Superposition Coding for Side-Information Channels*

Amir Bennatan†
Tel-Aviv University, Israel

David Burshtein†
Tel-Aviv University, Israel

Giuseppe Caire‡
Formerly with Eurecom Institute, France. Currently with the University of Southern California

Shlomo Shamai‡
Technion, Israel

Abstract

We present simple, practical codes designed for the binary and Gaussian dirty-paper channels. We show that the dirty paper decoding problem can be transformed into an equivalent multiple-access decoding problem, for which we apply superposition coding. Our concept is a generalization of the nested lattices approach of Zamir, Shamai and Erez. In a theoretical setting, our constructions are capable of achieving capacity using random component codes and maximum-likelihood decoding. We also present practical implementations of the constructions, and simulation results for both dirty-paper channels. Our results for the Gaussian dirty-paper channel are on par with the best known results for nested-lattices. We discuss the binary dirty-*tape* channel, for which we present a simple, effective coding technique. Finally, we propose a framework for extending our approach to general Gel'fand-Pinsker channels.

Index Terms - dirty paper, dirty tape, multiple-access channel, side information, superposition coding.

*IEEE Transactions on Information Theory, volume 52, no. 5, pp. 1872-1889, May 2006.

†The first two authors were supported by the Israel Science Foundation, grant no. 22/01-1, by an equipment grant from the Israel Science Foundation to the school of Computer Science in Tel Aviv University and by a fellowship from The Yitzhak and Chaya Weinstein Research Institute for Signal Processing at Tel Aviv University.

‡The last two authors were supported by the EU 6th framework programme, via the NEWCOM network of excellence.

I Introduction

Side-information channels were first considered by Shannon [43]. Such channels are characterized by an input X, output Y and state-dependent transition probability $p(y|x,s)$ where the channel state S is i.i.d., known to the transmitter and unknown to the receiver. Shannon [43] considered the case of state sequence known causally. Kusnetsov and Tsybakov [32] were the first to consider the case of state sequence known non-causally, and Gel'fand and Pinsker [29] obtained the capacity formula for this case.

The binary and Gaussian side-information channels are given by

$$Y = X + S + Z \tag{1}$$

With binary side-information channels, addition is over the binary field $\mathbb{F}_2$. The random variable S is referred to as *interference* (following [21]) and constitutes the channel state, which is known to the encoder. In this paper we assume S to be i.i.d. with uniform probability $P(S=0) = P(S=1) = 1/2$. Z is binary symmetric noise, distributed as Bernoulli(p), and X is the channel input, subject to an input constraint $1/n \cdot d_H(\mathbf{x}, \mathbf{0}) \le W$, where $d_H(\mathbf{x}, \mathbf{y})$ denotes Hamming distance between two n-vectors $\mathbf{x}$ and $\mathbf{y}$.

With Gaussian side-information channels, addition in (1) is over the real number field. The channel input X is subject to a power constraint P_X, i.e. $1/n \cdot \|\mathbf{x}\|^2 \le P_X$, where $\|\mathbf{x}\|^2$ denotes the square-distance norm of $\mathbf{x}$. The noise Z is distributed as a zero-mean Gaussian variable with variance P_Z. S is the known interference. We make no assumptions on the distribution of S.

Following Costa's "Writing on Dirty Paper" famous title [17], when the interference S is known non-causally, these channels are referred to as "dirty paper" channels. By analogy, the case of causally known interference is called "dirty tape" (see e.g. [9]).

Costa was the first to examine the Gaussian dirty paper problem. He obtained the remarkable result that the interference, known only to the encoder, incurs no loss of capacity in comparison with the standard interference-free channel. Costa assumed that S is Gaussian i.i.d distributed. This result was extended in [16] and [21] to arbitrarily distributed interference. Pradhan *et al*. [39] and Barron *et al*. [3] obtained the capacity of the binary dirty paper channel. Unlike the Gaussian dirty paper channel, here a penalty *is* paid for the interference known only to the transmitter.

Applications for side-information problems include data-hiding (see [14, 19, 36, 25]), where a host signal is modelled as interference, and a watermark is modelled as an additive transmitted signal X subject to a maximum distortion constraint. In the binary case, the host signal is typically a black and white image, or the least-significant bit-layer of a gray-scale image, and the signal is received through some memoryless transformation modelled as a BSC. In the Gaussian case the signals are allowed to be continuous, and the memoryless transformation is modelled as AWGN. Other applications of dirty tape include precoding for channels with ISI [21], where ISI is modelled as interference known causally at the encoder. Transmission over broadcast channels is an important application of dirty paper coding [11]. This application is particularly pronounced in the case of the MIMO Gaussian broadcast channel, see [52].

The achievability of capacity in dirty paper channels is proven by means of a random construction of codes and a random partition of their codewords into "bins". This method typically produces unstructured codes, which are infeasible for practical implementation. Zamir, Shamai and Erez [54] suggested a framework for introducing structure into the above "random binning" method. Their technique involves nested codes (and nested lattices). That is, they use a *fine* code $\mathcal{C}$ and a *coarse* code $\mathcal{C}_0$ such that $\mathcal{C}_0 \subset \mathcal{C}$. Their construction requires that the fine code $\mathcal{C}$ be designed as a good channel-code, while the coarse code $\mathcal{C}_0$ must be designed to be a good source-code.

