay links are of better quality than all other links, but the relay-to-destination link is not better than the transmitters-to-destination links. In this case,
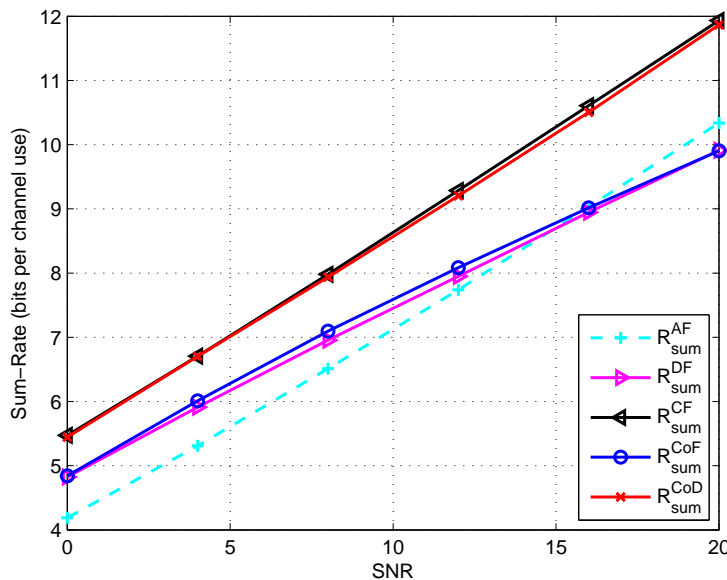
Fig. 3. Achievable sum rates under symmetric rate constraint. Numerical values are $P = 20$ dBW, $\sigma_{ar}^2 = \sigma_{br}^2 = 26$ dBW, $\sigma_{rd}^2 = 14$ dBW, and $\sigma_{ad}^2 = \sigma_{bd}^2 = 0$ dBW.

we observe that the curves show behaviors that are generally similar to those of Figure 2. Also, note that, the standard AF can provide sum rates that are better than those obtained using the strategy of proposition 1 at large SNR is precisely due to that the transmitters-to-destination links are of better quality than the relay-to-destination link for the example shown in Figure 3. (Recall that, although the strategy of proposition 1 does not require decoding the messages from the two users individually, its performance is limited by the bottleneck at the relay node nonetheless; and this explains why it can be outperformed by standard AF at large values of SNR as visible from Figure 3). Figure 4 shows similar curves for other combinations of channel coefficients, chosen such that the relay-to-destination link has better quality than all other links.

Remark 2: Recall that, as we mentioned previously, the optimization Algorithm B associated with the strategy of proposition 2 is non-convex. In the figures shown in this paper, the sum rate provided by this strategy, which is based on Wyner-Ziv compression at the relay and computing appropriate equations at the destination, are obtained by selecting *only* certain initial points for Algorithm B. For this reason, the sum rate offered by the coding strategy of proposition 2, i.e., R_{sum}^{CoD}, can possibly be larger than what is shown in the figures if one considers more initial points. Also, we note that even for initial points choices that yield a sum rate that is only as large as that obtained with standard compress-and-forward, there are advantages for using the coding scheme of proposition 2 instead of regular CF. In particular, while regular CF utilizes the rather complex joint typicality decoding, or maximum-Likelihood decoding, the coding scheme of proposition 2 utilizes linear decoding, and thus is more easily feasible.

Remark 3: 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
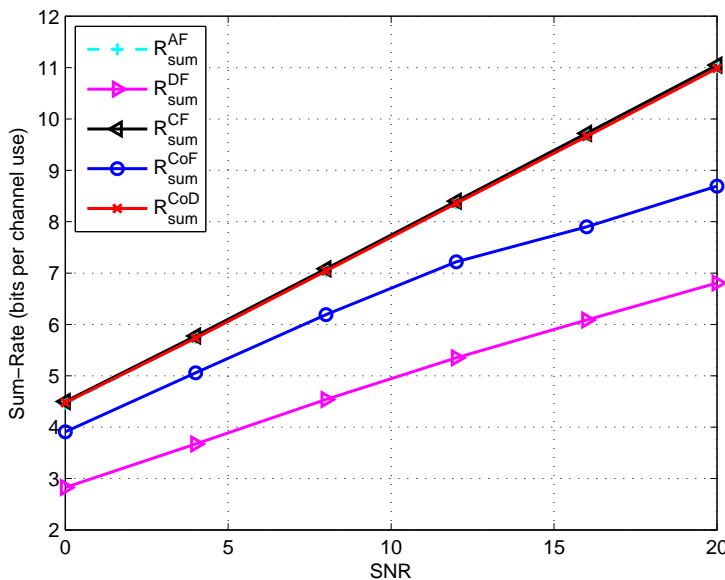
Fig. 4. Achievable sum rates under symmetric rate constraint. Numerical values are $P = 20$ dBW, $\sigma_{ar}^2 = \sigma_{br}^2 = 14$ dBW, $\sigma_{rd}^2 = 26$ dBW, and $\sigma_{ad}^2 = \sigma_{bd}^2 = 0$ dBW.

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, 3 and 4, the situation is reversed for the example shown in Figure 5. Related to this aspect, recall the discussion in Remark 1. Let us denote by $R_{\text{sum}}^{\text{St-CoD}}$ the achievable sum rate obtained by modifying the coding strategy of proposition 2 in a manner to get the first equation computed at the destination using *only* the signal received from the relay during the second transmission period, and the second equation computed at the destination using *only* the direct transmissions from the transmitters during the first transmission period. Figure 6 shows the improvement brought up by the aforementioned joint processing for the example depicted therein.

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 (59), as described in section IV-A. Recall that the algorithm involves allocating the integer coefficients and the users' 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 sources and relay by solving a series of geometric programs by means of an interior point approach [29].

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 sum rate is zero [31]. Let $R_{\text{sum}}^{\text{Ex}}$ denote the sum rate obtained by using the described exhaustive search-based algorithm. Figure 7
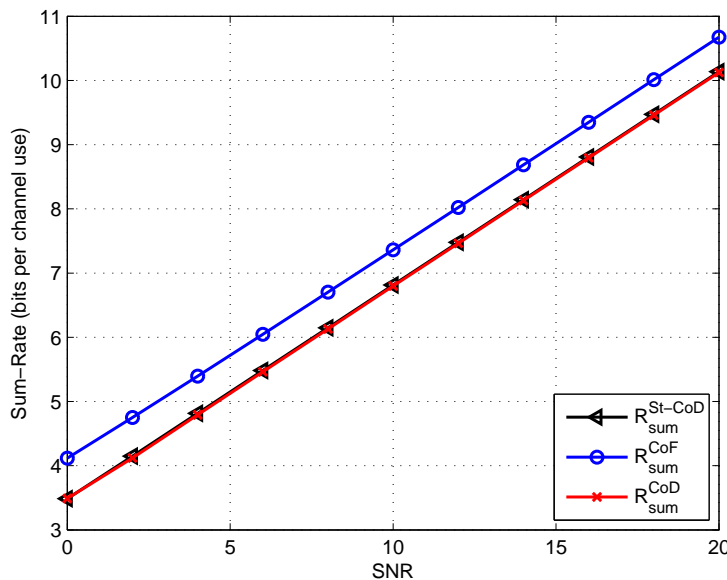
Fig. 5. Achievable sum rates under symmetric rate constraint. Numerical values are $P = 20$ dBW, $\sigma_{ar}^2 = \sigma_{br}^2 = \sigma_{rd}^2 = 20$ dBW, and $\sigma_{ad}^2 = \sigma_{bd}^2 = 0$ dBW.

shows that the number of iterations required for Algorithm A to converge, i.e., yield the same sum-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.

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 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 sum rate. Next, for each of the two coding schemes, we investigate the problem
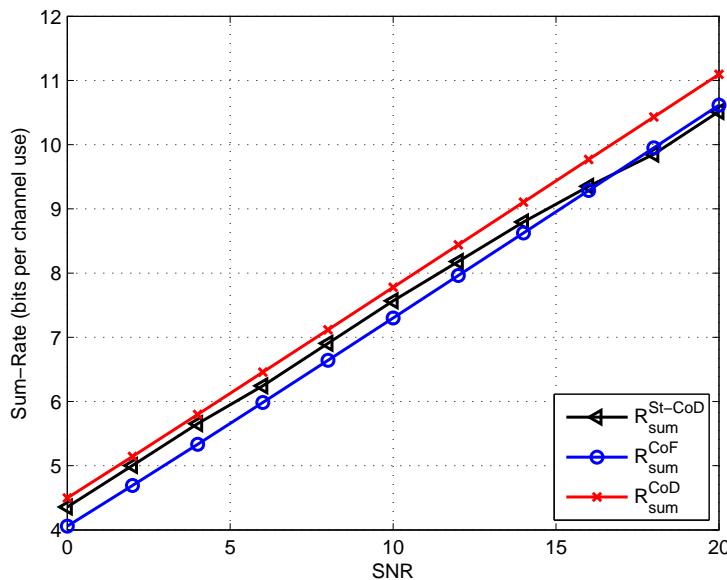
Fig. 6.   Achievable sum rates under symmetric rate constraint. Numerical values are $P = 20$ dBW, $\sigma_{ar}^2 = \sigma_{br}^2 = 14$ dBW, $\sigma_{rd}^2 = 26$ dBW, and $\sigma_{ad}^2 = \sigma_{bd}^2 = 0$ dBW.

of allocating the powers and the integer-valued coefficients of the recovered equations in a way to maximize the offered sum 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 sources and the relay into series of geometric programs. Comparing our coding schemes with classic relaying techniques, we show that the proposed strategies can offer substantial gains; and we illustrate this through some numerical examples.

ACKNOWLEDGMENT

REFERENCES

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

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

[3] P. Elias, D. R. Feinstein, and C. E. Shannon. A note on the maximal flow through a network. *IRE Trans. Inf. Theory*, IT-72:117–119, Dec. 1956.
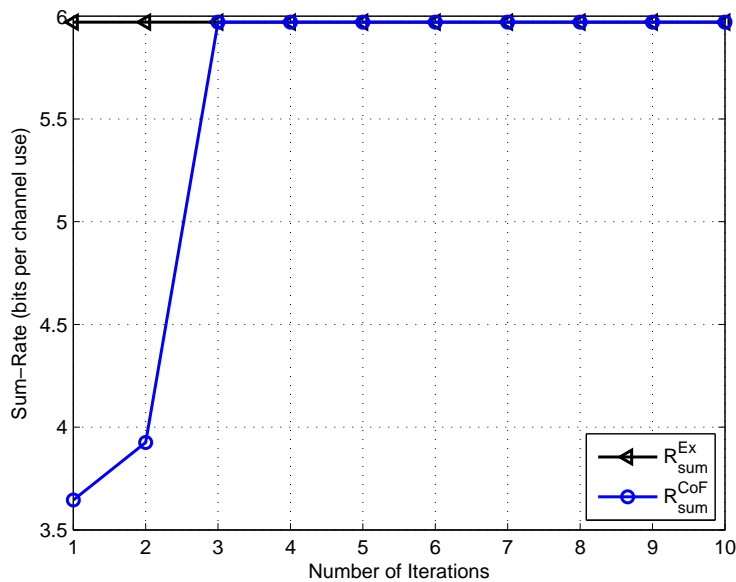
Fig. 7. Algorithm A v.s. Exhaustive search. Numerical values are $P = 20$ dBW, $\sigma_{ar}^2 = \sigma_{br}^2 = \sigma_{rd}^2 = 20$ dBW, and $\sigma_{ad}^2 = \sigma_{bd}^2 = 0$ dBW.

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

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

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

[7] C. Fragouli and E. Soljanin. Network coding fundamentals. *Found. Trends Commun. Info. Theory*, 2:1–333, 2007.

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

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

[10] J. H. Conway and N. J. A. Sloane. *Sphere Packing, Lattices and Groups*. third edition, John Willey & Sons INC., New York, 1988.

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

[12] S. Joon Kim, P. Mitran, and V. Tarokh. Performance bounds for bi-directional coded cooperation protocols. *IEEE Trans. Inf. Theory*, IT-54:5253–5241, Nov. 2008.

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

[14] M. El Soussi, A. Zaidi, and L. Vandendorpe. Resource allocation for multiple access relay channel with a compute-and-

forward relay. In *Int. Symposium on Wireless Communication Systems*, pages 809–813, Nov 2011.

[15] M. El Soussi, A. Zaidi, and L. Vandendorpe. Iterative sum-rate optimization for multiple access relay channels with a compute-and-forward relay. In *IEEE International Conference on Communications*, June 2012.

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

[17] M. Chiang, C. W. Tan, D. P. Palomar, D. O'Neill, and D. Julian. Power control by geometric programming. *IEEE Trans. Wireless Communications*, 6:2640–2651, 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. First IEEE Conference on Sensor and Ad Hoc Communications and Networks*, pages 191–199, Oct. 2004.

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

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

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

[22] G. D. Forney. Coset-codes: Part I: Introduction and geometrical classification. *IEEE Trans. Inf. Theory*, IT-34:1123–1151, 1988.

[23] G. D. Forney. On the role of MMSE estimation in approaching the information-theoretic limits of linear Gaussian channels: Shannon meets Wyner. In *Proc. 41st Annual Allerton Conf. Comm. Control, Computing*, pages 430–439, Monticello, IL, Oct. 2003.

[24] U. Erez, S. Shamai (Shitz), and R. Zamir. Capacity and lattice strategies for cancelling known interference. *IEEE Trans. Inf. Theory*, IT-51:3820–3833, Nov. 2005.

[25] U. Erez and R. Zamir. Achieving $\frac{1}{2}\log(1 + snr)$ on the awgn channel with lattice encoding and decoding. *IEEE Trans. Inf. Theory*, 50:2293–2314, Oct. 2004.

