2012 IEEE 27-th Convention of Electrical and Electronics Engineers in Israel

Achieving the Gains Promised by Integer-Forcing Equalization with Binary Codes

Or Ordentlich
Department of EE-Systems
Tel Aviv University
Ramat Aviv, 69978
Israel
ordent@eng.tau.ac.il

Uri Erez
Department of EE-Systems
Tel Aviv University
Ramat Aviv, 69978
Israel
uri@eng.tau.ac.il

***Abstract*—We address the problem of constructing practical coding schemes for Integer-Forcing (IF) equalization. Previously proposed IF schemes suggested the use of nested lattice codes or linear block codes over $\mathbb{Z}_q$. While such codes have good properties from a theoretic point of view, their implementation complexity may become significant when q is large. In this work we show that at high transmission rates, linear codes suitable for IF equalization may be constructed using binary codes, incurring only moderate losses compared to optimal capacity achieving nested lattice codes. The technique is based on Ungerboeck's method of mapping by set partitioning.**

I. Introduction

Integer-Forcing (IF) was proposed by Nazer and Gapstar [1] in the context of linear Gaussian networks, and then extended to Gaussian MIMO channels and Gaussian intersymbol-interference (ISI) channels in [2], [3], [4].

The main idea behind IF is that the linear nature of the channel can be exploited if the codebook used for transmitting messages over the channel is also linear. A linear channel typically "mixes" several transmitted codewords in a linear manner. The receiver is usually interested in reconstructing all or some of these codewords. Traditional receivers attempt to decode each one of these codewords separately treating the other codewords as interferences caused by the channel. An Integer-Forcing receiver, rather than trying to eliminate the effect of these other codewords, tries to *align* them. This can be done using the properties of a linear codebook over $\mathbb{Z}_q$: it is closed under integral linear combinations (over $\mathbb{Z}_q$). Therefore, if all codewords transmitted over the channel are taken from the same codebook $\mathcal{C}$ (which is linear over $\mathbb{Z}_q$), an IF receiver equalizes the channel coefficients to be integers, resulting in a received signal which is an integral linear combination of codewords (plus noise) and hence also a codeword. Then, if this set of linear combinations is full-rank over $\mathbb{Z}_q$, the original transmitted codewords can be reconstructed. This full-rank condition is most easily satisfied if q is prime, in which case the ring $\mathbb{Z}_q$ is a field.

Binary linear codes have been intensively studied over the past 60 years, and practical codes that approach the Shannon theoretical limits such as LDPC codes are implemented in many communication systems. While practical realizations have been suggested for linear codes over $\mathbb{Z}_q$ ($q > 2$) for moderate values of q, currently known decoding algorithms suffer from complexity which rapidly grows with the alphabet size. Designing low-complexity coding schemes suitable for IF receivers is thus of interest.

A recent work [7] addressed this problem using an algebraic approach. In this work we present a very simple coding and decoding scheme that constructs linear codes over $\mathbb{Z}_{2^M}$ using a single binary code.[1] The technique we use utilizes Ungerboeck's ideas for spectrally efficient transmission over the additive white Gaussian noise (AWGN) channel.

The paper is organized as follows. Section II reviews some applications of IF receivers. Section III describes our proposed coding and decoding schemes. Section IV demonstrates the performance of the proposed scheme via simulation results, and Section V concludes the paper.

II. Applications of Integer-Forcing equalization

We first briefly recall how IF may be used in linear Gaussian networks, Gaussian MIMO channels and Gaussian ISI channels. As we are only interested in conveying the main ideas of the presented schemes, the descriptions in this section are not worked out in full detail.