LDPC codes are likely candidates for codes $\mathcal{C}$ and $\mathcal{C}_0$. However, although LDPC codes are well suited for channel coding, the problem of finding a good source-coding algorithm for them remains open. Unless such an algorithm is found, the codes in their current form are unsuitable for selection as $\mathcal{C}_0$. We would like to select $\mathcal{C}$ as an LDPC code, but the nested structure of $\mathcal{C}$ and $\mathcal{C}_0$ means that the codes are entangled in a way that restricts the independent selection of $\mathcal{C}$. One approach for challenging this problem was considered by Philosof *et al.* [38, 37] and Erez and ten Brink [23] using coset dilution. In this paper we present an alternative to the nested lattices method of [54] using superposition of codes, which enables independent selection of a *quantization* and an *information-bearing* code.

We begin in Section II by presenting superposition-coding for the binary dirty-paper channel. We define the codes used and discuss encoding and decoding. We also show that in a random-coding setting, using minimum-distance encoding and maximum-likelihood decoding, our codes are capable of achieving capacity. Such constructions are not realizable in practice, but provide motivation and insight for the design of practical codes, as discussed later. Of particular importance is the insight provided by the analogy between our scheme's decoding problem and the problem of decoding over a multiple-access channel (MAC).

In Section III, we extend our scheme to the Gaussian dirty-paper channel. Our development follows closely in the lines of the binary dirty-paper case. In Section IV we briefly discuss codes for dirty-*tape* channels. These codes serve as an important benchmark for the performance of more complex codes for dirty-paper channels.

In Section V we show how the constructions of Sections II and III can be transformed to produce powerful codes for practical implementation. We discuss the selection of the component codes of our scheme, and the design of encoders and decoders. We also provide simulation results that confirm the effectiveness of our scheme.

In Section VI we propose a framework for the extension of superposition-coding to the general Gel'fand-Pinsker (noncausal side-information) problem. Our discussion is designed primarily to generate interest, while further research is required to produce practical results. Section VII concludes the paper.

II Superposition Coding for Binary Dirty Paper

II.1 Definition

Like the nested-lattices approach of [54], our approach begins with two codes: $\mathcal{C}_0$ and $\mathcal{C}_1$, referred to as the *quantization* code and the *information-bearing* code, respectively. The superposition code $\mathcal{C}$ is defined as $\mathcal{C} = \mathcal{C}_0 + \mathcal{C}_1$, i.e.,

$$\mathcal{C} = \{\mathbf{c} = \mathbf{c}_0 + \mathbf{c}_1 \;\;:\;\; \mathbf{c}_0 \in \mathcal{C}_0,\; \mathbf{c}_1 \in \mathcal{C}_1\} \tag{2}$$

In a random-coding setting (as will be discussed in Section II.2), we construct the quantization code $\mathcal{C}_0$ by random i.i.d selection according to a Bernoulli(1/2) distribution. The information-bearing code $\mathcal{C}_1$ is constructed by random i.i.d selection according to a Bernoulli(q) distribution, where q is a parameter that will be discussed later. The codes are constructed at rates R_0 and R_1, respectively, and block length n. In the practical application of the scheme (Section V.1), the codes will be selected differently, relying on insight provided by the random-coding discussion.

Encoding and decoding for the binary dirty-paper channel proceed as follows.

Encoder: The encoder selects a codeword $\mathbf{c}_1 \in \mathcal{C}_1$, and sends the sequence

$$\mathbf{x} \;\; = \;\; [\mathbf{c}_1 + \mathbf{s}]_{\mathcal{C}_0}$$

where $\mathbf{s}$ is the interference vector, and $[\mathbf{y}]_{\mathcal{C}} \triangleq \mathbf{y} + Q_{\mathcal{C}}(\mathbf{y})$. For any $\mathbf{y} \in \mathbb{F}_2^n$, $Q_{\mathcal{C}}(\mathbf{y})$ is defined by $Q_{\mathcal{C}}(\mathbf{y}) \triangleq \arg\;\; \min_{\mathbf{c}\in\mathcal{C}} d_H(\mathbf{y},\mathbf{c})$.

Defining $\mathbf{c}_0 \triangleq Q_{\mathcal{C}_0}(\mathbf{c}_1 + \mathbf{s})$, we may also write,

$$\mathbf{x} = \mathbf{c}_1 + \mathbf{s} + Q_{\mathcal{C}_0}(\mathbf{c}_1 + \mathbf{s}) = \mathbf{c}_0 + \mathbf{c}_1 + \mathbf{s}$$

The channel outputs the signal

$$\mathbf{y} = \mathbf{x} + \mathbf{s} + \mathbf{z} = \mathbf{c}_0 + \mathbf{c}_1 + \mathbf{z} \tag{3}$$

Decoder: The decoder computes the pair $(\hat{\mathbf{c}}_0, \hat{\mathbf{c}}_1)$ such that $\hat{\mathbf{c}}_0+\hat{\mathbf{c}}_1$ is closest to $\mathbf{y}$. $\hat{\mathbf{c}}_1$ is announced as the decoded codeword.