[26] C. A. Floudas. Nonlinear and mixed integer optimization. In *Oxford University Press*, 1995.

[27] E. L. Lawler and D.E. Wood. Branch-and -bound methods: A survey. *Operations Research*, pages 699–719, 1966.

[28] L. A. Wolsey. *Integer programming*. John Wiley and Sons, 1998.

[29] M. Chiang. Geometric programming for communication systems. In *Foundation and Trends in Commun. Inf. Theory*, pages 1–156, Aug. 2005.

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

[31] J. Zhan, B. Nazer, M. Gastpar, and U. Erez. Mimo compute-and-forward. In *IEEE Int. Symposium on Inf. Theory*, pages 2848–2852, June 2009.

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

Mohieddine El Soussi   Abdellatif Zaidi   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 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 sum-rate. Next, we address the power allocation and the selection of the integer-valued coefficients to maximize the offered sum-rate; an iterative coordinate descent method is proposed. The analysis shows that our coding schemes can offer substantial gains over standard relaying techniques. 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] are 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 interference-relay channel [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 be 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. 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 separate nodes, is less easy to achieve. By opposition to 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. 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 the destination are chosen appropriately, the destination can solve for the transmitted codewords. We consider individual power constraints at the transmitters and the relay, and analyse the sum rate offered by this coding scheme [14] [15].

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 Wyner-Ziv compression [16]. 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. 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 sum 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 alternatively. More specifically, we show that the problem of finding appropriate integer coefficients for a given set of powers has the same solution as an approximated 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 the 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 both schemes outperform 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, decode-and-forward or compress-and-forward relaying schemes. 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 sum 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 sum-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 $\mathbf{X}$; and, if $m = n$, $\det(\mathbf{X})$ to designate the determinant 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, we use $\text{Var}(\mathbf{x})$ to denote the power of $\mathbf{x}$ i.e. $\mathbb{E}[\|\mathbf{x}\|^2]$. 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 $(\boldsymbol{\lambda}_1 - \boldsymbol{\lambda}_2) \in \Lambda$. For an n-dimensional lattice $\Lambda \in \mathbb{R}^n$, there exists (at least one) matrix $\mathbf{G} \in \mathbb{R}^{n \times n}$ such that any lattice point $\boldsymbol{\lambda} \in \Lambda$ can be written as an integer combination of the columns of $\mathbf{G}$. The matrix $\mathbf{G}$ is called generator matrix of Λ, and satisfies

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

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.,

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

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.,

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

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 $\Lambda \subset \Lambda_{\mathrm{FINE}}$, 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 $\Lambda \subset \Lambda_{\mathrm{FINE}}$, 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:

(P1) $\big[[\mathbf{x}] \bmod \Lambda + \mathbf{y}\big] \bmod \Lambda = [\mathbf{x} + \mathbf{y}] \bmod \Lambda, \ \forall \ \mathbf{x} \in \mathbb{R}^n, \ \mathbf{y} \in \mathbb{R}^n$

(P2) $\big[k([\mathbf{x}] \bmod \Lambda)\big] \bmod \Lambda = [k\mathbf{x}] \bmod \Lambda, \ \forall \ k \in \mathbb{Z}, \ \mathbf{x} \in \mathbb{R}^n$

(P3) $\gamma([\mathbf{x}] \bmod \Lambda) = [\gamma\mathbf{x}] \bmod \gamma\Lambda, \ \forall \ \gamma \in \mathbb{R}, \ \mathbf{x} \in \mathbb{R}^n. \tag{9}$

B. System Model

We consider the communication system shown in Figure 1. Two transmitters A and B communicate with a destination D with the help of a common relay R. Transmitter A, and Transmitter 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 sum rate $R_{\mathrm{sum}} = R_a + R_b = 2R$. 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.
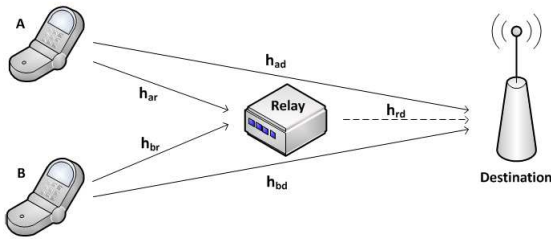

Fig. 1. Multiple-access channel with a half-duplex relay

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 at the 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 the 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, 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 the 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 $\mathbf{H} = [\mathbf{h}_d^T; \mathbf{h}_r^T] \in \mathbb{R}^{2 \times 2}$. Also, we will find it useful to sometimes use the notation $\boldsymbol{\beta}_{\boldsymbol{s}}$ to denote the vector composed of the first two components of vector $\boldsymbol{\beta}$, i.e., $\boldsymbol{\beta}_{\boldsymbol{s}} = [\beta_a, \beta_b]^T$ – 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. Sum 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 $\gamma = \sqrt{\beta_r^2 \mathrm{snr}/(1 + \mathrm{snr} \|\boldsymbol{\beta}_{\boldsymbol{s}} \circ \mathbf{h}_r\|^2)}$. 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

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

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 rather straightforwardly [13] that the following

sum rate is achievable with the relay operating on the amplify-and-forward mode,

$$R_{\text{sum}}^{\text{AF}} = \max \frac{1}{2} \min \left\{ \log \left(\det \left(\mathbf{I}_2 + \beta_a^2 \text{snr}(\mathbf{h}_a \mathbf{h}_a^T) \right) \right), \right.$$
$$\log \left(\det \left(\mathbf{I}_2 + \beta_b^2 \text{snr}(\mathbf{h}_b \mathbf{h}_b^T) \right) \right),$$
$$\left. \frac{1}{2} \log \left(\det \left(\mathbf{I}_2 + \beta_a^2 \text{snr}(\mathbf{h}_a \mathbf{h}_a^T) + \beta_b^2 \text{snr}(\mathbf{h}_b \mathbf{h}_b^T) \right) \right) \right\}. \quad (14)$$

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]

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

where the maximization is over $\boldsymbol{\beta}$. Under the constraint of *symmetric-rate*, it can be shown rather straightforwardly [13] that the following sum rate is achievable with the relay operating on the decode-and-forward mode,

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

where

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

3) Compress-and-Forward: At the end of the first transmission period, the relay quantizes the received $\mathbf{y}_r$ using Wyner-Ziv compression [14], 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], [20],

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

where

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

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,

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

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

$$R_{\text{sum}}^{\text{CF}} = \max \frac{1}{2} \min \left\{ \log \left(1 + \text{snr}|h_{ad}|^2 \beta_a^2 + \frac{\text{snr}|h_{ar}|^2 \beta_a^2}{1 + D} \right), \right.$$
$$\left. \log \left(1 + \text{snr}|h_{bd}|^2 \beta_b^2 + \frac{\text{snr}|h_{br}|^2 \beta_b^2}{1 + D} \right), \frac{1}{2} R^{\text{CF}} \right\} \quad (21)$$

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 Wyner-Ziv compression [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 sum 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 sum rate is achievable for the multiaccess relay model that we study:

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

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 $|\det(\mathbf{k}, \mathbf{t})| \geq 1$.

As we already indicated, in the coding scheme that we use for the proof 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 obtain two equations that can be solved for the transmitted codewords.

Proof: In what follows, we first describe the transmission scheme, the encoding procedures at the transmitters and the relay, and the decoding procedures at the relay and the destination. Then, we analyze the allowed sum rate.

Let Λ be an n-dimensional lattice of second moment $\sigma_\Lambda^2 = P$ and normalized second moment $G(\Lambda)$. Let $\mathcal{V}$ be its fundamental Voronoi region. Also, let $\Lambda_{\text{FINE}} \supseteq \Lambda$ be chosen such that the codebook $\mathcal{C} = \Lambda_{\text{FINE}} \cap \mathcal{V}$ be of cardinality 2^{2nR} [21]. Also, let $\mathbf{k} = [k_a, k_b] \in \mathbb{Z}^2$ and $\mathbf{t} = [t_a, t_b] \in \mathbb{Z}^2$ be given such that $|\det(\mathbf{k}, \mathbf{t})| = |k_a t_b - k_b t_a| \geq 1$.

The encoding and transmission scheme is as follows.

Encoding: 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

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

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

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

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.

As will be shown shortly the relay decodes correctly an integer combination $\mathbf{e}_2 = k_a \mathbf{v}_a + k_b \mathbf{v}_b$ from what it receives during the first transmission period. It then sends

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

during the second transmission period, where the scalar β_r is chosen so as to adjust its transmitted power during this period.

Decoding: During the first transmission period, the relay receives

$$\mathbf{y}_r \quad = \quad h_{ar} \mathbf{x}_a + h_{br} \mathbf{x}_b + \mathbf{z}_r. \qquad (26)$$

Next, the relay performs the following modulo-reduction operation on the received signal:

$$\mathbf{y}'_r = \left[\alpha_r \mathbf{y}_r + k_a \mathbf{u}_a + k_b \mathbf{u}_b \right] \bmod \Lambda$$

$$= \left[\alpha_r (h_{ar} \mathbf{x}_a + h_{br} \mathbf{x}_b + \mathbf{z}_r) + \frac{k_a}{\beta_a} \mathbf{x}_a - \frac{k_a}{\beta_a} \mathbf{x}_a + \frac{k_b}{\beta_b} \mathbf{x}_b - \frac{k_b}{\beta_b} \mathbf{x}_b + \right.$$

$$\left. k_a \mathbf{u}_a + k_b \mathbf{u}_b \right] \bmod \Lambda$$

$$\overset{(a)}{=} \left[(\alpha_r h_{ar} - \frac{k_a}{\beta_a}) \mathbf{x}_a + (\alpha_r h_{br} - \frac{k_b}{\beta_b}) \mathbf{x}_b + k_a [\mathbf{v}_a - \mathbf{u}_a] \bmod \Lambda + \right.$$

$$\left. k_b [\mathbf{v}_b - \mathbf{u}_b] \bmod \Lambda + \alpha_r \mathbf{z}_r + k_a \mathbf{u}_a + k_b \mathbf{u}_b \right] \bmod \Lambda$$

$$\overset{(b)}{=} \left[k_a \mathbf{v}_a + k_b \mathbf{v}_b + (\alpha_r h_{ar} - \frac{k_a}{\beta_a}) \mathbf{x}_a + (\alpha_r h_{br} - \frac{k_b}{\beta_b}) \mathbf{x}_b + \right.$$

$$\left. \alpha_r \mathbf{z}_r \right] \bmod \Lambda$$

$$\overset{(c)}{=} \left[k_a \mathbf{v}_a + k_b \mathbf{v}_b + \mathbf{z}'_r \right] \bmod \Lambda \qquad (27)$$

where (a) follows by substituting $\mathbf{x}_a$ and $\mathbf{x}_b$ using (23) and (24); (b) follows by using the properties (P1) and (P2) in (9); and (c) follows by substituting,

$$\mathbf{z}'_r \triangleq \left[\alpha_r \mathbf{z}_r + (\alpha_r h_{ar} - \frac{k_a}{\beta_a}) \mathbf{x}_a + (\alpha_r h_{br} - \frac{k_b}{\beta_b}) \mathbf{x}_b \right] \bmod \Lambda. \qquad (28)$$

The parameter $\alpha_r \in \mathbb{R}$ is some inflation factor whose optimal value will be specified below and $\mathbf{z}'_r$ is the *effective* noise at the relay. Since an integer combination of lattice points is a lattice point, $\mathbf{e}_2 = [k_a \mathbf{v}_a + k_b \mathbf{v}_b] \in \Lambda$ and, thus, the equivalent channel $\mathbf{y}'_r$ is a modulo-lattice additive noise (MLAN) channel [22] with noise equal to $\mathbf{z}'_r$. By decoding the lattice point $\mathbf{e}_2 \in \Lambda$ from the MLAN channel $\mathbf{y}'_r$, the relay obtains an integer combination of the users' codewords, $\mathbf{e}_2 = [k_a \mathbf{v}_a + k_b \mathbf{v}_b]$.

During the first transmission period, the destination receives

$$\mathbf{y}_d = h_{ad} \mathbf{x}_a + h_{bd} \mathbf{x}_b + \mathbf{z}_d. \qquad (29)$$

Similar to the relay, the destination computes a linear combination with integer coefficients of the transmitters' codewords by performing the modulo-reduction operation:

$$\mathbf{y}'_d = \left[\alpha_d \mathbf{y}_d + t_a \mathbf{u}_a + t_b \mathbf{u}_b \right] \bmod \Lambda$$

$$= \left[t_a \mathbf{v}_a + t_b \mathbf{v}_b + \mathbf{z}'_d \right] \bmod \Lambda \qquad (30)$$

where $\alpha_d \in \mathbb{R}$ is some inflation factor and $\mathbf{z}'_d$ is the effective noise at the destination, given by

$$\mathbf{z}'_d \triangleq \left[\alpha_d \mathbf{z}_d + (\alpha_d h_{ad} - \frac{t_a}{\beta_a}) \mathbf{x}_a + (\alpha_d h_{bd} - \frac{t_b}{\beta_b}) \mathbf{x}_b \right] \bmod \Lambda. \qquad (31)$$

Thus, during the first transmission period, by using the MLAN channel $\mathbf{y}'_d$, the destination obtains a first integer combination $\mathbf{e}_1 = [t_a \mathbf{v}_a + t_b \mathbf{v}_b] \in \Lambda$ of the users' codewords.