A. The linear Gaussian network

In [1], a linear Gaussian network was considered, consisting of L distributed (non-cooperating) users, K distributed relays and a centralized decoder. All relays are connected to the centralized decoder through bit pipes, each with rate $\mathcal{R}_0$, see Figure 1. Each relay observes a linear combination of the transmitted codewords corrupted by Gaussian noise. Namely, relay k observes

$$\mathbf{y}_k = \sum_{l=1}^{L} h_{kl}\mathbf{x}_l + \mathbf{z}_k, \tag{1}$$

where $\mathbf{x}_l \in \mathbb{R}^{1\times N}$ is the codeword transmitted by user l, h_{kl} is the channel coefficient between user l and relay k

[1]The fact that $q = 2^M$ is not a prime does not pose any difficulties in reconstructing the original transmitted codewords as demonstrated in the sequel

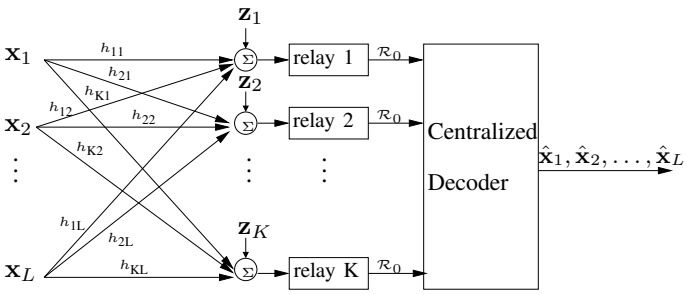

Fig. 1. A schematic description of the linear Gaussian network.

and $\mathbf{z}_k \in \mathbb{R}^{1\times N}$ is Gaussian noise which is uncorrelated in time and in space, i.e., all components of $\mathbf{z}_k$ are i.i.d., and the additive noise vectors seen by different relays are mutually independent. The centralized decoder is interested in reproducing the transmitted codewords.

The IF approach as was presented and analyzed in [1] suggests that each relay should try to decode an integral linear combination of the transmitted codewords rather than trying to decode a codeword transmitted by a certain user. If all the codewords are taken from the same linear codebook $\mathcal{C}$ over $\mathbb{Z}_q$, then from the linearity of the codebook it follows that

$$\mathbf{u}_k = \left(\sum_{l=1}^{L} a_{kl}\mathbf{x}_l\right) \bmod q \in \mathcal{C}, \tag{2}$$

where $a_{kl} \in \mathbb{Z}$ are integer coefficients. Relay k therefore chooses a set of integer coefficients $\{a_{kl}\}_{l=1}^{L}$ and computes

$$\begin{aligned}\tilde{\mathbf{y}}_k &= (\alpha_k \mathbf{y}_k) \bmod q \\ &= \left(\sum_{l=1}^{L} a_{kl}\mathbf{x}_l + \sum_{l=1}^{L} (\alpha_k h_{kl} - a_{kl})\,\mathbf{x}_l + \alpha_k \mathbf{z}_k\right) \bmod q \\ &= (\mathbf{u}_k + \tilde{\mathbf{z}}_k) \bmod q,\end{aligned} \tag{3}$$

where

$$\tilde{\mathbf{z}}_k = \sum_{l=1}^{L} (\alpha_k h_{kl} - a_{kl})\,\mathbf{x}_l + \alpha_k \mathbf{z}_k. \tag{4}$$

The purpose of the scaling factor α_k in (3) is to scale $\{h_{kl}\}_{l=1}^{L}$ towards $\{a_{kl}\}_{l=1}^{L}$. It is evident from (4) that the noise term $\tilde{\mathbf{z}}_k$ consists of a Gaussian component and a component that corresponds to the "distance" between $\{\alpha_k h_{kl}\}_{l=1}^{L}$ and the integer vector $\{a_{kl}\}_{l=1}^{L}$. Thus, if the channel coefficients $\{h_{kl}\}_{l=1}^{L}$ are "close" to being integers, the noise enhancement caused by the scaling of $\mathbf{y}_k$ would not be big. Each relay k therefore looks for the set of integers which is the nearest to the direction of its channel coefficients and tries to decode it. The decoded codewords are passed to the centralized decoder through the bit pipes. If the centralized decoder receives a full-rank set (over $\mathbb{Z}_q$) of integral combinations of the transmitted codewords it can separate them and recover the original codeword each user transmitted.