Note that the encoder begins by selecting a codeword $\mathbf{c}_1 \in \mathcal{C}_1$, and decoding terminates by producing $\hat{\mathbf{c}}_1$. Thus the effective rate of the transmission scheme is the rate R_1 of $\mathcal{C}_1$.

The operation of the decoder is identical to that of a decoder for the additive multiple-access (MAC) channel where two *virtual* users with codebooks $\mathcal{C}_0$ and $\mathcal{C}_1$ send independently selected codewords $\mathbf{c}_0$ and $\mathbf{c}_1$, respectively. This choice of decoder is motivated by the observation that since $\mathbf{s}$ is uniformly distributed over $\mathbb{F}_2^n$ and is independent of $\mathbf{c}_1$, the codeword $\mathbf{c}_0$ is independent of $\mathbf{c}_1$. The analogy with the MAC channel has an important function in the design of practical decoders, that will be discussed in Section V.

It is instructive to consider the above superposition scheme in terms of the random-binning achievability scheme of Gel'fand and Pinsker [29]. In [29], a random codebook $\mathcal{C}$ is constructed, and its codewords are randomly partitioned into bins (subsets) $\{\mathcal{C}_m : m = 1, \ldots, M\}$, where M is the number of distinct messages that may be transmitted. Each message is associated with a bin, and encoding begins by selecting a bin $\mathcal{C}_m$ to match with the transmitted message m. The encoder proceeds to search within $\mathcal{C}_m$ for a codeword $\mathbf{c}$ that is "nearest" (in terms of joint-typicality) to the interference vector $\mathbf{s}$. The transmitted vector $\mathbf{x}$ is a function of $\mathbf{s}$ and $\mathbf{c}$.

The decoder searches the entire code $\mathcal{C}$ for the codeword $\hat{\mathbf{c}}$ that is "nearest" (again, in terms of joint-typicality) to the received $\mathbf{y}$. It then determines the unique bin $\hat{\mathcal{C}}_m$ that contains $\hat{\mathbf{c}}$ and declares the decoded message $\hat{m}$ to be the one matching the bin.

With superposition coding, the codebook $\mathcal{C}$ is given by the superposition code $\mathcal{C}_0 + \mathcal{C}_1$. For each codeword $\mathbf{c}_1 \in \mathcal{C}_1$, the coset code $\mathbf{c}_1 + \mathcal{C}_0 = \{\mathbf{c}_1 + \mathbf{c}_0 \ : \ \mathbf{c}_0 \in \mathcal{C}_0\}$ is equivalent to a bin of [29].

Encoding begins by selecting a codeword $\mathbf{c}_1 \in \mathcal{C}_1$. This selection is equivalent to the selection of the bin $\mathbf{c}_1 + \mathcal{C}_0$. The operation of evaluating $Q_{\mathcal{C}_0}(\mathbf{c}_1 + \mathbf{s})$ can equivalently be presented as a search, over the coset code $\mathbf{c}_1 + \mathcal{C}_0$, for the word $\mathbf{c} = \mathbf{c}_1 + \mathbf{c}_0$ that is nearest to $\mathbf{s}$. The transmitted vector is $\mathbf{x} = \mathbf{c} - \mathbf{s}$ (subtraction being evaluated over $\mathbb{F}_2$, and is identical to addition). Assuming the code $\mathcal{C}_0$ is dense "enough" (a precise analysis of which will be provided in Section II.2 below), the nearest codeword $\mathbf{c}$ will be close to $\mathbf{s}$, and the transmitted difference $\mathbf{x}$ between the two vectors will not violate the Hamming input constraint.

The channel adds $\mathbf{s}$ to $\mathbf{x}$, and thus the result is equivalent to the transmission of $\mathbf{c} = \mathbf{c}_0 + \mathbf{c}_1$.

The channel also adds the unknown noise $\mathbf{z}$, so the resulting output is $\mathbf{y} = \mathbf{c} + \mathbf{z} = \mathbf{c}_0 + \mathbf{c}_1 + \mathbf{z}$. The decoder searches the superposition code $\mathcal{C}$ for the codeword $\hat{\mathbf{c}} = \hat{\mathbf{c}}_0 + \hat{\mathbf{c}}_1$ that is nearest to $\mathbf{y}$. Assuming that $\mathcal{C}$ is not "too" dense, the correct codeword $\mathbf{c}$ will be recovered. Given $\hat{\mathbf{c}} = \hat{\mathbf{c}}_0 + \hat{\mathbf{c}}_1$, $\hat{\mathbf{c}}_1$ identifies the bin $\hat{\mathbf{c}}_1 + \mathcal{C}_0$ to which $\hat{\mathbf{c}}$ belongs.

It is interesting to note on the case of nonuniform interference S, i.e. S here is distributed as Bernoulli(s) where $s \neq 1/2$. This is an instance of the general noncausal side-information channel whose capacity was evaluated by Gel'fand and Pinsker [29] and is given by

$$C = \sup_{p(u|s)} \{I(U;Y) - I(U;S)\} \tag{4}$$

where U is an auxiliary random variable with conditional distribution $p(u|s)$ and X is a deterministic function of S and U, $X = f(S,U)$. In our case of nonuniform binary dirty paper, the capacity achieving $p(u|s)$ and $f(s,u)$ are unknown. However, consider the case $U = S + X$, where X is distributed as Bernoulli(W). This choice of U is interesting because it produces an achievable rate $I(U;Y) - I(U;S)$ that is greater than the capacity in the uniform interference case (the expression for which will be given in Section II.2). But with this choice, U is not uniformly distributed in $\{0,1\}$. Thus, the capacity-achieving scheme prescribes bins that are different to the ones constructed in this section. This problem of generating codewords according to nonuniform distributions was considered in various contexts (see e.g. [7] which discusses the Slepian-Wolf problem, [51] which discusses binning in a theoretical context and [35] in the context of wideband Gel'fand-Pinsker channels). An approach that is based on superposition coding is proposed in Section VI.