During the second transmission period, the destination receives from the relay the signal,

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

$$= h_{rd} \beta_r \left([k_a \mathbf{v}_a + k_b \mathbf{v}_b - \mathbf{u}_r] \bmod \Lambda \right) + \tilde{\mathbf{z}}_d. \qquad (32)$$

Again, by performing a modulo-reduction operation on the obtained signal, the destination gets

$$\tilde{\mathbf{y}}'_d = \left[\tilde{\alpha}_d \tilde{\mathbf{y}}_d + \mathbf{u}_r \right] \bmod \Lambda$$

$$= \left[k_a \mathbf{v}_a + k_b \mathbf{v}_b + \tilde{\mathbf{z}}'_d \right] \bmod \Lambda, \qquad (33)$$

where $\tilde{\alpha}_d \in \mathbb{R}$ is some inflation factor whose optimal value will be specified below and $\tilde{\mathbf{z}}'_d$ is the effective noise at the destination, given by

$$\tilde{\mathbf{z}}'_d \triangleq \left[\tilde{\alpha}_d \tilde{\mathbf{z}}_d + (\tilde{\alpha}_d h_{rd} - \frac{1}{\beta_r}) \tilde{\mathbf{x}}_r \right] \bmod \Lambda. \qquad (34)$$

Thus, during the second transmission period, the destination obtains a second linear combination $\mathbf{e}_2 = [k_a \mathbf{v}_a + t_b \mathbf{v}_b] \in \Lambda$ that relates the users' codewords. This combination has been computed and forwarded by the relay.

Summarizing, over the entire transmission time, the destination collects two linear combinations with integer coefficients that relate the users' codewords, as

$$\left[\begin{array}{c} \mathbf{e}_1 \\ \mathbf{e}_2 \end{array} \right] = \left[\begin{array}{cc} t_a & t_b \\ k_a & k_b \end{array} \right] \left[\begin{array}{c} \mathbf{v}_a \\ \mathbf{v}_b \end{array} \right]. \qquad (35)$$

Now, since the integer-valued matrix in (35) 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 (35).

The destination is able to recover the messages $\hat{W}_a$ and $\hat{W}_b$ reliably only if the message rate is less than or equal to the computational rate [13].

Rate Analysis:

Step 1) First, we find the rate at which the relay decodes reliably the combination $\mathbf{e}_2 = [k_a \mathbf{v}_a + t_b \mathbf{v}_b] \in \Lambda$. Consider the channel from $\mathbf{e}_2$ to $\mathbf{y}'_r$ as given by (27). Due to the dither $\mathbf{u}_a$, the input $\mathbf{x}_a$ of Transmitter A is independent of $\mathbf{v}_a = \phi_a(W_a) \in \mathcal{C}$, and is uniformly distributed over $\mathcal{V}$ (see, e.g., [23], [24]). Similarly, due to the dither $\mathbf{u}_b$, the input $\mathbf{x}_b$ of Transmitter B is independent of $\mathbf{v}_b = \phi_b(W_b) \in \mathcal{C}$, and is uniformly distributed over $\mathcal{V}$. The effective noise $\mathbf{z}'_r$ is then independent of $\mathbf{v}_a$ and $\mathbf{v}_b$, and so of $\mathbf{e}_2 = k_a \mathbf{v}_a + $

IEEE Transactions on Wireless Communications

$k_b \mathbf{v}_b$. It is composed of a "self noise" component and a Gaussian noise component; and its average power satisfies

$$
\begin{aligned}
\mathrm{Var}\,(\mathbf{z}'_r) &\le \mathrm{Var}\Big(\alpha_r \mathbf{z}_r + (\alpha_r h_{ar} - \frac{k_a}{\beta_a})\mathbf{x}_a + (\alpha_r h_{br} - \frac{k_b}{\beta_b})\mathbf{x}_b\Big) \\
&= \alpha_r^2 N + P(\alpha_r h_{ar}\beta_a - k_a)^2 + P(\alpha_r h_{br}\beta_b - k_b)^2.
\end{aligned}
\tag{36}
$$

where the inequality follows since the modulo-reduction operation can only reduce the entropy.

The rate (per channel use) at which the relay decodes the linear combination with integer coefficients $\mathbf{e}_2 = k_a \mathbf{v}_a + k_b \mathbf{v}_b$ is given by [25],

$$
\begin{aligned}
\frac{1}{2n}I(\mathbf{e}_2; \mathbf{y}'_r) &= \frac{1}{2n}h(\mathbf{y}'_r) - \frac{1}{2n}h(\mathbf{y}'_r|\mathbf{e}_2) \\
&\overset{(d)}{=} \frac{1}{2n}h(\mathbf{y}'_r) - \frac{1}{2n}h(\mathbf{z}'_r) \\
&\overset{(e)}{=} \frac{1}{2n}\log^{+} \mathrm{Vol}\,(\mathcal{V}) - \frac{1}{2n}h(\mathbf{z}'_r) \\
&\overset{(f)}{\ge} \frac{1}{4}\log^{+}\left(\frac{P}{\alpha_r^2 N + P(\alpha_r h_{ar}\beta_a - k_a)^2 + P(\alpha_r h_{br}\beta_b - k_b)^2}\right) \\
&\quad - \frac{1}{4}\log\left(2\pi e G(\Lambda)\right) \\
&\overset{(g)}{\ge} \frac{1}{4}\log^{+}\left(k_a^2 + k_b^2 - \frac{P(h_{ar}\beta_a k_a + h_{br}\beta_b k_b)^2}{N + P(|h_{ar}|^2\beta_a^2 + |h_{br}|^2\beta_b^2)}\right)^{-1} \\
&\quad - \frac{1}{4}\log\left(2\pi e G(\Lambda)\right) \\
&= \frac{1}{4}\log^{+}\left(\left(\|\mathbf{k}\|^2 - \frac{P((\boldsymbol{\beta_s} \circ \mathbf{h}_r)^T \mathbf{k})^2}{N + P\|\boldsymbol{\beta_s} \circ \mathbf{h}_r\|^2}\right)^{-1}\right) - \frac{1}{4}\log\left(2\pi e G(\Lambda)\right)
\end{aligned}
\tag{37}
$$

where (d) follows from the fact that $\mathbf{z}'_r$ is independent of $\mathbf{e}_2$; (e) follows from the fact that $\mathbf{y}'_r$ is uniformly distributed over $\mathcal{V}$; (f) follows from the fact that the entropy of $\mathbf{z}'_r$ is upper-bounded by that of an i.i.d Gaussian vector that has the same second moments and by using (36); and (g) follows through straightforward algebra and by noticing that the right hand side (RHS) of (36) is maximized with the value of α_r set to

$$
\alpha_r^{\star} = \frac{P(\boldsymbol{\beta_s} \circ \mathbf{h}_r)^T \mathbf{k}}{N + P\|\boldsymbol{\beta_s} \circ \mathbf{h}_r\|^2}.
\tag{38}
$$

Step 2) Using analysis and algebra that are essentially similar to those in Step 1) above, it can be shown that the destination decodes the linear combination $\mathbf{e}_1 = t_a \mathbf{v}_a + t_b \mathbf{v}_b$ from what it gets directly from the direct links at rate

$$
\frac{1}{2n}I(\mathbf{e}_1; \mathbf{y}'_d) \ge \frac{1}{4}\log^{+}\left(\left(\|\mathbf{t}\|^2 - \frac{P((\boldsymbol{\beta_s} \circ \mathbf{h}_r)^T \mathbf{t})^2}{N + P\|\boldsymbol{\beta_s} \circ \mathbf{h}_r\|^2}\right)^{-1}\right) - \frac{1}{4}\log\left(2\pi e G(\Lambda)\right).
\tag{39}
$$

Similarly, it can be easily shown that the destination decodes the linear combination that is computed at the relay and forwarded to the destination during the second transmission period at rate

$$
\frac{1}{2n}I(\mathbf{e}_2; \tilde{\mathbf{y}}'_d) \ge \frac{1}{4}\log\left(1 + \frac{P|h_{rd}|^2\beta_r^2}{N}\right) - \frac{1}{4}\log\left(2\pi e G(\Lambda)\right).
\tag{40}
$$

Let us define $R_{\mathrm{sr}}(\Lambda)$ as the RHS of (37), $R_{\mathrm{sd}}(\Lambda)$ as the RHS of (39), and $R_{\mathrm{rd}}(\Lambda)$ as the RHS of (40).

The above means that using the coding scheme that we described, which employs the n-dimensional lattice Λ, the destination can decode the transmitters' codewords correctly at a transmission rate

equal to the minimum among $R_{\mathrm{sr}}(\Lambda)$, $R_{\mathrm{sd}}(\Lambda)$ and $R_{\mathrm{rd}}(\Lambda)$, i.e., $R(\Lambda) = \min\{R_{\mathrm{sr}}(\Lambda), R_{\mathrm{sd}}(\Lambda), R_{\mathrm{rd}}(\Lambda)\}$. The allowed sum rate is given by $R_{\mathrm{sum}}^{\mathrm{CoF}}(\Lambda) = 2R(\Lambda) = 2\min\{R_{\mathrm{sr}}(\Lambda), R_{\mathrm{sd}}(\Lambda), R_{\mathrm{rd}}(\Lambda)\}$. Noticing that $2\pi e G(\Lambda) \to 1$ when $n \to \infty$ [10], the desired sum rate (22) is obtained by taking the limit of $R_{\mathrm{sum}}^{\mathrm{CoF}}(\Lambda)$ as n goes to infinity; and this completes the proof of Proposition 1. □

Although the achievable sum 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 sum 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 sum rate is achievable:

$$
R_{\mathrm{sum}}^{\mathrm{CoD}} = \max \frac{1}{2}\min\left\{\log^{+}\left(\frac{\mathrm{snr}}{\mathrm{snr}\|\boldsymbol{\beta_s} \circ \mathbf{H}^T \boldsymbol{\alpha}_t - \mathbf{t}\|^2 + (\boldsymbol{\alpha}_t \circ \boldsymbol{\alpha}_t)^T \mathbf{n}_d}\right),\right.
$$
$$
\left.\log^{+}\left(\frac{\mathrm{snr}}{\mathrm{snr}\|\boldsymbol{\beta_s} \circ \mathbf{H}^T \boldsymbol{\alpha}_k - \mathbf{k}\|^2 + (\boldsymbol{\alpha}_k \circ \boldsymbol{\alpha}_k)^T \mathbf{n}_d}\right)\right\},
\tag{41}
$$

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, $\mathbf{n}_d = [1, 1 + D/N]^T \in \mathbb{R}^2$, and D is given by

$$
D = \frac{N^2\left(1 + \mathrm{snr}\|\boldsymbol{\beta_s} \circ \mathbf{h}_r\|^2\right)}{|h_{rd}|^2 P_r} - \frac{N^2\left(\mathrm{snr}(\boldsymbol{\beta_s} \circ \mathbf{h}_r)^T(\boldsymbol{\beta_s} \circ \mathbf{h}_d)\right)^2}{|h_{rd}|^2 P_r\left(1 + \mathrm{snr}\|\boldsymbol{\beta_s} \circ \mathbf{h}_d\|^2\right)},
\tag{42}
$$

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 $|\det(\mathbf{k}, \mathbf{t})| \ge 1$.

As we indicated previously, 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 relates the users' codewords.

Proof: The transmission scheme, and the encoding procedures at the transmitters are similar to those in the proof of Proposition 1. Therefore they will be outlined only, for brevity. We will insist more on aspects of the coding scheme that are inherently different from those of the coding scheme of Proposition 1.

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).

The relay quantizes what it receives during the first transmission period using Wyner-Ziv compression [16], accounting for the available side information $\mathbf{y}_d$ at the destination. Let $\hat{\mathbf{y}}_r$ be the compressed version of $\mathbf{y}_r$ given by

$$\hat{\mathbf{y}}_r = \mathbf{y}_r + \mathbf{d} \qquad (43)$$

where $\mathbf{d}$ is a Gaussian random vector whose elements are i.i.d with zero mean and variance D; and is independent of all other signals. Also let $\hat{R}_{\mathrm{WZ}}$ be the resulting compression rate. During the second transmission period, the relay conveys the description $\hat{\mathbf{y}}_r$ of $\mathbf{y}_r$ to the destination. To this end, it sends an independent Gaussian input $\tilde{\mathbf{x}}_r$ with power $\beta_r^2 P$ and carries the Wyner-Ziv compression index of $\hat{\mathbf{y}}_r$.

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

$$
\begin{aligned}
\mathbf{y}_d &= h_{ad}\mathbf{x}_a + h_{bd}\mathbf{x}_b + \mathbf{z}_d \\
\tilde{\mathbf{y}}_d &= h_{rd}\tilde{\mathbf{x}}_r + \tilde{\mathbf{z}}_d.
\end{aligned}
\qquad (44)
$$

The destination computes two linearly independent combinations with integer coefficients that relate the users' codewords, as follows.