B. The Gaussian MIMO channel

Consider the channel model

$$\mathbf{Y} = \mathbf{H}\mathbf{X} + \mathbf{Z} \tag{5}$$

where $\mathbf{H} \in \mathbb{R}^{N_r\times N_t}$ is the channel's fading matrix, $\mathbf{X} \in \mathbb{R}^{N_t\times N}$ is a matrix such that each one of its rows is a codeword of length N, and $\mathbf{Z} \in \mathbb{R}^{N_r\times N}$ is a matrix of i.i.d. Gaussian components. The receiver is interested in decoding all the transmitted codewords, i.e. all of the rows in the matrix $\mathbf{X}$. The simplest approach would be to invert the channel $\mathbf{H}$ by

$$\mathbf{Y}' = \mathbf{H}^{\dagger}\mathbf{Y} = \mathbf{X} + \mathbf{Z}' \tag{6}$$

where $\mathbf{H}^{\dagger} = \left(\mathbf{H}^T\mathbf{H}\right)^{-1}\mathbf{H}^T$, and $\mathbf{Z}' = \mathbf{H}^{\dagger}\mathbf{Z}$. If the matrix $\mathbf{H}$ is near singular, the noise enhancement caused by this approach can significantly degrade the performance of this receiver compared to an optimal one. An IF receiver equalizes the channel $\mathbf{H}$ to an integer channel matrix $\mathbf{A} \in \mathbb{Z}^{N_t\times N_t}$, rather than the unit matrix $\mathbf{I}_{N_t\times N_t}$. The receiver therefore computes

$$\tilde{\mathbf{Y}} = \left(\mathbf{A}\mathbf{H}^{\dagger}\mathbf{Y}\right) \bmod q = \left(\mathbf{A}\mathbf{X} + \tilde{\mathbf{Z}}\right) \bmod q \tag{7}$$

where $\tilde{\mathbf{Z}} = \mathbf{A}\mathbf{H}^{\dagger}\mathbf{Z}$. If again all the transmitted codewords (all the rows of $\mathbf{X}$) are taken from the same linear codebook $\mathcal{C}$ over $\mathbb{Z}_q$, then from the linearity of $\mathcal{C}$ it follows that

$$\mathbf{U} = (\mathbf{A}\mathbf{X}) \bmod q \tag{8}$$

is a matrix of dimensions $N_t \times N$ such that each of its rows is a codeword in $\mathcal{C}$. The decoder can therefore first decode the rows of $\mathbf{U}$, and then use them in order to reconstruct the original transmitted codewords, assuming $\mathbf{A}$ is full-rank over $\mathbb{Z}_q$.

The Zero-Forcing (ZF) equalizer (6) always chooses $\mathbf{A} = \mathbf{I}_{N_t\times N_t}$, regardless of $\mathbf{H}$, whereas the IF equalizer allows any choice of $\mathbf{A} \in \mathbb{Z}^{N_t\times N_t}$ adding another degree of freedom that can often significantly improve performances by reducing the noise enhancement caused by the equalization step. For more details see [3].

C. The Gaussian ISI channel

Consider the channel model

$$y_n = \sum_{l=0}^{L} h_l x_{n-l} + z_n = x_n + \mathrm{ISI}_n + z_n \tag{9}$$

where z_n is white Gaussian noise, $\{h_l\}_{l=0}^{L}$ are the channel coefficients, and ISI_n is intersymbol interference resulting from other data symbols. Using D-transform notation we have

$$Y(D) = H(D)\,X(D) + Z(D). \tag{10}$$

The most straightforward approach the receiver could take would be to cancel the ISI completely by applying the linear

filter $A_{\text{ZF}}(D) = 1/H(D)$, which results in the equivalent channel

$$\begin{aligned} Y'(D) &= A_{\text{ZF}}(D)\left(H(D)X(D) + Z(D)\right) \\ &= X(D) + Z'(D), \end{aligned} \tag{11}$$