II.2 Random Coding Analysis

We now provide a precise analysis of superposition coding using randomly generated codes. In this section we assume an encoder-decoder pair that uses minimum-distance quantization and maximum-likelihood decoding. As noted in Section I, when unstructured codes (like the above discussed random codes) are applied, such an encoder-decoder pair is not realizable in practice. The same applies to our development in Section III.2, where we will discuss superposition coding for the Gaussian dirty-paper channel in a random-coding setting. In Section V we will discuss the implementation of our approach in a practical setting.

Our analysis in this section parallels that of Erez, Shamai and Zamir [21, 54] with nested lattices. They proved that the nested-lattices approach can be used to enable reliable transmission at rates arbitrarily close to the dirty-paper capacity. However, their analysis, like ours, assumed optimal lattices and an encoder-decoder pair that are not realizable in practice, at rates approaching channel capacity. The main purpose of their discussion was to motivate the interest in the nested-lattices approach.

Similarly, an important objective of the developments in this section and in Section III.2, is to show that there is no loss of optimality that is necessarily incurred by parting with the highly structured lattice constructions of [54]. The developments thus provides motivation for the development of Section V, where codes that are unsuitable for constructing lattices are successfully applied to the dirty-paper problem[1]. An important additional objective is to provide important insight that will guide the practical design of codes in Section V.

We construct $\mathcal{C}_0$ and $\mathcal{C}_1$ as discussed in Section II.1. We define an *encoder error* as the event that the transmitted $\mathbf{x}$ violates the codeword constraint, i.e. $d_H(\mathbf{x}, \mathbf{0}) > nW$. A *decoder error* occurs if $\widehat{\mathbf{c}}_1 \neq \mathbf{c}_1$. The error probability of the scheme is the probability of the union of these two events. We begin with an analysis of the probability of an encoder error:

Lemma 1 *The probability of an encoder error approaches zero with n, if R_0 satisfies*

$$R_0 > 1 - h(W) \tag{5}$$

where $h(\cdot)$ denotes the binary entropy function[2]*.*

The proof of this lemma relies on the following observation: Since $\mathbf{s}$ is uniformly distributed over $\mathbb{F}_2^n$ and independent of $\mathbf{c}_1$, then also $\mathbf{c}_1 + \mathbf{s}$ is uniformly distributed over $\mathbb{F}_2^n$. $R(W) = 1 - h(W)$ is the binary Hamming-weight rate-distortion function [18]. Hence, by choosing $\mathcal{C}_0$ in the ensemble of random codes with a rate that is greater than $1 - h(W)$, we ensure that the encoder will find a codeword at distance at most nW from $\mathbf{c}_1 + \mathbf{s}$ with an ensemble average probability that approaches 1 as $n \to \infty$.

Now consider the probability of a decoder error. Given the formal analogy with the MAC (Section II.1), we can conclude that the probability of a decoder error vanishes with n if the rate pair (R_0, R_1) lies within the capacity region of the MAC. This condition is sufficient, i.e., it yields an achievability result. In fact, the decoder is only interested in reliable decoding of $\mathcal{C}_1$, while in the MAC both $\mathcal{C}_1$ and $\mathcal{C}_0$ must be reliably decoded. However, as we shall see in the following, for an appropriate choice of the parameter q governing the ensemble of $\mathcal{C}_1$, the additional condition

[1] This point is further explained in Section III.3.

[2] Note that throughout this paper, we measure rate in bits, and hence the base of the log function is always 2, including in the expression for $h(\cdot)$.

of decoding reliably also $\mathcal{C}_0$ incurs no loss of optimality. We therefore proceed to the following lemma,

Lemma 2 *Consider the MAC*

$$Y = X_0 + X_1 + Z \tag{6}$$

where all variables are over $\mathbb{F}_2$, where Z is binary noise distributed as Bernoulli(p), and where user 1 is subject to a Hamming weight input constraint q. The capacity region is given by all pairs $(R_0, R_1) \in \mathbb{R}_+^2$ satisfying

$$\begin{aligned} R_1 &\leq h(p \otimes q) - h(p) \\ R_1 + R_0 &\leq 1 - h(p) \end{aligned} \tag{7}$$

where $p \otimes q \triangleq p(1-q) + q(1-p)$.

The proof of this lemma is provided in Appendix A.

Finally, we combine Lemmas 1 and 2 to obtain:

Theorem 1 *If R_0 and R_1 satisfy (5) and (7) then the average probability of error approaches zero with the block length n.*

Note that the code $\mathcal{C}_0$ has dual requirements to be both a good quantization code (to enable effective encoding) and a good channel code (for effective decoding). Since the averaged probabilities of encoder and decoder errors both approach zero with n, we are ensured that most codes in the random-coding ensemble are good in both senses.