Step 1) 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 $\hat{\mathbf{y}}_r$ of $\mathbf{y}_r$ if the constraint (54) below is satisfied (see the "Rate Analysis" section).
Next, the destination combines $\mathbf{y}_d$ and $\hat{\mathbf{y}}_r$ and uses the obtained signal to compute a linear combination with integer coefficients of the users' codewords. More specifically, let

$$
\begin{aligned}
\mathbf{y}_t &= \alpha_{1t}\mathbf{y}_d + \alpha_{2t}\hat{\mathbf{y}}_r \\
&= (\alpha_{1t}h_{ad} + \alpha_{2t}h_{ar})\mathbf{x}_a + (\alpha_{1t}h_{bd} + \alpha_{2t}h_{br})\mathbf{x}_b + \\
&\quad \alpha_{1t}\mathbf{z}_d + \alpha_{2t}\mathbf{z}_r + \alpha_{2t}\mathbf{d},
\end{aligned}
\qquad (45)
$$

for some $\boldsymbol{\alpha}_t = [\alpha_{1t}, \alpha_{2t}]^T \in \mathbb{R}^2$. The destination uses the obtained signal $\mathbf{y}_t$ to compute a linear combination with integer coefficients of the transmitters' codewords by performing the modulo reduction operation

$$
\begin{aligned}
\mathbf{y}_t' &= [\mathbf{y}_t + t_a\mathbf{u}_a + t_b\mathbf{u}_b] \bmod \Lambda \\
&= [t_a\mathbf{v}_a + t_b\mathbf{v}_b + \mathbf{z}_t'] \bmod \Lambda
\end{aligned}
\qquad (46)
$$

where the algebra follows (27) and $\mathbf{z}_t'$ is the effective noise given by

$$
\mathbf{z}_t' \triangleq \Big[\alpha_{1t}\mathbf{z}_d + \alpha_{2t}\mathbf{z}_r + \alpha_{2t}\mathbf{d} + (\alpha_{1t}h_{ad} + \alpha_{2t}h_{ar} - \frac{t_a}{\beta_a})\mathbf{x}_a + (\alpha_{1t}h_{bd} + \alpha_{2t}h_{br} - \frac{t_b}{\beta_b})\mathbf{x}_b\Big] \bmod \Lambda.
\qquad (47)
$$

Finally, by decoding the lattice point $\mathbf{e}_1 = [t_a\mathbf{v}_a + t_b\mathbf{v}_b] \in \Lambda$ using the MLAN channel $\mathbf{y}_t'$, the destination obtains a first linear combination with integer coefficients of the users' codewords.

Step 2 Second, the destination again combines $\mathbf{y}_d$ and $\hat{\mathbf{y}}_r$ and uses the obtained signal to compute a second linear combination with integer coefficients of the users' codewords which is different from the one decoded in *Step 1*. More specifically, let

$$
\begin{aligned}
\mathbf{y}_k &= \alpha_{1k}\mathbf{y}_d + \alpha_{2k}\hat{\mathbf{y}}_r \\
&= (\alpha_{1k}h_{ad} + \alpha_{2k}h_{ar})\mathbf{x}_a + (\alpha_{1k}h_{bd} + \alpha_{2k}h_{br})\mathbf{x}_b + \\
&\quad \alpha_{1k}\mathbf{z}_d + \alpha_{2k}\mathbf{z}_r + \alpha_{2k}\mathbf{d},
\end{aligned}
\qquad (48)
$$

for some $\boldsymbol{\alpha}_k = [\alpha_{1k}, \alpha_{2k}]^T \in \mathbb{R}^2$. The destination uses the obtained signal $\mathbf{y}_k$ to compute a linear combination with integer coefficients

of the transmitters' codewords by performing the modulo reduction operation

$$
\begin{aligned}
\mathbf{y}_k' &= [\mathbf{y}_k + k_a\mathbf{u}_a + k_b\mathbf{u}_b] \bmod \Lambda \\
&= [k_a\mathbf{v}_a + k_b\mathbf{v}_b + \mathbf{z}_k'] \bmod \Lambda
\end{aligned}
\qquad (49)
$$

where the algebra follows (27) and $\mathbf{z}_k'$ is the effective noise given by

$$
\mathbf{z}_k' \triangleq \Big[\alpha_{1k}\mathbf{z}_d + \alpha_{2k}\mathbf{z}_r + \alpha_{2k}\mathbf{d} + (\alpha_{1k}h_{ad} + \alpha_{2k}h_{ar} - \frac{k_a}{\beta_a})\mathbf{x}_a + (\alpha_{1k}h_{bd} + \alpha_{2k}h_{br} - \frac{k_b}{\beta_b})\mathbf{x}_b\Big] \bmod \Lambda.
\qquad (50)
$$

Finally, by decoding the lattice point $\mathbf{e}_2 = [k_a\mathbf{v}_a + k_b\mathbf{v}_b] \in \Lambda$ using the MLAN channel $\mathbf{y}_k'$, the destination obtains a second linear combination with integer coefficients of the users' codewords.

Rate Analysis:

The relay compresses its output $\mathbf{y}_r$ at the per-channel use Wyner-Ziv quantization rate

$$
\begin{aligned}
\hat{R}_{\mathrm{WZ}} &= \frac{1}{2n}I(\mathbf{y}_r; \hat{\mathbf{y}}_r | \mathbf{y}_d) \\
&= \frac{1}{2n}h(\hat{\mathbf{y}}_r | \mathbf{y}_d) - \frac{1}{2n}h(\hat{\mathbf{y}}_r | \mathbf{y}_r, \mathbf{y}_d) \\
&\overset{(a)}{=} \frac{1}{2n}h(\hat{\mathbf{y}}_r | \mathbf{y}_d) - \frac{1}{2n}h(\mathbf{d}) \\
&\overset{(b)}{=} \frac{1}{4n}\log(2\pi e)^n \Big| \mathbb{E}[\hat{\mathbf{y}}_r\hat{\mathbf{y}}_r^T] - \mathbb{E}[\hat{\mathbf{y}}_r]\mathbb{E}[\hat{\mathbf{y}}_r | \mathbf{y}_d]^T] \Big| - \frac{1}{4}\log(2\pi e D) \\
&\overset{(c)}{=} \frac{1}{4}\log\left(1 + \frac{N + P\|\boldsymbol{\beta}_s \circ \mathbf{h}_r\|^2}{D} - \frac{[P(\boldsymbol{\beta}_s \circ \mathbf{h}_r)^T(\boldsymbol{\beta}_s \circ \mathbf{h}_d)]^2}{D(N + P\|\boldsymbol{\beta}_s \circ \mathbf{h}_d\|^2)}\right)
\end{aligned}
\qquad (51)
$$

where (a) follows since $\mathbf{d}$ is independent of $(\mathbf{y}_r, \mathbf{y}_d)$; (b) follows since $(\hat{\mathbf{y}}_r, \mathbf{y}_d)$ are jointly Gaussian and $\mathbf{d}$ is a Gaussian vector; (c) follows through by straightforward algebra, and by noticing that the minimum mean square error (MMSE) estimator of $\hat{\mathbf{y}}_r$ given $\mathbf{y}_d$ is given by

$$
\mathbb{E}[\hat{\mathbf{y}}_r | \mathbf{y}_d] = \frac{P(\boldsymbol{\beta}_s \circ \mathbf{h}_r)^T(\boldsymbol{\beta}_s \circ \mathbf{h}_d)}{N + P\|\boldsymbol{\beta}_s \circ \mathbf{h}_d\|^2} \mathbf{y}_d.
\qquad (52)
$$

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

$$
\hat{R}_{\mathrm{WZ}} \leq \frac{1}{4}\log\left(1 + \frac{P|h_{rd}|^2\beta_r^2}{N}\right).
\qquad (53)
$$

From (51) and (53), we get the following constraint on the distortion

$$
D \geq \frac{N^2\left(1 + \mathrm{snr}\|\boldsymbol{\beta_s} \circ \mathbf{h}_r\|^2\right)}{|h_{rd}|^2 P_r} - \frac{N^2\left(\mathrm{snr}(\boldsymbol{\beta_s} \circ \mathbf{h}_r)^T(\boldsymbol{\beta_s} \circ \mathbf{h}_d)\right)^2}{|h_{rd}|^2 P_r\left(1 + \mathrm{snr}\|\boldsymbol{\beta_s} \circ \mathbf{h}_d\|^2\right)}.
\qquad (54)
$$

The above implies that, under the constraint (54), the destination recovers the lossy version $\hat{\mathbf{y}}_r$ of what it was sent by the relay during the second transmission period.

The destination processes $\mathbf{y}_t$ and $\mathbf{y}_k$ to obtain the linear combinations $\mathbf{e}_1 = [t_a\mathbf{v}_a + t_b\mathbf{v}_b]$, and $\mathbf{e}_2 = [k_a\mathbf{v}_a + k_b\mathbf{v}_b]$ of the users' codewords. Using the MLAN channel $\mathbf{y}_t'$ given by (46) and proceeding similar to above, we get that the destination obtains the desired combination $\mathbf{t}$ at mutual information satisfying

$$
\frac{1}{2n}I(\mathbf{e}_1; \mathbf{y}_t') \geq \frac{1}{4}\log^+\left(\frac{\mathrm{snr}}{\mathrm{snr}\|\boldsymbol{\beta}_s \circ \mathbf{H}^T\boldsymbol{\alpha}_t - \mathbf{t}\|^2 + (\boldsymbol{\alpha}_t \circ \boldsymbol{\alpha}_t)^T\mathbf{n}_d}\right) - \frac{1}{4}\log\left(2\pi e G(\Lambda)\right),
\qquad (55)
$$

where the distortion D satisfies the constraint (54) and $\boldsymbol{\alpha}_t$ should be chosen to minimize the effective noise $\mathbf{z}'_t$ in (47), i.e., such that

$$\boldsymbol{\alpha}_t^\star = \left(\mathbf{G}\mathbf{G}^T + \mathbf{N}_d\right)^{-1}\mathbf{G}\mathbf{t}, \qquad (56)$$

where $\mathbf{G} = \left[(\boldsymbol{\beta}_s \circ \mathbf{h}_d)^T; \ (\boldsymbol{\beta}_s \circ \mathbf{h}_r)^T\right] \in \mathbb{R}^{2\times 2}$ and $\mathbf{N}_d = [1/\text{snr}, 0; 0, \ 1/\text{snr} + D/P] \in \mathbb{R}^{2\times 2}$. In a similar way, we get that the destination obtains the desired combination $\mathbf{k}$ at mutual information satisfying

$$\frac{1}{2n}I(\mathbf{e}_2; \mathbf{y}'_k) \geq \frac{1}{4}\ \log^+\left(\frac{\text{snr}}{\text{snr}\|\boldsymbol{\beta}_s \circ \mathbf{H}^T\boldsymbol{\alpha}_k - \mathbf{k}\|^2 + (\boldsymbol{\alpha}_k \circ \boldsymbol{\alpha}_k)^T\mathbf{n}_d}\right) - \frac{1}{4}\log\left(2\pi e G(\Lambda)\right), \qquad (57)$$

where the distortion D satisfies the constraint (54), and $\boldsymbol{\alpha}_k$ should be chosen to minimize the effective noise $\mathbf{z}'_k$ in (50), i.e., such that

$$\boldsymbol{\alpha}_k^\star = \left(\mathbf{G}\mathbf{G}^T + \mathbf{N}_d\right)^{-1}\mathbf{G}\mathbf{k}. \qquad (58)$$

Let us define $R_1(\Lambda)$ as the RHS of (55) and $R_2(\Lambda)$ as the RHS of (57). The above means that using the lattice-based coding scheme that we described, the destination can decode the transmitters' codewords correctly at the transmission sum rate $R_{\text{sum}}^{\text{CoD}}(\Lambda) = 2\min\{R_1(\Lambda), R_2(\Lambda)\}$ provided that the condition (54) is satisfied. Furthermore, investigating the expression of $R_1(\Lambda)$, it can easily be seen that it decreases with increasing D. Also, observing that the RHS of (54) decreases if β_r increases, the largest rate $R_1(\Lambda)$ is then obtained by taking the equality in the distortion constraint (54) with $\beta_r^2 = P_r/P$. Finally, observing that $2\pi e G(\Lambda) \to 1$ when $n \to \infty$ [10], the desired sum rate (41) is obtained by taking the limit of $R_{\text{sum}}^{\text{CoD}}(\Lambda)$ as n goes to infinity; and this completes the proof of Proposition 2. □

Remark 1: There are some high level similarities among the coding strategies of proposition 1 and proposition 2. In particular, they both consist essentially in decoding two linearly independent equations. However, as we mentioned previously, 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. A direct consequence of all the processing being performed locally at the destination with the latter coding scheme is that both computations of the required equations utilize *all* the output available at the destination, i.e., the output received during the first transmission period as well as the output received during the second transmission period, in a joint manner. Recall that, 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, and 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 joint processing at the destination somehow gives some advantage to the coding strategy of proposition 2. The reader may refer to Section V where this aspect will be illustrated through some numerical examples and discussed further.

IV. SUM RATES OPTIMIZATION

Section IV-A is devoted to find optimal powers and integer-coefficients that maximize the sum 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 sum rate $R_{\text{sum}}^{\text{CoF}}$ as given by (22) in Proposition 1. The optimization problem can be stated as:

$$\text{(A)} \ : \max \frac{1}{2}\min\left\{\log^+\left(\left(\|\mathbf{t}\|^2 - \frac{P((\boldsymbol{\beta}_s \circ \mathbf{h}_d)^T\mathbf{t})^2}{N + P\|\boldsymbol{\beta}_s \circ \mathbf{h}_d\|^2}\right)^{-1}\right),\right.$$
$$\log^+\left(\left(\|\mathbf{k}\|^2 - \frac{P((\boldsymbol{\beta}_s \circ \mathbf{h}_r)^T\mathbf{k})^2}{N + P\|\boldsymbol{\beta}_s \circ \mathbf{h}_r\|^2}\right)^{-1}\right),$$
$$\left.\log\left(1 + \frac{P|h_{rd}|^2\beta_r^2}{N}\right)\right\}, \qquad (59)$$

where the maximization is over $\boldsymbol{\beta}$ such that $0 \leq |\beta_a| \leq \sqrt{P_a/P}$, $0 \leq |\beta_b| \leq \sqrt{P_b/P}$, $0 \leq |\beta_r| \leq \sqrt{P_r/P}$ and over the integer coefficients $\mathbf{k}$ and $\mathbf{t}$ such that $|\det(\mathbf{k}, \mathbf{t})| \geq 1$.