where $Z'(D) = Z(D)/H(D)$ is colored Gaussian noise. If $H(D)$ has zeros near the unit circle, the resulting noise z'_n will have a much larger variance than z_n, resulting in poor performances. An IF equalizer chooses $A_{\text{IF}}(D) = I(D)/H(D)$, where $I(D)$ is an integer-valued filter, giving rise to the equivalent channel

$$\begin{aligned} \tilde{Y}(D) &= \left(A_{\text{IF}}(D)\left(H(D)X(D) + Z(D)\right)\right) \bmod q \\ &= \left(I(D)X(D) + \tilde{Z}(D)\right) \bmod q, \end{aligned} \tag{12}$$

where $\tilde{Z}(D) = Z(D)\frac{I(D)}{H(D)}$. The linear convolution performed by the channel can easily be transformed into a cyclic one using standard techniques (as commonly used in OFDM). Thus, (12) can be transformed into

$$\tilde{y}_n = (\mathbf{i} \otimes \mathbf{x} + \tilde{\mathbf{z}}) \bmod q, \tag{13}$$

where $\mathbf{i}$ is a vector holding the coefficients of $I(D)$, and $\otimes$ denotes cyclic convolution.

Definition 1: A linear block code $\mathcal{C}$ of length N over $\mathbb{Z}_q$ is called cyclic, if for every codeword $\mathbf{x} \in \mathcal{C}$, all cyclic shifts of $\mathbf{x}$ are also codewords in $\mathcal{C}$.

The following is an immediate consequence of the definition.

Lemma 1: Let $\mathcal{C}$ be a cyclic code of length N over $\mathbb{Z}_q$. Then for any vector $\mathbf{i}$ of length N with integer entries

$$\mathcal{C} \otimes \mathbf{i} \subseteq \mathcal{C}.$$

That is, $\mathcal{C}$ is closed under integer-valued cyclic convolution over $\mathbb{Z}_q$.

It follows from Lemma 1 that if $\mathbf{x}$ is a codeword in the linear cyclic codebook $\mathcal{C}$, so is the cyclic convolution over $\mathbb{Z}_q$ between $\mathbf{x}$ and $\mathbf{i}$. The receiver may thus decode $\mathbf{x} \otimes \mathbf{i}$ and reconstruct the original transmitted codeword $\mathbf{x}$ from it. The IF receiver is depicted in Figure 2. For more details see [4]. If there exists an integer-valued filter $I(D)$ such that the frequency response of the filter $A_{\text{IF}}\left(e^{j\omega}\right) = I\left(e^{j\omega}\right)/H\left(e^{j\omega}\right)$ is nearly flat, the noise enhancement incurred by $A_{\text{IF}}(D)$ would be small, and the performance of the IF equalizer would not have a significant loss compared to an optimal receiver.

III. Combining IF equalization with natural labeling

In the previous section we recalled how IF equalization in some cases may outperform traditional equalization techniques with minimal increase in complexity. However, the IF equalization techniques described require the use of a linear (perhaps cyclic) code over $\mathbb{Z}_q$.[2] When working at a high signal-to-noise ratio (SNR), high transmission rates are desired. This implies that the cardinality of the codebook's alphabet must be larger than 2, and the use of binary codes is precluded. While from a purely theoretic perspective the requirement that the codes be linear over an alphabet with a large cardinality incurs no performance loss, from a practical point of view it is beneficial if coding and decoding may be restricted to using binary codes.

We now show how one can incorporate binary codes to yield higher-order (larger constellation) linear codes suitable for IF equalization. The technique we use is *mapping by set partitioning* (MSP), which is commonly used for the AWGN channel.