Fig. 1 presents the capacity region prescribed by Theorem 1. A dashed line marks the constraint imposed by (5). Transmission is possible at any point that is within the MAC capacity region and is above the dashed line. This is the achievable region of superposition coding.

The MAC capacity region is a function of parameters q and p. To show that superposition coding can achieve capacity, we must show that we can select q such that this region contains a point (R_0, R_1) where R_1 (the effective rate) equals the capacity C of the dirty-paper channel, (1).

The expression for C (see Pradhan *et al*. [39] and Barron *et al*. [3]) is given by,

$$C = \begin{cases} h(W) - h(p) & \text{for } W_c \leq W \leq 1/2 \\ \alpha W & \text{for } 0 \leq W \leq W_c \end{cases} \tag{8}$$

where $W_c = 1 - 2^{-h(p)}$ and $\alpha = \log((1-W_c)/W_c)$. Note that in the range $0 \leq W \leq W_c$, capacity is achieved by time-sharing, with a duty-cycle $\theta = W/W_c$, of the standard random-binning scheme for Hamming distortion equal to W_c and "silence" (zero rate and zero Hamming distortion)[3].

We select q such that $q \otimes p \geq W$. For any $W \leq 1/2$ there exists $q^\star \in [0, 1/2]$ such that this condition holds for all $q \geq q^\star$. With this selection of q, we obtain that the rate pair $R_0 = 1 - h(W)$, $R_1 = h(W) - h(p)$ falls within the MAC capacity region. Thus, pairs of quantization and information-bearing codes $\mathcal{C}_0$ and $\mathcal{C}_1$ can be found such that the resulting scheme achieves rates arbitrarily close to $h(W) - h(p)$, as desired (capacity in the range $0 < W < W_c$ is again obtained by time-sharing).

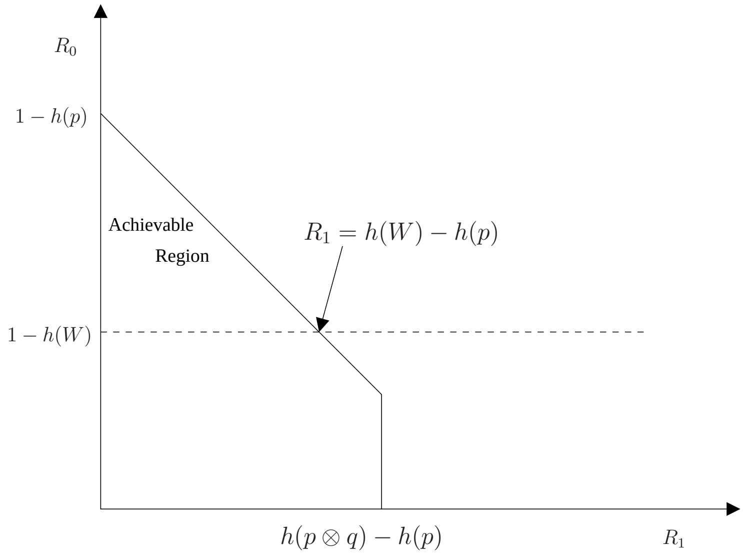

Figure 1: Capacity region of the corresponding binary MAC channel.

III Superposition Coding for Gaussian Dirty Paper

III.1 Definition

We now extend superposition coding to the Gaussian dirty paper problem. We again consider two codes, a quantization code $\mathcal{C}_0$ and an information-bearing code $\mathcal{C}_1$. The superposition code is defined as $\mathcal{C} = \mathcal{C}_0 + \mathcal{C}_1 \bmod A$ (addition being the standard addition over the real-number field), where the operation $\bmod A$ is applied componentwise as follows: Given a scalar x, $x \bmod A \triangleq x - Q_A(x)$ such that $Q_A(x)$ is the nearest multiple of A to x. The dynamic range of x is thus reduced to $[-A/2, A/2]$.

[3] Unlike the case of Gaussian channel studied by Costa [17] and examined in Section III, the binary dirty paper capacity is strictly less than the capacity if interference was not present, which is given by $C = h(p \otimes W) - h(p)$.

The modulo-A operation is borrowed from the construction-A approach to generating lattices from linear codes. Its effect can be equivalently modelled as the tessellation of the entire space $\mathbb{R}^n$ with replicas of the n-dimensional cube $[-A/2, A/2]^n$. Note that it must not be confused with the modulo-lattice operation of nested lattices scheme of [54], which serves a different purpose.

In a random-coding setting (as will be discussed in Section III.2), we construct the quantization code $\mathcal{C}_0$ by random i.i.d selection according to a uniform distribution in the range $[-A/2, A/2]$. The information-bearing code $\mathcal{C}_1$ is constructed by random i.i.d selection according to a zero-mean distribution with a variance Q (Q is a parameter that will be determined later). The exact distribution will be discussed in Appendix B and approaches a Gaussian distribution as $A \to \infty$. $\mathcal{C}_0$ and $\mathcal{C}_1$ have rates R_0 and R_1, respectively, and block length n.