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 $\boldsymbol{\beta}$ and integer coefficients $\mathbf{t}$ and $\mathbf{k}$ alternatively. We note that the allocation of the vector $\boldsymbol{\beta}$ determines the power that each of the transmitters and the relay should use for the transmission. For this reason, we will sometimes refer loosely to the process of selecting the vector $\boldsymbol{\beta}$ appropriately as the power allocation process. Let, with a slight abuse of notation, denote by $R_{\text{sum}}^{\text{CoF}}[\iota]$ the value of the sum rate at some iteration $\iota \geq 0$. To compute $R_{\text{sum}}^{\text{CoF}}$ as given by (59) 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 $R_{\text{sum}}^{\text{CoF}}$ as given by (59)

1: Initialization: set $\iota = 1$ and $\boldsymbol{\beta} = \boldsymbol{\beta}^{(0)}$.
2: Set $\boldsymbol{\beta} = \boldsymbol{\beta}^{(\iota-1)}$ in (59), 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 (59), and solve the obtained problem using Algorithm A-2 given below. Denote by $\boldsymbol{\beta}^{(\iota)}$ the found $\boldsymbol{\beta}$.
4: Increment the iteration index as $\iota = \iota + 1$, and go back to Step 2.
5: Terminate if $\|\boldsymbol{\beta}^{(\iota)} - \boldsymbol{\beta}^{(\iota-1)}\| \leq \epsilon$, $|R_{\text{sum}}^{\text{CoF}}[\iota] - R_{\text{sum}}^{\text{CoF}}[\iota-1]| \leq \epsilon$ or $R_{\text{sum}}^{\text{CoF}}[\iota] \leq R_{\text{sum}}^{\text{CoF}}[\iota-1]$.

As described in Algorithm A, we compute the appropriate preprocessing vector $\boldsymbol{\beta}$ and integer coefficients $\mathbf{k}$ and $\mathbf{t}$, alternatively. More specifically, at iteration $\iota \geq 1$, 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 (59) computed with the choice of the preprocessing vector $\boldsymbol{\beta}$ set to its value obtained from the previous iteration, i.e., $\boldsymbol{\beta} = \boldsymbol{\beta}^{(\iota-1)}$ (for the initialization, set $\boldsymbol{\beta}^{(0)}$ to a default value). As we will show, this sub-problem is a MIQP problem with quadratic constraints; and we solve it iteratively using Algorithm A-1. Next, for the found integer coefficients, the algorithm computes adequate preprocessing vector $\boldsymbol{\beta}^{(\iota)}$ that corresponds to a maximum of (59) 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 geometric programming and successive convex optimization approach (see Algorithm A-2 below). The iterative process in Algorithm A terminates if one of the following two conditions holds: 1) either of $\|\boldsymbol{\beta}^{(\iota)} - \boldsymbol{\beta}^{(\iota-1)}\|$ and $|R_{\text{sum}}^{\text{CoF}}[\iota] - R_{\text{sum}}^{\text{CoF}}[\iota-1]|$ is smaller than a prescribed small strictly positive constant ϵ — in this case, the optimized value of the sum

rate is $R_{\text{sum}}^{\text{CoF}}[\iota]$, and is attained using the preprocessing power vector $\boldsymbol{\beta}^{\star} = \boldsymbol{\beta}^{(\iota)}$ and integer vectors $\mathbf{k}^{\star} = \mathbf{k}^{(\iota)}$ and $\mathbf{t}^{\star} = \mathbf{t}^{(\iota)}$; or 2) the sum rate stops increasing, i.e., $R_{\text{sum}}^{\text{CoF}}[\iota] \leq R_{\text{sum}}^{\text{CoF}}[\iota - 1]$ — in this case, the optimized value of the sum rate is $R_{\text{sum}}^{\text{CoF}}[\iota - 1]$, and is attained using the preprocessing power vector $\boldsymbol{\beta}^{\star} = \boldsymbol{\beta}^{(\iota-1)}$ and integer vectors $\mathbf{k}^{\star} = \mathbf{k}^{(\iota-1)}$ and $\mathbf{t}^{\star} = \mathbf{t}^{(\iota-1)}$.

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 $\boldsymbol{\beta}$. Investigating the objective function in (59), it can easily be seen that this problem can be equivalently stated as

$$\min_{\mathbf{k}, \mathbf{t}, \Delta_1} \quad \Delta_1 \tag{60a}$$

$$\text{s. t.} \quad \Delta_1 \geq \|\mathbf{t}\|^2 - \frac{P((\boldsymbol{\beta}_s \circ \mathbf{h}_d)^T \mathbf{t})^2}{N + P\|\boldsymbol{\beta}_s \circ \mathbf{h}_d\|^2} \tag{60b}$$

$$\Delta_1 \geq \|\mathbf{k}\|^2 - \frac{P((\boldsymbol{\beta}_s \circ \mathbf{h}_r)^T \mathbf{k})^2}{N + P\|\boldsymbol{\beta}_s \circ \mathbf{h}_r\|^2} \tag{60c}$$

$$\Delta_1 \geq \frac{N}{N + \beta_r^2 |h_{rd}|^2 P}, \tag{60d}$$

$$|\det(\mathbf{k}, \mathbf{t})| \geq 1 \tag{60e}$$

$$\mathbf{k} \in \mathbb{Z}^2, \ \mathbf{t} \in \mathbb{Z}^2, \ \Delta_1 \in \mathbb{R}. \tag{60f}$$

Note that Δ_1 is simultaneously an extra optimization variable and the objective function in (60). 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 (59).

In order to reformulate problem (60) in a manner that will be convenient for solving it, we 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 $c_1 = c_2 = 0$, $c_3 = N/(N + \beta_r^2|h_{rd}|^2 P)$ 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

$$\mathbf{F}_1 = \begin{bmatrix} 2(\mathbf{I}_2 - \boldsymbol{\Omega}_1) & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}, \qquad \mathbf{F}_2 = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & 2(\mathbf{I}_2 - \boldsymbol{\Omega}_2) & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}, \tag{61}$$

with

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

$$\boldsymbol{\Omega}_2 := \frac{P}{N + P\|\mathbf{h}_r\|^2}(\boldsymbol{\beta}_s \circ \mathbf{h}_r)(\boldsymbol{\beta}_s \circ \mathbf{h}_r)^T, \tag{62}$$

$$\mathbf{F}_3 = \mathbf{0}, \ \text{and} \ \mathbf{F}_4 = \begin{bmatrix} \mathbf{0} & 0 & -2 & 0 \\ \mathbf{0} & 2 & 0 & 0 \\ \mathbf{0} & 0 & 0 & 0 \end{bmatrix}. \tag{63}$$

The optimization problem (60) can now be reformulated equivalently as

$$\min_{\mathbf{b}} \quad \mathbf{a}_0^T \mathbf{b}$$

$$\text{s. t.} \quad \frac{1}{2}\mathbf{b}^T \mathbf{F}_i \mathbf{b} + \mathbf{a}_i^T \mathbf{b} \leq c_i \quad i = 1, \cdots, 4$$

$$\mathbf{k} \in \mathbb{Z}^2, \ \mathbf{t} \in \mathbb{Z}^2, \ \Delta_1 \in \mathbb{R} \tag{64}$$

The equivalent optimization problem (64) is a MIQP problem with quadratic constraints [26]. 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 [26]. 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 $\mathbf{t}$.

In order to transform the optimization problem (64) into one that is MIQP-compatible (i.e., in which all the quadratic constraints are associated with semi-definite matrices), we replace the quadratic constraint (60e) with one that is linear, as follows. We introduce the following two real-valued vectors $\tilde{\mathbf{k}} = [\tilde{k}_a, \tilde{k}_b]^T \in \mathbb{R}^2$ and $\tilde{\mathbf{t}} = [\tilde{t}_a, \tilde{t}_b]^T \in \mathbb{R}^2$ defined such that they satisfy

$$\mathbf{k} = \boldsymbol{\kappa} \circ \exp(\tilde{\mathbf{k}}), \quad \mathbf{t} = \boldsymbol{\tau} \circ \exp(\tilde{\mathbf{t}}), \tag{65}$$

where $\boldsymbol{\kappa} = [\kappa_a, \kappa_b]^T \in \mathbb{R}^2$ and $\boldsymbol{\tau} = [\tau_a, \tau_b]^T \in \mathbb{R}^2$ are constant vectors to be chosen appropriately. Thus, the constraint (60e) can now be rewritten equivalently as

$$|\det(\mathbf{k}, \mathbf{t})| \geq 1 \ \text{iff} \ |\kappa_a \tau_b \exp(\tilde{k}_a + \tilde{t}_b) - \kappa_b \tau_a \exp(\tilde{k}_b + \tilde{t}_a)| \geq 1. \tag{66}$$

Now, we linearize the constraint (66) by selecting the constant vectors $\boldsymbol{\kappa}$ and $\boldsymbol{\tau}$ such that the first order Taylor series approximations $\exp(\tilde{\mathbf{k}}) \approx \mathbf{1} + \tilde{\mathbf{k}}$ and $\exp(\tilde{\mathbf{t}}) \approx \mathbf{1} + \tilde{\mathbf{t}}$ hold. Hence, the constraint (60e) can be rewritten as

$$|\det(\mathbf{k}, \mathbf{t})| \geq 1 \ \text{iff} \ |\kappa_a \tau_b(1 + \tilde{k}_a + \tilde{t}_b) - \kappa_b \tau_a(1 + \tilde{k}_b + \tilde{t}_a)| \gtrsim 1. \tag{67}$$

Note that the constraint (67) is now linear. The optimization problem (64) has the same solution as the following problem which is MIQP-compatible,

$$\min_{\mathbf{k}, \mathbf{t}, \Delta_1} \quad \Delta_1 \tag{68a}$$

$$\text{s. t.} \quad \|\mathbf{t}\|^2 - \frac{P((\boldsymbol{\beta}_s \circ \mathbf{h}_d)^T \mathbf{t})^2}{N + P\|\boldsymbol{\beta}_s \circ \mathbf{h}_d\|^2} - \Delta_1 \leq 0 \tag{68b}$$

$$\|\mathbf{k}\|^2 - \frac{P((\boldsymbol{\beta}_s \circ \mathbf{h}_r)^T \mathbf{k})^2}{N + P\|\boldsymbol{\beta}_s \circ \mathbf{h}_r\|^2} - \Delta_1 \leq 0 \tag{68c}$$

$$\frac{N}{N + \beta_r^2 |h_{rd}|^2 P} - \Delta_1 \leq 0 \tag{68d}$$

$$-|\kappa_a \tau_b(1 + \tilde{k}_a + \tilde{t}_b) - \kappa_b \tau_a(1 + \tilde{k}_b + \tilde{t}_a)| \lesssim -1 \tag{68e}$$

$$\frac{k_i}{\kappa_i} - 1 - \tilde{k}_i \leq 0, \quad -\frac{k_i}{\kappa_i} + 1 + \tilde{k}_i \leq 0, \quad i = a, b \tag{68f}$$

$$\frac{t_i}{\tau_i} - 1 - \tilde{t}_i \leq 0, \quad -\frac{t_i}{\tau_i} + 1 + \tilde{t}_i \leq 0, \quad i = a, b \tag{68g}$$

$$\mathbf{k}, \ \mathbf{t} \in \mathbb{Z}^2, \ \tilde{\mathbf{k}}, \ \tilde{\mathbf{t}}, \ \boldsymbol{\kappa}, \ \boldsymbol{\tau} \in \mathbb{R}^2, \Delta_1 \in \mathbb{R}. \tag{68h}$$

The optimization problem (68) can be solved iteratively using Algorithm A-1 hereinafter.

3) Power Allocation Policy: Let us now focus on the problem of finding an appropriate preprocessing vector $\boldsymbol{\beta}$ for given integer vectors $\mathbf{k} \in \mathbb{Z}^2$ and $\mathbf{t} \in \mathbb{Z}^2$. Again, investigating the objective function in (59), it can easily be seen that this problem can be

---

Algorithm A-1 Integer coefficients selection for $R_{\mathrm{sum}}^{\mathrm{CoF}}$ as given by (59)

1: Initialization: set $\iota_1 = 1$.
2: Use the branch-and-bound algorithm of [27], [28] to solve for $\Delta_1^{(\iota_1)}$, $\mathbf{k}^{(\iota_1)}$ and $\mathbf{t}^{(\iota_1)}$ with the constraint (68e) substituted with $-\kappa_a \tau_b (1 + \tilde{k}_a + \tilde{t}_b) + \kappa_b \tau_a (1 + \tilde{k}_b + \tilde{t}_a) \leq -1$.
3: Update the values of $\boldsymbol{\kappa}$ and $\boldsymbol{\tau}$ in a way to satisfy (65); and increment the iteration index as $\iota_1 = \iota_1 + 1$.
4: Terminate if $\exp(\tilde{\mathbf{k}}^{(\iota_1)}) \approx \mathbf{1} + \tilde{\mathbf{k}}^{(\iota_1)}$ and $\exp(\tilde{\mathbf{t}}^{(\iota_1)}) \approx \mathbf{1} + \tilde{\mathbf{t}}^{(\iota_1)}$. Denote the found solution as $\Delta_1^{\min,1}$.
5: Redo steps 1 to 4 with in Step 2 the constraint (68e) substituted with $\kappa_a \tau_b (1 + \tilde{k}_b + \tilde{t}_b) - \kappa_b \tau_a (1 + \tilde{k}_b + \tilde{t}_a) \leq -1$. In Step 4, denote the found solution as $\Delta_1^{\min,2}$.
6: Select the integer coefficients corresponding to the minimum among $\Delta_1^{\min,1}$ and $\Delta_1^{\min,2}$.