We begin by reviewing the concept of MSP for AWGN channels. In his celebrated paper [9] Ungerboeck observed that since the error probability of a maximum-likelihood (ML) decoder is determined by the Euclidean distance (ED) between codewords rather than the Hamming distance, the performance of a code heavily depends on the mapping between codewords and constellation points. This mapping may be designed such as to maximize the minimum ED between points in the signal space corresponding to different codewords. The design technique developed in [9] in order to meet this criterion is called MSP. A simple example of a coded modulation scheme following the guidelines of Ungerboeck's MSP can be obtained using a single encoded binary stream in conjunction with *natural labeling*. Specifically, in this scheme a vector of length N and rate $R = r + (M-1)$ bits/channel use is generated by dividing the information bits into a group of Nr ($r < 1$) bits to be encoded using a binary code of rate r, and $M-1$ groups of N bits each - $\mathbf{x}_{u_1} \ldots, \mathbf{x}_{u_{M-1}}$ to be transmitted uncoded. The coded and uncoded bits are then mapped to a 2^M – PAM constellation by[3]

$$\mathbf{x} = \mathbf{x}_c + \sum_{m=1}^{M-1} \mathbf{x}_{u_m} 2^m. \tag{14}$$

The result of this construction is a linear block code over $\mathbb{Z}_{2^M}$. From a nested lattices [10] point of view it can be viewed as using Construction A (see [11]) with a binary linear code in order to construct the fine lattice, where the coarse lattice is $\mathbb{Z}_{2^M}$.

The decoder observes

$$\mathbf{y} = \mathbf{x} + \mathbf{z} \tag{15}$$

where $\mathbf{z}$ is WGN. It first reduces the observations modulo 2 resulting in

$$\mathbf{y}_{\text{bin}} = (\mathbf{x} + \mathbf{z}) \bmod 2 = (\mathbf{x}_c + \mathbf{z}) \bmod 2. \tag{16}$$

The codeword $\mathbf{x}_c$ can now be decoded from $\mathbf{y}_{\text{bin}}$. After decoding $\mathbf{x}_c$, it can be subtracted from $\mathbf{y}$ and the decoder

[2]The coding scheme can be extended to allow for shaping by using a nested lattices codebook.

[3]The transmitted signal, in effect, would be $c(\mathbf{x} - d)$ where d is a constant chosen such that the average energy of the constellation points would be minimized, and c is chosen such that the power constraint would be met. We use the constellation $\{0, 1, \ldots, 2^M - 1\}$ throughout this paper for simplicity of exposition.

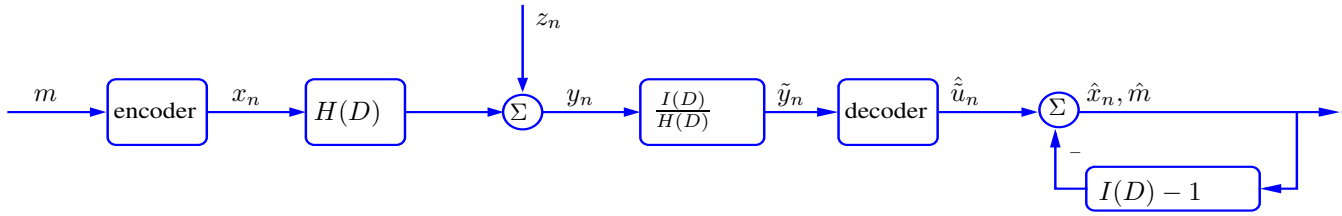

Fig. 2. A schematic description of a IF equalizer for the linear Gaussian channel.

remains only with the task of detecting the uncoded bits which can easily be accomplished by a slicer. This simple coded modulation scheme can achieve a high coding gain, since the points of the constellation with the smallest ED are protected by the code, and once the coded bits are known, the minimum possible ED between constellation points is doubled.

We now show how the same transmission scheme can be applied for each of the three applications of IF introduced in the previous section.

A. Decoding in the linear Gaussian network

Lemma 2: Let the transmitted signal each user l transmits in the linear Gaussian network model introduced in Section II-A be constructed by natural labeling, i.e.,

$$\mathbf{x}_l = \mathbf{x}_{c_l} + \sum_{m=1}^{M-1} \mathbf{x}_{u_{m_l}}, \tag{17}$$

where $\mathbf{x}_{c_l}$, $l = 1, \ldots, L$ are all codewords from the same binary linear codebook $\mathcal{C}$. Substituting $q = 2$ in (3) results in

$$\tilde{\mathbf{y}}_{k_{\text{bin}}} = \left(\left(\sum_{l=1}^{L} a_{kl} \mathbf{x}_{c_l} \right) \bmod 2 + \tilde{\mathbf{z}}_k \right) \bmod 2, \tag{18}$$