Encoder: The encoder selects a codeword $\mathbf{c}_1 \in \mathcal{C}_1$, and sends the sequence:

$$\mathbf{x} \;=\; [\alpha\mathbf{s} + \mathbf{d} - \mathbf{c}_1 \bmod A]_{\mathcal{C}_0} \bmod A$$

A and α are arbitrary constants that will be discussed later. $\mathbf{d}$ is a randomly selected *dither* signal, borrowed from the nested lattices approach of [54]. However, unlike [54], the elements of the dither are defined to be uniformly i.i.d in the range $[-A/2, A/2]$. This dither corresponds to the dither of Philosof *et al.* [38].

$[\boldsymbol{\xi}]_{\mathcal{C}_0} \triangleq Q_{\mathcal{C}_0}(\boldsymbol{\xi}) - \boldsymbol{\xi}$, $Q_{\mathcal{C}_0}(\boldsymbol{\xi})$ being the codeword of $\mathcal{C}_0$ that is closest to $\boldsymbol{\xi}$ assuming a *modulo A distance metric.* The mod A distance between two vectors $\mathbf{x}$ and $\mathbf{y}$ is given by

$$\|\mathbf{y} - \mathbf{x}\|_A^2 \triangleq \sum_{i=1}^{n} (y_i - x_i \bmod A)^2$$

We thus obtain,

$$\begin{aligned} \mathbf{x} &= [Q_{\mathcal{C}_0}(\alpha\mathbf{s} + \mathbf{d} - \mathbf{c}_1 \bmod A) - (\alpha\mathbf{s} + \mathbf{d} - \mathbf{c}_1)] \bmod A \\ &= \mathbf{c}_0 + \mathbf{c}_1 - \alpha\mathbf{s} - \mathbf{d} \bmod A \end{aligned}$$

where $\mathbf{c}_0 \triangleq Q_{\mathcal{C}_0}(\alpha\mathbf{s} + \mathbf{d} - \mathbf{c}_1 \bmod A)$.

Our choice of $\mathcal{C}_0$ will be shown in Section III.2 to be capable of quantization with a mean square distortion P_X, assuming mod A distance. Thus $\mathbf{x}$ is guaranteed to satisfy the power constraint, $1/n \cdot \sum_{i=1}^n x_i^2 \le P_X$.

The received signal is

$$\mathbf{y} = \mathbf{x} + \mathbf{s} + \mathbf{z}$$

Decoder: The decoder computes

$$\begin{aligned}\hat{\mathbf{y}} &\triangleq \alpha\mathbf{y} + \mathbf{d} \bmod A \\ &= \mathbf{c}_0 + \mathbf{c}_1 - (1-\alpha)\mathbf{x} + \alpha\mathbf{z} \bmod A \\ &= \mathbf{c}_0 + \mathbf{c}_1 + \hat{\mathbf{z}} \bmod A \end{aligned} \tag{9}$$

where the effective noise $\hat{\mathbf{z}}$ is $\hat{\mathbf{z}} = -(1-\alpha)\mathbf{x} + \alpha\mathbf{z}$. The decoder evaluates the pair $(\hat{\mathbf{c}}_0, \hat{\mathbf{c}}_1)$ such that $\hat{\mathbf{c}}_0 + \hat{\mathbf{c}}_1$ is closest to $\hat{\mathbf{y}}$, assuming $\bmod A$ distance. $\hat{\mathbf{c}}_1$ is announced as the decoded codeword.

¿From the above construction, it is clear that $\mathbf{c}_0$ and $\mathbf{c}_1$ are independent, implying the analogy to the MAC channel. However, the effective noise $\hat{\mathbf{z}}$ contains a "self-noise" element $\mathbf{x}$ that, for particular choices of the codes $\mathcal{C}_0$ and $\mathcal{C}_1$, is not independent of $\mathbf{c}_0$ and $\mathbf{c}_1$, undermining an assumption of the Gaussian MAC model. In Section III.2 we show that under a random-coding assumption, the MAC model is valid and a decoder designed for the MAC channel is capable of achieving capacity. In section V we show that the analogy is valuable in a practical setting as well.

III.2 Random Coding Analysis

As in our analysis of superposition coding for binary dirty-paper, we now analyze the scheme for the Gaussian channel using randomly generated codes. The purpose of this section is once again to provide motivation and guidelines to a construction using practical codes, which is discussed in Section V.

We assume that $\mathcal{C}_0$ and $\mathcal{C}_1$ are constructed as in Section III.1. In Theorem 2 (below) we consider the probability of error. The proof, which is provided in Appendix B, employs an encoder/decoder pair that rely on joint-typicality instead of the minimum-distance metrics that were used in Section III.1. However, it is important to note that the decoder is similar to the MAC decoder that is used in information-theoretic proofs [18] (the only difference being that it tests for strong-typicality instead of weak-typicality). The decoder evaluates $\hat{\mathbf{y}} \triangleq \alpha\mathbf{y}$ and decodes $\hat{\mathbf{c}}_0$ and $\hat{\mathbf{c}}_1$ assuming a virtual MAC channel

$$\hat{Y} = U_0 + U_1 + \hat{Z} \bmod A \tag{10}$$

where U_0 and U_1 are virtual, independent users and $\hat{Z}$ is independent zero-mean Gaussian noise with variance $P_{\hat{Z}} = (1-\alpha)^2 P_X + \alpha^2 P_Z$. Thus, the theorem reinforces the analogy to the MAC channel. In the sequel, we further assume that the MAC channel is characterized by a power constraint Q on user 1, paralleling our development in Section II.2.