---

equivalently stated as

$$\min_{\boldsymbol{\beta}, \Delta_2} \quad \Delta_2 \tag{69a}$$

$$\text{s. t.} \quad \Delta_2 \geq \|\mathbf{t}\|^2 - \frac{P((\boldsymbol{\beta}_s \circ \mathbf{h}_d)^T \mathbf{t})^2}{N + P\|\boldsymbol{\beta}_s \circ \mathbf{h}_d\|^2} \tag{69b}$$

$$\Delta_2 \geq \|\mathbf{k}\|^2 - \frac{P((\boldsymbol{\beta}_s \circ \mathbf{h}_r)^T \mathbf{k})^2}{N + P\|\boldsymbol{\beta}_s \circ \mathbf{h}_r\|^2} \tag{69c}$$

$$\Delta_2 \geq \frac{N}{N + P|h_{rd}|^2 \beta_r^2}, \tag{69d}$$

$$-\sqrt{\frac{P_i}{P}} \leq \beta_i \leq \sqrt{\frac{P_i}{P}}, \quad i = a, b, r \tag{69e}$$

$$\boldsymbol{\beta}_s \in \mathbb{R}^2, \ \boldsymbol{\beta} \in \mathbb{R}^3, \ \Delta_2 \in \mathbb{R}. \tag{69f}$$

Here, similarly to the previous section, Δ_2 is simultaneously an extra optimization variable and the objective function in (69). Also, it is easy to see that the value of $\boldsymbol{\beta}$ that achieves the minimum value of Δ_2 also achieves a maximum value of the objective function in (59).

The optimization problem in (69) is non-linear and non-convex. We use geometric programming [17] for solving it. Geometric programming is a special form of convex optimization for which efficient algorithms have been developed and are known in the related literature [29]. 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 [29]

$$\text{minimize} \quad f_0(\boldsymbol{\beta}, \Delta_2) \tag{70a}$$

$$\text{subject to} \quad f_j(\boldsymbol{\beta}, \Delta_2) \leq 1, \quad j = 1, \cdots, J, \tag{70b}$$

$$g_l(\boldsymbol{\beta}, \Delta_2) = 1, \quad l = 1, \cdots, L, \tag{70c}$$

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 $\boldsymbol{\beta}$ and Δ_2. In its standard form, (70) 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 (70) into one that is equivalent and convex. That is, (70) is a GP nonlinear, nonconvex optimization problem that can be transformed into a nonlinear, convex optimization problem.

In our case, in the problem (69), the constraints (69b) and (69c) contain functions that are non posynomial. Also, the variables in (69) are not all positive, thus preventing a direct application of logarithmic transformation. In what follows, we first transform the problem (69) into one equivalent in which the constraints involve functions that are all posynomial and the variables are all positive;

and then we develop an algorithm for solving the equivalent problem.

Let $\mathbf{c} = [c_a, c_b, c_r]^T \in \mathbb{R}^3$ and $\boldsymbol{\delta} = [\delta_a, \delta_b, \delta_r]^T \in \mathbb{R}^3$, such that $c_i > \sqrt{P_i/P}$ and $\delta_i = \beta_i + c_i$ for $i = a, b, r$. Note that the elements of $\boldsymbol{\delta}$ are all strictly positive. Also, for convenience, we define the following functions, for $\mathbf{z} = [z_a, z_b] \in \mathbb{Z}^2$,

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

$$\begin{aligned} \psi_2^i(\boldsymbol{\delta}, \Delta_2, \mathbf{z}) = \ & \Delta_2 \left(N + P|h_{ai}|^2 (\delta_a^2 + c_a^2) + P|h_{bi}|^2 (\delta_b^2 + c_b^2) \right) + \\ & 2P \left(z_a^2 + z_b^2 \right) \left(|h_{ai}|^2 \delta_a c_a + |h_{bi}|^2 \delta_b c_b \right) + \\ & P|h_{ai}|^2 z_a^2 (\delta_a^2 + c_a^2) + P|h_{bi}|^2 z_b^2 (\delta_b^2 + c_b^2) + \\ & 2P h_{ai} h_{bi} z_a z_b (\delta_a \delta_b + c_a c_b). \end{aligned} \tag{71}$$

Let us now define the following functions, $f_1(\boldsymbol{\delta}, \Delta_2) = \psi_1^d(\boldsymbol{\delta}, \Delta_2, \mathbf{t})$, $f_2(\boldsymbol{\delta}, \Delta_2) = \psi_1^r(\boldsymbol{\delta}, \Delta_2, \mathbf{k})$, $g_1(\boldsymbol{\delta}, \Delta_2) = \psi_2^d(\boldsymbol{\delta}, \Delta_2, \mathbf{t})$, $g_2(\boldsymbol{\delta}, \Delta_2) = \psi_2^r(\boldsymbol{\delta}, \Delta_2, \mathbf{k})$, $f_3(\boldsymbol{\delta}, \Delta_2) = N + 2\Delta_2 P|h_{rd}|^2 \delta_r c_r$, and $g_3(\boldsymbol{\delta}, \Delta_2) = \Delta_2 \left(N + P|h_{rd}|^2 (\delta_r^2 + c_r^2) \right)$.

It is now easy to see that the optimization problem (69) can be stated in the following form.

$$\min_{\boldsymbol{\delta}, \Delta_2} \quad \Delta_2 \tag{72a}$$

$$\text{s. t.} \quad \frac{f_1(\boldsymbol{\delta}, \Delta_2)}{g_1(\boldsymbol{\delta}, \Delta_2)} \leq 1, \quad \frac{f_2(\boldsymbol{\delta}, \Delta_2)}{g_2(\boldsymbol{\delta}, \Delta_2)} \leq 1, \quad \frac{f_3(\boldsymbol{\delta}, \Delta_2)}{g_3(\boldsymbol{\delta}, \Delta_2)} \leq 1 \tag{72b}$$

$$-\sqrt{\frac{P_i}{P}} + c_i \leq \delta_i \leq \sqrt{\frac{P_i}{P}} + c_i, \quad i = a, b, r \tag{72c}$$

$$\boldsymbol{\delta} \in \mathbb{R}^3, \ \mathbf{c} \in \mathbb{R}^3, \ \Delta_2 \in \mathbb{R}. \tag{72d}$$

The constraints (72b) 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 [29]. One can transform a Complementary GP problem into a GP problem using series of approximations. In order to get posynomials, we approximate the functions $g_1(\boldsymbol{\delta}, \Delta_2)$, $g_2(\boldsymbol{\delta}, \Delta_2)$ and $g_3(\boldsymbol{\delta}, \Delta_2)$ with monomials, by using the following lemma [17].

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

$$g(\boldsymbol{\delta}, \Delta_2) \geq \tilde{g}(\boldsymbol{\delta}, \Delta_2) = \prod_j \left(\frac{u_j(\boldsymbol{\delta}, \Delta_2)}{\gamma_j} \right)^{\gamma_j}. \tag{73}$$

Here, $\gamma_j = u_j(\boldsymbol{\delta}^{(0)}, \Delta_2^{(0)})/g(\boldsymbol{\delta}^{(0)}, \Delta_2^{(0)})$, $\forall j$, for any fixed positive $\boldsymbol{\delta}^{(0)}$ and $\Delta_2^{(0)}$ then $\tilde{g}(\boldsymbol{\delta}^{(0)}, \Delta_2^{(0)}) = g(\boldsymbol{\delta}^{(0)}, \Delta_2^{(0)})$, and $\tilde{g}(\boldsymbol{\delta}^{(0)}, \Delta_2^{(0)})$ is the best local monomial approximation to $g(\boldsymbol{\delta}^{(0)}, \Delta_2^{(0)})$ near $\boldsymbol{\delta}^{(0)}$ and $\Delta_2^{(0)}$.

Let $\tilde{g}_1(\boldsymbol{\delta}, \Delta_2)$, $\tilde{g}_2(\boldsymbol{\delta}, \Delta_2)$ and $\tilde{g}_3(\boldsymbol{\delta}, \Delta_2)$ be the monomial approximations of the functions $g_1(\boldsymbol{\delta}, \Delta_2)$, $g_2(\boldsymbol{\delta}, \Delta_2)$ and $g_3(\boldsymbol{\delta}, \Delta_2)$ obtained using Lemma 1. Using these monomial approximations, the ratios of posynomials involved in the constraint (72b) can be upper bounded by posynomials. The optimal solution of the problem obtained using the convex approximations is also optimal for the original problem (69), i.e., satisfies the Karush-Kuhn-Tucker

(KKT) conditions of the original problem (69), if the applied approximations satisfy the following three properties [30], [17]:

1) $g_j(\boldsymbol{\delta}, \Delta_2) \leq \tilde{g}_j(\boldsymbol{\delta}, \Delta_2)$ for all $\boldsymbol{\delta}$ and Δ_2 where $\tilde{g}_j(\boldsymbol{\delta}, \Delta_2)$ is the approximation of $g_j(\boldsymbol{\delta}, \Delta_2)$.

2) $g_j(\boldsymbol{\delta}^{(0)}, \Delta_2^{(0)}) = \tilde{g}_j(\boldsymbol{\delta}^{(0)}, \Delta_2^{(0)})$ where $\boldsymbol{\delta}^{(0)}$ and $\Delta_2^{(0)}$ are the optimal solution of the approximated problem in the previous iteration.

3) $\nabla g_j(\boldsymbol{\delta}^{(0)}, \Delta_2^{(0)}) = \nabla \tilde{g}_j(\boldsymbol{\delta}^{(0)}, \Delta_2^{(0)})$, where $\nabla g_j(\cdot)$ stands for the gradient of function $g_j(\cdot)$.

Summarizing, applying the aforementioned transformations, we transformed the original optimization problem (69) first into a Complementary GP problem (72) and then into a GP problem by applying the convex approximations (73). Finally, the obtained GP problem can be solved easily using, e.g., an interior point approach. More specifically, the problem of finding the appropriate preprocessing vector $\boldsymbol{\delta}$ for given integer vectors $\mathbf{k}$ and $\mathbf{t}$ can be solved using Algorithm A-2 hereinafter.

Algorithm A-2 Power allocation policy for $R_{\text{sum}}^{\text{CoF}}$ as given by (59)

1: Set $\boldsymbol{\delta}^{(0)}$ to some initial value. Compute $\Delta_2^{(0)}$ using $\boldsymbol{\delta}^{(0)}$ and set $\iota_2 = 1$.
2: Approximate $g(\boldsymbol{\delta}^{(\iota_2)}, \Delta_2^{(\iota_2)})$ with $\tilde{g}(\boldsymbol{\delta}^{(\iota_2)}, \Delta_2^{(\iota_2)})$ around $\boldsymbol{\delta}^{(\iota_2-1)}$ and $\Delta_2^{(\iota_2-1)}$ using (73).
3: Solve the resulting approximated GP problem using an interior point approach. Denote the found solutions as $\boldsymbol{\delta}^{(\iota_2)}$ and $\Delta_2^{(\iota_2)}$.
4: Increment the iteration index as $\iota_2 = \iota_2 + 1$ and go back to Step 2 using $\boldsymbol{\delta}$ and Δ_2 of step 3.
5: Terminate if $\|\boldsymbol{\delta}^{(\iota_2)} - \boldsymbol{\delta}^{(\iota_2-1)}\| \leq \epsilon$ or if $\Delta_2^{(\iota_2)} \geq \Delta_2^{(\iota_2-1)}$.

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

The algorithms that we develop in this section to solve the optimization problem involved in the maximization of the sum rate given in 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 $R_{\text{sum}}^{\text{CoD}}$ as given by (41) in Proposition 2. The optimization problem can be stated as:

$$\text{(B): } \max \frac{1}{2} \min \left\{ \log^+ \left(\frac{\text{snr}}{\text{snr}\|\boldsymbol{\beta}_s \circ \mathbf{H}^T \boldsymbol{\alpha}_t - \mathbf{t}\|^2 + (\boldsymbol{\alpha}_t \circ \boldsymbol{\alpha}_t)^T \mathbf{n}_d} \right), \right.$$
$$\left. \log^+ \left(\frac{\text{snr}}{\text{snr}\|\boldsymbol{\beta}_s \circ \mathbf{H}^T \boldsymbol{\alpha}_k - \mathbf{k}\|^2 + (\boldsymbol{\alpha}_k \circ \boldsymbol{\alpha}_k)^T \mathbf{n}_d} \right) \right\}, \quad (74)$$

where the distortion D is given by

$$D = \frac{N^2 \left(1 + \text{snr}\|\boldsymbol{\beta}_s \circ \mathbf{h}_r\|^2\right)}{|h_{rd}|^2 P_r} - \frac{N^2 \left(\text{snr}(\boldsymbol{\beta}_s \circ \mathbf{h}_r)^T (\boldsymbol{\beta}_s \circ \mathbf{h}_d)\right)^2}{|h_{rd}|^2 P_r \left(1 + \text{snr}\|\boldsymbol{\beta}_s \circ \mathbf{h}_d\|^2\right)}, \quad (75)$$

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 $|\det(\mathbf{k}, \mathbf{t})| \geq 1$.

In order to compute $R_{\text{sum}}^{\text{CoD}}$ as given by (74), we develop the following iterative algorithm which optimizes the integer coefficients and the powers alternatively, and to which we refer to as "Algorithm B" in reference to the optimization problem (B).