Proof:

$$\begin{aligned}
\tilde{\mathbf{y}}_{k_{\text{bin}}} &= \left(\sum_{l=1}^{L} a_{kl} \mathbf{x}_l + \tilde{\mathbf{z}}_k \right) \bmod 2 \\
&= \left(\sum_{l=1}^{L} a_{kl} \left(\mathbf{x}_{c_l} + \sum_{m=1}^{M-1} \mathbf{x}_{u_{m_l}} 2^m \right) + \tilde{\mathbf{z}}_k \right) \bmod 2 \\
&= \left(\sum_{l=1}^{L} a_{kl} \mathbf{x}_{c_l} + \sum_{m=1}^{M-1} 2^m \sum_{l=1}^{L} a_{kl} \mathbf{x}_{u_{m_l}} + \tilde{\mathbf{z}}_k \right) \bmod 2 \\
&= \left(\left(\sum_{l=1}^{L} a_{kl} \mathbf{x}_{c_l} \right) \bmod 2 + \tilde{\mathbf{z}}_k \right) \bmod 2,
\end{aligned} \tag{19}$$

where the last equality follows since $\sum_{l=1}^{L} a_{kl} \mathbf{x}_{u_{m_l}} \in \mathbb{Z}^{1 \times N}$, and thus the reduction modulo 2 completely eliminates the effect of the uncoded bits from $\tilde{\mathbf{y}}_{k_{\text{bin}}}$. ■

A direct consequence of Lemma 2 is that the decoding at each relay can be performed via a two-stage process. At the first stage relay k scales its observation by α_k and reduces it modulo 2. From the linearity of the codebook it follows that $\tilde{\mathbf{u}}_{k_{\text{bin}}} = \left(\sum_{l=1}^{L} a_{kl} \mathbf{x}_{c_l} \right) \bmod 2 \in \mathcal{C}$, and can be decoded. In the second decoding stage the decoder uses the information from the previous step in order to decide whether each entry of

$$\begin{aligned}
\tilde{\mathbf{u}}_k &= \sum_{l=1}^{L} a_{kl} \mathbf{x}_l \\
&= \sum_{l=1}^{L} a_{kl} \mathbf{x}_{c_l} + \sum_{m=1}^{M-1} 2^m \sum_{l=1}^{L} a_{kl} \mathbf{x}_{u_{m_l}}
\end{aligned} \tag{20}$$

is even or odd. Once the decoder knows which of the entries of $\tilde{\mathbf{u}}_k$ are even and which are odd, slicers with double step size can be applied in order to detect it.

Each relay k then passes the integral linear combination it decoded along with the coefficients $\{a_{kl}\}_{l=1}^{L}$ to the centralized decoder through the bit pipes, and the centralized decoder can now reconstruct the original transmitted codewords assuming the set of linear integral equations is full-rank over the *reals*.[4]

B. Decoding in the Gaussian MIMO channel

Lemma 3: Let the matrix $\mathbf{X}$ in the Gaussian MIMO channel model introduced in Section II-B be constructed by natural labeling, i.e.,

$$\mathbf{X} = \mathbf{X}_c + \sum_{m=1}^{M-1} \mathbf{X}_{u_m} 2^m, \tag{21}$$

where the rows of $\mathbf{X}_c$ are codewords from the same binary linear codebook $\mathcal{C}$ and the matrices $\mathbf{X}_{u_m}$ $m = 1, \ldots, M-1$ are matrices of uncoded bits, then substituting $q = 2$ in (7) gives

$$\tilde{\mathbf{Y}}_{\text{bin}} = \left((\mathbf{A}\mathbf{X}_c) \bmod 2 + \tilde{\mathbf{Z}} \right) \bmod 2 \tag{22}$$

Proof: The proof follows along the same lines that of Lemma 2 and is omitted here. ■