Theorem 2 *Given the above selection of $\mathcal{C}_0$ and $\mathcal{C}_1$, an encoder/decoder pair can be designed such that the following holds:*

1. *The probability of an encoder error (i.e., a violation of the power constraint) approaches zero with the block length n, if R_0 satisfies*

$$R_0 > \log A - \frac{1}{2}\log(2\pi e P_X) + \delta_1 \tag{11}$$

where $\delta_1 \to 0$ as $A \to \infty$.

2. *The probability of a decoder error approaches zero with n if the pair (R_0, R_1) lies in the interior of the capacity region of the MAC channel, whose transition probabilities are determined by (10), and where user 1 is subject to the power constraint Q. This capacity region is given by,*

$$R_0 + R_1 \leq \log A - \frac{1}{2}\log(2\pi e P_{\hat{Z}}) + \delta_2 \tag{12}$$

$$R_1 \leq \frac{1}{2}\log\left(1 + \frac{Q}{P_{\hat{Z}}}\right) + \delta_3 \tag{13}$$

where $\delta_2, \delta_3 \to 0$ as $A \to \infty$.

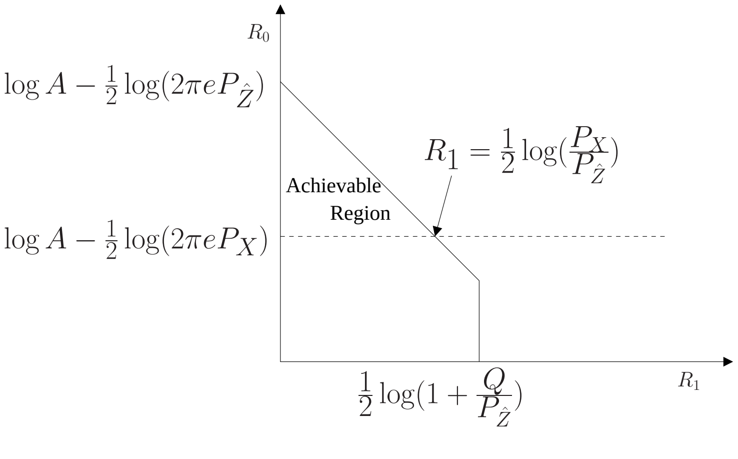
R_0
$\log A - \frac{1}{2}\log(2\pi e P_{\hat{Z}})$
$R_1 = \frac{1}{2}\log(\frac{P_X}{P_{\hat{Z}}})$
Achievable
Region
$\log A - \frac{1}{2}\log(2\pi e P_X)$
$\frac{1}{2}\log(1 + \frac{Q}{P_{\hat{Z}}})$
R_1

Figure 2: Capacity region of the corresponding AWGN MAC channel.

The proof of the theorem is provided in Appendix B. Fig. 2 is similar to Fig. 1 and presents the capacity region prescribed by Theorem 2. A dashed line marks the constraint imposed by equation (11) (neglecting elements δ_1,δ_2 and δ_3). Transmission is possible at any point that is within the MAC capacity region and is above the dashed line.

In a manner similar to the binary dirty paper case, the MAC capacity region is a function not only of the power constraint P_X and the noise variance P_Z but also of parameters α and Q. As in Section II.2, our desire is to select α and Q such that a point (R_0, R_1) where $R_1 = 1/2 \log(1 + P_X/P_Z)$ falls within the achievable region.

Combining equations (11) and (12), we obtain,

$$R_1 < \frac{1}{2} \log \frac{P_X}{P_{\hat{Z}}} + (\delta_2 - \delta_1)$$

As in the analysis of Costa [17] and of Erez, Shamai and Zamir [21], the variance $P_{\hat{Z}}$ of the effective noise $\hat{Z}$ is minimized by a selection of $\alpha = P_X/(P_X + P_Z)$, producing $P_{\hat{Z}} = P_X P_Z/(P_X + P_Z)$. With this selection of α,

$$R_1 < \frac{1}{2} \log \frac{P_X(P_X + P_Z)}{P_X P_Z} + (\delta_2 - \delta_1) = \frac{1}{2} \log \left(1 + \frac{P_X}{P_Z}\right) + (\delta_2 - \delta_1)$$

which is the no-interference AWGN channel capacity. To avoid being bounded away from capacity by (13), we select $Q \geq P_X^2/(P_X + P_Z)$ and obtain

$$\frac{1}{2} \log \left(1 + \frac{Q}{P_{\hat{Z}}}\right) = \frac{1}{2} \log \left(1 + \frac{Q(P_X + P_Z)}{P_X P_Z}\right) \geq \frac{1}{2} \log \left(1 + \frac{P_X}{P_Z}\right)$$

For the above selections of α and Q, the no-interference AWGN channel capacity is achieved at the point $R_1 = 1/2 \log(1 + P_X/P_Z)$ and $R_0 = \log A - 1/2 \log(2\pi e P_X)$.