Algorithm B Iterative algorithm for computing $R_{\text{sum}}^{\text{CoD}}$ as given by (74)

1: Choose an initial feasible vector $\boldsymbol{\beta}_s^{(0)}$ and set $\iota = 1$.
2: Solve (74) with $\boldsymbol{\beta}_s = \boldsymbol{\beta}_s^{(\iota-1)}$ 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 (74) with $\mathbf{k} = \mathbf{k}^{(\iota)}$ and $\mathbf{t} = \mathbf{t}^{(\iota)}$ for the optimal $\boldsymbol{\beta}_s$ using Algorithm B-2 and assign it to $\boldsymbol{\beta}_s^{(\iota)}$.
4: Increment the iteration index as $\iota = \iota + 1$ and go back to Step 2.
5: Terminate if $\|\boldsymbol{\beta}_s^{(\iota)} - \boldsymbol{\beta}_s^{(\iota-1)}\| \leq \epsilon$, $|R_{\text{sum}}^{\text{CoD}}[\iota] - R_{\text{sum}}^{\text{CoD}}[\iota-1]| \leq \epsilon$ or if $R_{\text{sum}}^{\text{CoD}}[\iota] \leq R_{\text{sum}}^{\text{CoD}}[\iota-1]$.

---

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

$$\min_{\mathbf{k}, \mathbf{t}, \Theta_1} \quad \Theta_1 \quad (76a)$$
$$\text{s. t.} \quad \Theta_1 \geq \mathbf{t}^T \boldsymbol{\Omega} \mathbf{t} \quad (76b)$$
$$\Theta_1 \geq \mathbf{k}^T \boldsymbol{\Omega} \mathbf{k} \quad (76c)$$
$$\det(\mathbf{k}, \mathbf{t}) = |k_a t_b - k_b t_a| \geq 1 \quad (76d)$$
$$\mathbf{k}, \mathbf{t} \in \mathbb{Z}^2, \ \Theta_1 \in \mathbb{R}, \quad (76e)$$

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 (76) into a MIQP problem with quadratic constraints [26] as before. We 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

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

$$\text{and} \qquad \mathbf{F}_3 = \begin{bmatrix} \mathbf{0} & 0 & -2 & 0 \\ \mathbf{0} & 2 & 0 & 0 \\ \mathbf{0} & 0 & 0 & 0 \end{bmatrix}. \quad (77)$$

The optimization problem (76) can now be reformulated equivalently as

$$\min_{\mathbf{b}} \quad \mathbf{a}_0^T \mathbf{b}$$
$$\text{s. t.} \quad \frac{1}{2} \mathbf{b}^T \mathbf{F}_i \mathbf{b} + \mathbf{a}_i^T \mathbf{b} \leq c_i \quad i = 1, 2, 3$$
$$\mathbf{k} \in \mathbb{Z}^2, \ \mathbf{t} \in \mathbb{Z}^2, \ \Theta_1 \in \mathbb{R} \quad (78)$$

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 $\mathbf{t}$. In order to transform the optimization problem (78) into one that is MIQP-compatible (i.e., in which all the quadratic constraints are associated with semi-definite matrices), we replace the quadratic constraint (76d) with one that is linear, as performed in the previous section.

Finally, the MIQP optimization problem is given by,

$$\min_{\mathbf{k}, \mathbf{t}, \Theta_1} \Theta_1 \tag{79a}$$

$$\text{s. t. } \Theta_1 \geq \mathbf{t}^T \Omega \mathbf{t} \tag{79b}$$

$$\Theta_1 \geq \mathbf{k}^T \Omega \mathbf{k} \tag{79c}$$

$$- \left| \kappa_a \tau_b (1 + \tilde{k}_a + \tilde{t}_b) - \kappa_b \tau_a (1 + \tilde{k}_b + \tilde{t}_a) \right| \lesssim -1 \tag{79d}$$

$$\frac{k_i}{\kappa_i} - 1 - \tilde{k}_i \leq 0, \quad -\frac{k_i}{\kappa_i} + 1 + \tilde{k}_i \leq 0, \quad i = a, b \tag{79e}$$

$$\frac{t_i}{\tau_i} - 1 - \tilde{t}_i \leq 0, \quad -\frac{t_i}{\tau_i} + 1 + \tilde{t}_i \leq 0, \quad i = a, b \tag{79f}$$

$$\mathbf{k}, \mathbf{t} \in \mathbb{Z}^2, \; \tilde{\mathbf{k}}, \tilde{\mathbf{t}}, \boldsymbol{\kappa}, \boldsymbol{\tau} \in \mathbb{R}^2, \Theta_1 \in \mathbb{R}. \tag{79g}$$

The optimization problem (79) can be solved iteratively using Algorithm B-1 hereinafter.

Algorithm B-1 Integer coefficients selection for $R_{\text{sum}}^{\text{CoD}}$ as given by (74)

1: Initialization: set $\iota_1 = 1$.
2: Use the branch-and-bound algorithm of [27], [28] to solve for $\Theta_1^{(\iota_1)}$, $\mathbf{k}^{(\iota_1)}$ and $\mathbf{t}^{(\iota_1)}$ with the constraint (79d) substituted with $-\kappa_a \tau_b (1 + \tilde{k}_a + \tilde{t}_b) + \kappa_b \tau_a (1 + \tilde{k}_b + \tilde{t}_a) \leq -1$.
3: Update the values of $\boldsymbol{\kappa}$ and $\boldsymbol{\tau}$ in a way to satisfy (65); and increment the iteration index as $\iota_1 = \iota_1 + 1$.
4: Terminate if $\exp(\tilde{\mathbf{k}}^{(\iota_1)}) \approx \mathbf{1} + \tilde{\mathbf{k}}^{(\iota_1)}$ and $\exp(\tilde{\mathbf{t}}^{(\iota_1)}) \approx \mathbf{1} + \tilde{\mathbf{t}}^{(\iota_1)}$. Denote the found solution as $\Theta_1^{\min,2}$.
5: Redo steps 1 to 4 with in Step 2 the constraint (79d) substituted with $\kappa_a \tau_b (1 + \tilde{k}_a + \tilde{t}_b) - \kappa_b \tau_a (1 + \tilde{k}_b + \tilde{t}_a) \leq -1$. In Step 4, denote the found solution as $\Theta_1^{\min,2}$.
6: 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,

$$\min_{\boldsymbol{\beta}_{\text{s}}, \boldsymbol{\alpha}_t, \boldsymbol{\alpha}_k, \Theta_2} \Theta_2 \tag{80a}$$

$$\text{s. t.} \Theta_2 \geq \frac{\text{snr} \|\boldsymbol{\beta}_s \circ \mathbf{H}^T \boldsymbol{\alpha}_t - \mathbf{t}\|^2 + (\boldsymbol{\alpha}_t \circ \boldsymbol{\alpha}_t)^T \mathbf{n}_d}{\text{snr}}, \tag{80b}$$

$$\Theta_2 \geq \frac{\text{snr} \|\boldsymbol{\beta}_s \circ \mathbf{H}^T \boldsymbol{\alpha}_k - \mathbf{k}\|^2 + (\boldsymbol{\alpha}_k \circ \boldsymbol{\alpha}_k)^T \mathbf{n}_d}{\text{snr}}, \tag{80c}$$

$$D \geq \frac{N^2 \left(1 + \text{snr} \|\boldsymbol{\beta}_s \circ \mathbf{h}_r\|^2\right)}{|h_{rd}|^2 P_r} -$$
$$\frac{N^2 \left(\text{snr}(\boldsymbol{\beta}_s \circ \mathbf{h}_r)^T (\boldsymbol{\beta}_s \circ \mathbf{h}_d)\right)^2}{|h_{rd}|^2 P_r \left(1 + \text{snr} \|\boldsymbol{\beta}_s \circ \mathbf{h}_d\|^2\right)} \tag{80d}$$

$$-\sqrt{\frac{P_i}{P}} \leq \beta_i \leq \sqrt{\frac{P_i}{P}}, \quad i = a, b \tag{80e}$$

$$\boldsymbol{\beta}_s \in \mathbb{R}^2, \; \Theta_2 \in \mathbb{R}. \tag{80f}$$

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 $i = a, b$. Note that the elements of $\boldsymbol{\delta}$ are all strictly positive. We can reformulate the optimization

---

Algorithm B-2 Power allocation policy for $R_{\text{sum}}^{\text{CoD}}$ as given by (74)

1: Set $\boldsymbol{\delta_s}^{(0,0)}$ 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 $\boldsymbol{\delta_s}^{(\iota_2-1,\iota_3)}$.
3: Approximate $g(\boldsymbol{\delta}^{(\iota_2,\iota_3)}, \Theta_2^{(\iota_2,\iota_3)})$ with $\tilde{g}(\boldsymbol{\delta_s}^{(\iota_2,\iota_3)}, \Theta_2^{(\iota_2,\iota_3)})$ around $\boldsymbol{\delta}^{(\iota_2-1,\iota_3)}$ and $\Theta_2^{(\iota_2-1,\iota_3)}$ using (73).
4: Solve the resulting approximated GP problem using an interior point approach. Denote the found solutions as $\boldsymbol{\delta_s}^{(\iota_2,\iota_3)}$ 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 $\boldsymbol{\delta_s}$ and Θ_2 of step 4.
6: Terminate if $\|\boldsymbol{\delta_s}^{(\iota_2,\iota_3)} - \boldsymbol{\delta_s}^{(\iota_2-1,\iota_3)}\| \leq \epsilon$ or if $\Theta_2^{(\iota_2,\iota_3)} \geq \Theta_2^{(\iota_2-1,\iota_3)}$ and denote by $\boldsymbol{\delta}$ the final value.
7: Increment the iteration index as $\iota_3 = \iota_3 + 1$, set $\iota_2 = 1$, and $\boldsymbol{\delta_s}^{(\iota_2-1,\iota_3)} = \boldsymbol{\delta}$ and then go back to Step 2.
8: Terminate if $|R_{\text{sum}}^{\text{CoD}}[\iota_3] - R_{\text{sum}}^{\text{CoD}}[\iota_3 - 1]| \leq \epsilon$ or if $R_{\text{sum}}^{\text{CoD}}[\iota_3] \leq R_{\text{sum}}^{\text{CoD}}[\iota_3 - 1]$.

---

problem as,

$$\min_{\boldsymbol{\delta_s}, \boldsymbol{\alpha}_t, \boldsymbol{\alpha}_k, \Theta_2} \Theta_2 \tag{81a}$$

$$\text{s. t. } \frac{f_1(\boldsymbol{\delta}, \Theta_2, \boldsymbol{\alpha}_t, \boldsymbol{\alpha}_k)}{g_1(\boldsymbol{\delta}, \Theta_2, \boldsymbol{\alpha}_t, \boldsymbol{\alpha}_k)} \leq 1, \quad \frac{f_2(\boldsymbol{\delta}, \Theta_2, \boldsymbol{\alpha}_t, \boldsymbol{\alpha}_k)}{g_2(\boldsymbol{\delta}, \Theta_2, \boldsymbol{\alpha}_t, \boldsymbol{\alpha}_k)} \leq 1, \tag{81b}$$

$$\frac{f_3(\boldsymbol{\delta}, \Theta_2)}{g_3(\boldsymbol{\delta}, \Theta_2)} \leq 1 \tag{81c}$$

$$-\sqrt{\frac{P_i}{P}} + c_i \leq \delta_i \leq \sqrt{\frac{P_i}{P}} + c_i, \quad i = a, b \tag{81d}$$

$$\boldsymbol{\delta}_s \in \mathbb{R}^2, \; \mathbf{c} \in \mathbb{R}^2, \; \Theta_2 \in \mathbb{R}. \tag{81e}$$

The constraints (81b) correspond to the constraints (80b), (80c), and (80d). These functions consist of ratios of posynomials, i.e., are not posynomial — recall that a ratio of posynomials is in general non posynomial. As before, we transform the Complementary GP problem into a GP problem using series of approximations.

As there are several variables to optimize over simultaneously, the optimization is carried out in two steps. First, we optimize the power value $\boldsymbol{\delta_s}$ using geometric programming with successive convex approximation in a way that is essentially similar to the above, for a fixed value of $\boldsymbol{\alpha}_t$ and $\boldsymbol{\alpha}_k$. Next, we optimize the value of $\boldsymbol{\alpha}_t$ and $\boldsymbol{\alpha}_k$ for a fixed value of $\boldsymbol{\delta}_s$. This process is repeated until convergence. More specifically, the problem of finding the appropriate preprocessing vector $\boldsymbol{\delta}_s$ for a given integer vectors $\mathbf{k}$ and $\mathbf{t}$ can be solved using Algorithm B-2.

V. NUMERICAL EXAMPLES

In this section, we provide some numerical examples. We measure the performance with the sum rate obtained under the constraint of symmetric rate, i.e., two times the maximum rate at which both users can communicate simultaneously with the destination, in a reliable fashion. 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; that from Transmitter B to the relay is modeled with a zero-mean Gaussian distribution

with variance σ_{br}^2; and that from the relay to the destination is modeled with a zero-mean Gaussian distribution with variance σ_{rd}^2. Similar assumptions and notations are used for the direct links from the transmitters to the destination. 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.

Fig. 2. Achievable sum rates under symmetric rate constraint. Numerical values are $P = 20$ dBW, $\sigma_{\mathrm{ar}}^2 = \sigma_{\mathrm{br}}^2 = \sigma_{\mathrm{rd}}^2 = 20$ dBW, and $\sigma_{\mathrm{ad}}^2 = \sigma_{\mathrm{bd}}^2 = 0$ dBW.