From the linearity of $\mathcal{C}$ it follows that each row of $\tilde{\mathbf{U}}_{\text{bin}} = (\mathbf{A}\mathbf{X}_c) \bmod 2$ is a codeword in $\mathcal{C}$. Thus, as in the linear Gaussian network, the decoder first decodes $\tilde{\mathbf{U}}_{\text{bin}}$ which is possible due to Lemma 3. Once $\tilde{\mathbf{U}}_{\text{bin}}$ is decoded, the decoder knows which of the entries of $\tilde{\mathbf{U}} = \mathbf{A}\mathbf{X}$ are even and which are odd, and can then apply slicers of double step size in order to detect them. After successfully decoding $\tilde{\mathbf{U}}$, the decoder can reconstruct $\mathbf{X}$ assuming $\mathbf{A}$ is full rank over the reals.

[4]The rate of the bit pipes $\mathcal{R}_0$ may be too small in order to pass the decoded codewords over the reals. In this case the relays can reduce the decoded codewords modulo q' before transmitting them, where q' is the smallest integer greater than 2^M such that the set of linear equations the relays decoded is full-rank over q'.

C. Decoding in the Gaussian ISI channel

Lemma 4: Let the transmitted signal in the Gaussian ISI channel introduced in Section II-C be constructed by natural labeling as in (14), where $\mathbf{x}_c$ is a codeword in a linear *cyclic* binary codebook $\mathcal{C}$. Substituting $q = 2$ into (13) gives

$$\tilde{y}_{\text{bin}} = ((\mathbf{i} \otimes \mathbf{x}_c) \bmod 2 + \tilde{\mathbf{z}}) \bmod 2. \tag{23}$$

Proof: This proof is similar to the proof of Lemma 2 and is omitted. ■

It follows from Lemma 1 that $\tilde{\mathbf{u}}_{\text{bin}} = (\mathbf{i} \otimes \mathbf{x}) \bmod 2$ is a codeword from $\mathcal{C}$ and can thus be decoded. Once the decoder has complete knowledge of $\tilde{\mathbf{u}}_{\text{bin}}$, it knows which samples of $\tilde{\mathbf{u}} = \mathbf{i} \otimes \mathbf{x}$ are even and which are odd, and can apply a slicer with double step size in order to decode it. After $\tilde{\mathbf{u}}$ is decoded $\mathbf{x}$ can easily be reconstructed from it.

Remark - Unfortunately the proposed natural labeling scheme can not be extended to multilevel coding with *multiple* coded layeres, i.e., it is not possible to transmit more than one coded layer while maintaining the simple decoding scheme we described. The reason for this is that in the first decoding step we only decode a linear integral combination of the coded bits *reduced modulo 2*. This enables us to divide the signal space into cosets of even points and odd points, but we cannot distinguish one point of a coset from another. If all the layers but the first are uncoded we don't need to distinguish between the points in the coset since all of them could have equally been transmitted. However if there are more coded layers we cannot distinguish which of the points are legal codewords and which are not.

IV. Simulation results

In order to demonstrate the proposed scheme we consider the Gaussian MIMO channel with the channel matrix $\mathrm{H} = [1 - 2; 1 - 3]$. We use a natural labeling scheme with rate $R = 3.83$bits/channel use for each transmit antenna. The coded bits are encoded using a linear LDPC code with rate $5/6$ and a blocklength of $64,800$bits. We evaluate the bit error rate (BER) of the IF equalizer and the ZF equalizer vs. the SNR per transmit antenna in Figure 3. The performance of the same encoding scheme with optimal decoding over an AWGN is also plotted for reference.

The results demonstrate a gap-to-capacity of 4dB at BER $= 10^{-6}$, which is not much different than the gap-to-capacity one would attain using the same transmission scheme with optimal decoding over the AWGN channel. The threshold phenomena in the IF curve of Figure 3 is caused by the transition from the SNR regime where the BER is dictated by the performances of the coded layer, to the SNR regime where the coded bits are decoded successfully, and the uncoded bits constitute the bottleneck.

V. Conclusion

We have reviewed three scenarios where IF equalization schemes are beneficial and designed a practical coding scheme suitable for IF equalization based on natural labeling. Simulation results confirm that the losses incurred by the suboptimal encoding and decoding scheme we have proposed are similar to those of standard coded modulation techniques for the AWGN channel.

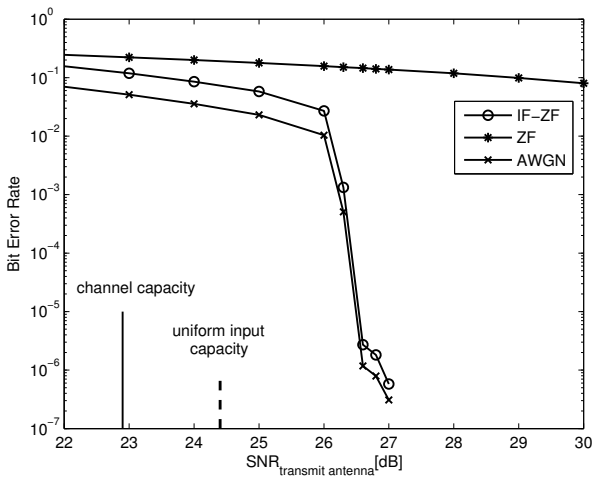

Fig. 3. Bit error rate vs. SNR per transmit antenna curve for IF and ZF equalizers evaluated over the Gaussian MIMO channel $\mathrm{H} = [1\ -2; 1\ -3]$. Natural labeling transmission with $R = 3.83$bits/channel use was used. The performance of the same encoding scheme with optimal decoding is also evaluated over an AWGN channel for reference.

Acknowledgment

This work was supported in part by the U.S. - Israel Binational Science Foundation under grant 2008/455.

References

[1] B. Nazer and M. Gastpar, "Compute-and-Forward: Harnessing Interference through Structured codes", In *IEEE Transactions on Information Theory*, Submitted Aug. 2009. See http://arxiv.org/abs/0908.2119.

[2] J. Zhan, B. Nazer, M. Gastpar and U. Erez, "MIMO compute-and-forward", In *Seoul, South Korea*, June 2009

[3] J. Zhan, B. Nazer, U. Erez and M. Gastpar, "Integer-forcing linear receivers", in *Proc. ISIT 2010*, Austin, Texas, June 2010.

[4] O. Ordentlich and U. Erez, "Cyclic coded Integer-Forcing equalization", In *48th Annual Allerton Conference*, , (Monticello, IL), September 2010.

[5] K. Narayanan, M. P. Wilson, and A. Sprintson, "Joint physical layer coding and network coding for bi-directional relaying", In *45th Annual Allerton Conference*, , (Monticello, IL), September 2007.

[6] W. Nam, S. -Y. Chung, and Y. H. Lee, "Nested lattice codes for gaussian relay networks with interference", In *IEEE Transactions on Information Theory*, Submitted February 2009. See http://arxiv.org/abs/0902.2436.

[7] C. Feng, D. Silva and F. R. Kschischang, "An algebraic approach to physical-layer network coding", In *Proc. ISIT 2010*, Austin, Texas, June 2010.

[8] G. D. Forney, Jr. and G. Ungerboeck, "Modulation and coding for linear Gaussian channels", In *IEEE Trans. Information Theory*, vol. IT-44, pp. 2384–2415, Oct. 1998.

[9] G. Ungerboeck, "Channel coding with multilevel/phase signals", In *IEEE Trans. Information Theory*, vol. IT-28, pp. 55–67, Oct. 1982.

[10] R. Zamir, S. Shamai (Shitz), and U. Erez, Nested linear/lattice codes for structured multiterminal binning, In *IEEE Trans. Inf. Theory*, vol. 48, no. 1, pp. 1250-1276, Jun. 2002.

[11] J. H. Conway and N. J. A. Sloane, "Sphere Packings, Lattices and Groups", New York: Springer-Verlag, 1988.