Figure 2 depicts the evolution of the sum rate obtained using the so-called *compute-and-forward at the relay* approach, i.e., the sum rate $R_{\mathrm{sum}}^{\mathrm{CoF}}$ of proposition 1 as given by (22); and the sum rate obtained using the so-called *compute-and-forward at the relay and compute at the destination* approach, i.e., the sum rate $R_{\mathrm{sum}}^{\mathrm{CoD}}$ of proposition 2 as given by (41), as functions of the signal-to-noise ratio SNR $= 10\log(P/N)$ (in decibels). Note that the curves correspond to numerical values of channel coefficients chosen such that the transmitters-to-relay links are of comparable quality to that of relay-destination link and are better than transmitters-to-destination links. For comparison reasons, the figure also shows the sum rates obtained using the classical strategies of Section II-C, i.e., the sum rate $R_{\mathrm{sum}}^{\mathrm{AF}}$ allowed by standard amplify-and-forward as given by (14), the sum rate $R_{\mathrm{sum}}^{\mathrm{DF}}$ allowed by standard decode-and-forward as given by (16), and the sum rate $R_{\mathrm{sum}}^{\mathrm{CF}}$ allowed by standard compress-and-forward as given by (21). All the sum rates are evaluated under the constraint of symmetric rate by the users.

For the example shown in Figure 2, we observe that both strategies of proposition 1 and proposition 2 perform well. The strategy of proposition 2 achieves a sum rate that is larger than what is obtained using classical relaying techniques, for all ranges of SNR. Also, the strategy of proposition 1 outperforms standard AF and DF for all ranges of SNR. Figure 3 depicts the same curves for other combinations of channel coefficients, chosen such that the transmitters-to-relay links are of better quality than all other links, but the relay-to-destination link is not better than the transmitters-to-destination links. In this case, we observe that the curves show behaviors that are generally similar to those of Figure 2. Also, note that, the standard AF can provide sum rates that are better than those obtained using the strategy of proposition 1 at large SNR is precisely due to that the transmitters-to-destination links are of better quality than the relay-to-destination link for the example
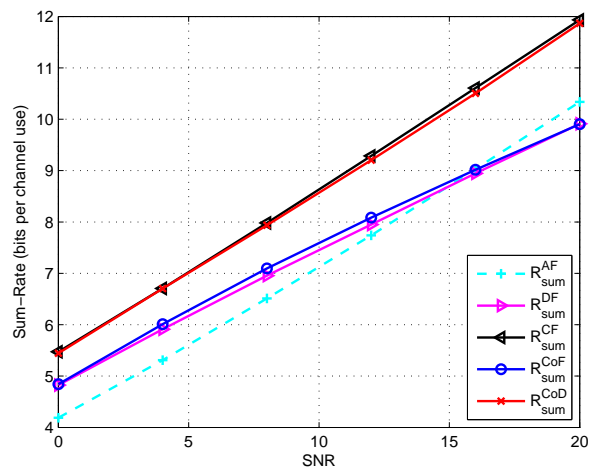

Fig. 3. Achievable sum rates under symmetric rate constraint. Numerical values are $P = 20$ dBW, $\sigma_{\mathrm{ar}}^2 = \sigma_{\mathrm{br}}^2 = 26$ dBW, $\sigma_{\mathrm{rd}}^2 = 14$ dBW, and $\sigma_{\mathrm{ad}}^2 = \sigma_{\mathrm{bd}}^2 = 0$ dBW.
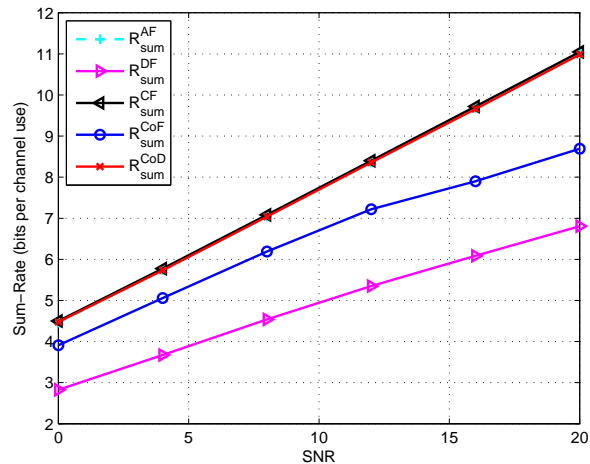

Fig. 4. Achievable sum rates under symmetric rate constraint. Numerical values are $P = 20$ dBW, $\sigma_{\mathrm{ar}}^2 = \sigma_{\mathrm{br}}^2 = 14$ dBW, $\sigma_{\mathrm{rd}}^2 = 26$ dBW, and $\sigma_{\mathrm{ad}}^2 = \sigma_{\mathrm{bd}}^2 = 0$ dBW.

shown in Figure 3. (Recall that, although the strategy of proposition 1 does not require decoding the messages from the two users individually, its performance is limited by the bottleneck at the relay node nonetheless; and this explains why it can be outperformed by standard AF at large values of SNR as visible from Figure 3). Figure 4 shows similar curves for other combinations of channel coefficients, chosen such that the relay-to-destination link has better quality than all other links.

Remark 2: Recall that, as we mentioned previously, the optimization Algorithm B associated with the strategy of proposition 2 is non-convex. In the figures shown in this paper, the sum rate provided by this strategy, which is based on Wyner-Ziv compression at the relay and computing appropriate equations at the destination, are obtained by selecting *only* certain initial points for Algorithm B. For this reason, the sum rate offered by the coding strategy of proposition 2, i.e., $R_{\mathrm{sum}}^{\mathrm{CoD}}$, can possibly be larger than what is shown in the figures if one considers more initial points. Also,

we note that even for initial points choices that yield a sum rate that is only as large as that obtained with standard compress-and-forward, there are advantages for using the coding scheme of proposition 2 instead of regular CF. In particular, while regular CF utilizes the rather complex joint typicality decoding, or maximum-Likelihood decoding, the coding scheme of proposition 2 utilizes linear decoding, and thus is more easily feasible.
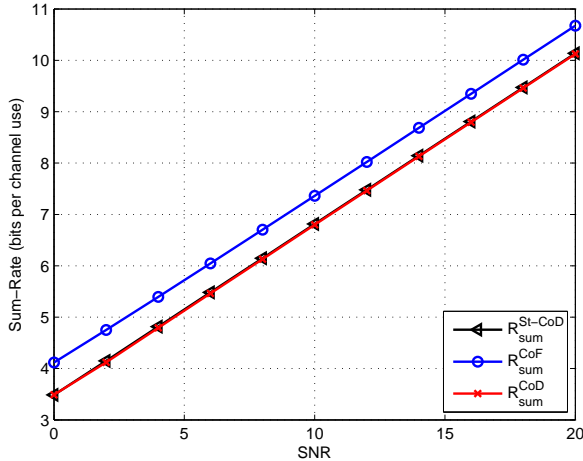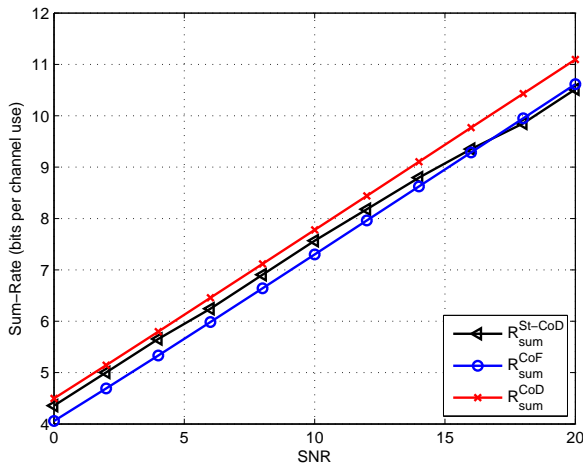
strategy of proposition 2 in a manner to get the first equation computed at the destination using *only* the signal received from the relay during the second transmission period, and the second equation computed at the destination using *only* the direct transmissions from the transmitters during the first transmission period. Figure 6 shows the improvement brought up by the aforementioned joint processing for the example depicted therein.

Fig. 5. Achievable sum rates under symmetric rate constraint. Numerical values are $P = 20$ dBW, $\sigma_{ar}^2 = \sigma_{br}^2 = \sigma_{rd}^2 = 20$ dBW, and $\sigma_{ad}^2 = \sigma_{bd}^2 = 0$ dBW.
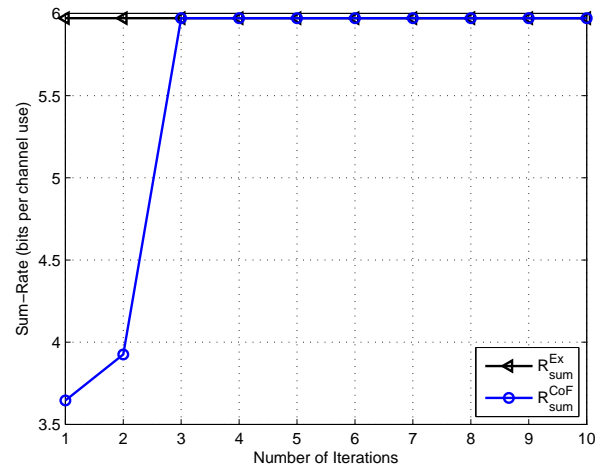

Fig. 7. Algorithm A v.s. Exhaustive search. Numerical values are $P = 20$ dBW, $\sigma_{ar}^2 = \sigma_{br}^2 = \sigma_{rd}^2 = 20$ dBW, and $\sigma_{ad}^2 = \sigma_{bd}^2 = 0$ dBW.

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 (59), as described in section IV-A. Recall that the algorithm involves allocating the integer coefficients and the users' 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 sources and relay by solving a series of geometric programs by means of an interior point approach [29].

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 $\|\mathbf{k}\|^2 \leq 1 + \|\mathbf{h}_r\|^2 \mathrm{snr}$ and $\|\mathbf{t}\|^2 \leq 1 + \|\mathbf{h}_d\|^2 \mathrm{snr}$, since otherwise the allowed sum rate is zero [31]. Let $R_{\mathrm{sum}}^{\mathrm{Ex}}$ denote the sum rate obtained by using the described exhaustive search-based algorithm. Figure V shows that the number of iterations required for Algorithm A to converge, i.e., yield the same sum-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.

Fig. 6. Achievable sum rates under symmetric rate constraint. Numerical values are $P = 20$ dBW, $\sigma_{ar}^2 = \sigma_{br}^2 = 14$ dBW, $\sigma_{rd}^2 = 26$ dBW, and $\sigma_{ad}^2 = \sigma_{bd}^2 = 0$ dBW.

Remark 3: 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, 3 and 4, the situation is reversed for the example shown in Figure 5. Related to this aspect, recall the discussion in Remark 1. Let us denote by $R_{\mathrm{sum}}^{\mathrm{St\text{-}CoD}}$ the achievable sum rate obtained by modifying the coding

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 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 sum 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 sum 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 sources and the relay into series of geometric programs. Comparing our coding schemes with classic relaying techniques, we show that the proposed strategies can offer substantial gains; and we illustrate this through some numerical examples.

ACKNOWLEDGMENT

REFERENCES

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

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

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

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

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

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

[7] C. Fragouli and E. Soljanin. Network coding fundamentals. *Found. Trends Commun. Info. Theory*, 2:1–333, 2007.

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

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

[10] J. H. Conway and N. J. A. Sloane. *Sphere Packing, Lattices and Groups*. third edition, John Willey & Sons INC., New York, 1988.

[11] K. Narayanan, M. P. Wilson, and A. Sprinston. Joint physical layer coding and network coding for bidirectional relaying. In *Proc. Allerton Conf. on Comm., Control, and Computing*, IL, USA, Sept. 2007.

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

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

[14] M. El Soussi, A. Zaidi, and L. Vandendorpe. Resource allocation for multiple access relay channel with a compute-and-forward relay. In *Int. Symposium on Wireless Communication Systems*, pages 809–813, Nov 2011.

[15] M. El Soussi, A. Zaidi, and L. Vandendorpe. Iterative sum-rate optimization for multiple access relay channels with a compute-and-forward relay. In *IEEE International Conference on Communications*, June 2012.

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

[17] M. Chiang, C. W. Tan, D. P. Palomar, D. O'Neill, and D. Julian. Power control by geometric programming. *IEEE Trans. Wireless Communications*, 6:2640–2651, 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. First IEEE Conference on Sensor and Ad Hoc Communications and Networks*, pages 191–199, Oct. 2004.

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

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

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

[22] G. D. Forney. Coset-codes: Part I: Introduction and geometrical classification. *IEEE Trans. Inf. Theory*, IT-34:1123–1151, 1988.

[23] G. D. Forney. On the role of MMSE estimation in approaching the information-theoretic limits of linear Gaussian channels: Shannon meets wyner. In *Proc. 41st Annual Allerton Conf. Comm. Control, Computing*, pages 430–439, Monticello, IL, Oct. 2003.

[24] U. Erez, S. Shamai (Shitz), and R. Zamir. Capacity and lattice strategies for cancelling known interference. *IEEE Trans. Inf. Theory*, IT-51:3820–3833, Nov. 2005.

[25] U. Erez and R. Zamir. Achieving $\frac{1}{2}\log(1+snr)$ on the awgn channel with lattice encoding and decoding. *IEEE Trans. Inf. Theory*, 50:2293–2314, Oct. 2004.

[26] C. A. Floudas. Nonlinear and mixed integer optimization. In *Oxford University Press*, 1995.

[27] E. L. Lawler and D.E. Wood. Branch-and -bound methods: A survey. *Operations Research*, pages 699–719, 1966.

[28] L. A. Wolsey. *Integer programming*. John Wiley and Sons, 1998.

[29] M. Chiang. Geometric programming for communication systems. In *Foundation and Trends in Commun. Inf. Theory*, pages 1–156, Aug. 2005.

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

[31] J. Zhan, B. Nazer, M. Gastpar, and U. Erez. Mimo compute-and-forward. In *IEEE Int. Symposium on Inf. Theory*, pages 2848–2852, June 2009.