III.3 Comparison with Nested Lattices

The nested-lattices scheme, as proposed by Zamir, Shamai and Erez [54], is an instance of a more general scheme that was proposed by the same authors [21][Section 4]. The scheme of [21] uses a lattice Λ and a code V that is uniformly distributed within the lattice's fundamental Voronoi region. Λ is typically created by construction-A from a code $\mathcal{C}_\Lambda$ (see Philosof *et al.* [37, 38] and Erez and ten Brink [23]). Letting $\mathcal{C}_0 = \mathcal{C}_\Lambda$ and $\mathcal{C}_1 = V$, the scheme coincides with superposition-coding. Superposition-coding, however, is not restricted to lattices Λ and codes V as described in [21]. For example, for construction-A to produce a lattice, $\mathcal{C}_\Lambda$ needs to be linear under modulo-q arithmetic (q being some positive integer). With superposition coding, the equivalent $\mathcal{C}_0$ is allowed to be nonlinear. In the practical implementation of superposition coding (Section V-B), we make use of this extra degree of freedom and design $\mathcal{C}_0$ to be a trellis code, which is not linear under modulo-q arithmetic. Furthermore, in [21], the information-bearing code V is found to be capacity-achieving if designed to be Gaussian distributed with a variance of at least $P_X^2/(P_X + P_Z)$, coinciding with our development of Section III.2. However, the focus in [21] is on a code V whose codewords are uniformly distributed within the fundamental Voronoi cell of Λ, roughly corresponding to i.i.d selection according to a Gaussian distribution with a variance Q equal to P_X. This choice is approximated by nested-lattices (which are an application of the more general scheme of [21]). In Section V.2 we will show that in a practical implementation of superposition coding, a choice of $Q = P_X^2/(P_X + P_Z)$ (i.e. $Q < P_X$) is advantageous. Therefore, the codes designed for superposition coding will be fundamentally different to ones designed for nested lattices.

The formulation of the superposition-coding scheme in Sections III.1 and III.2 provides a different perspective on the decoding problem than the one provided in [54] and [21]. The equivalence with the MAC decoding problem will enable the application, in Section V, of a powerful iterative decoder that was developed for the MAC channel. Furthermore, it provides important insight, and will enable the observation in Section V that the decoder works best when the code $\mathcal{C}_1$ is designed with the above mentioned value of Q. The formulation of the problem in this way makes the design of the decoder considerably simpler and more systematic task in comparison with [23].

The theoretical results of Section III.2 are valid for A approaching infinity. This requirement appears to be common also to nested lattices, where A is the argument for construction-A (i.e., $\Lambda = C_\Lambda + A \cdot \mathbb{Z}^n$). For nested lattices coding to achieve capacity, the lattices need to be "good" for both source and channel coding [54]. Good nested lattices were presented in [22]. The lattices were generated using construction-A, and exhibit values of A that approach infinity[4]. However, further research is necessary in order to establish if this requirement is indeed inherent to any method that produces good lattices.

In Section III.2, the distribution of the code $\mathcal{C}_1$ was designed to be approximately Gaussian. Thus, strictly speaking, in a practical implementation the empirical distributions of the codewords of $\mathcal{C}_1$ need to be "shaped" to approximate a Gaussian distribution. With standard Gaussian no-interference channels, experience shows that this issue is of minor importance when the channel's signal-to-noise ratio (SNR) is low (see e.g. [26]). Most of the work on nested-lattices has focused on the low-SNR regime. In Section V.2 we have followed this example, for reasons that will be explained in that section. Therefore, building on the experience available with no-interference channels, the codes constructed in Section V.2 do not employ shaping.

[4] The construction of [22] examines lattices with a constant $A = 1$. However, it is assumed that the lattices are to be scaled to fit any practical problem setting. The effective radius r_Λ^{effec} of the lattices is held (approximately) constant while the block length n is taken to infinity. This implies that the effective power approaches zero, requiring a scaling factor that approaches infinity.

We now, however, briefly discuss the issue of shaping in the context of our comparison with the nested-lattices approach. With superposition coding, shaping of $\mathcal{C}_1$ could be achieved using various standard methods that are available in the literature (a discussion and relevant references are available in [26]). Additional simple and effective techniques, however, are provided in [4, 5] using quantization mapping and in [27, 44] using nonuniform spaced signal constellations.

To examine the issue of shaping with nested-lattices, we begin with the above mentioned generalized scheme of [21]. With this scheme, the information-bearing code is distributed uniformly within the fundamental Voronoi cell of the lattice Λ. If this Voronoi cell is approximately spherical, it yields an information bearing code whose words have empirical distributions that are approximately Gaussian. Erez *et al.* [22] showed that there exist optimal lattices, for which this indeed is the case. With nested-lattices, the information-bearing code is designed to approximate uniform-distribution within the fundamental Voronoi cell. Thus, like superposition-coding, the information-bearing codewords' approximate a Gaussian empirical distributions.

Lastly, an advantage of superposition coding over the coset dilution method of [37, 38, 23] is a simpler encoder design. With coset dilution, encoding involves selection of a coset leader to represent the transmitted codeword. This step is not required by superposition coding.

IV Dirty Tape Channels

We now briefly discuss the Gaussian and binary dirty-*tape* channels, where the interference S is only known causally. Dirty tape coding schemes can be used as low-complexity solutions for dirty paper problems, by simply ignoring the noncausal portion of the known interference. Ignoring part of the known data is clearly suboptimal, but nonetheless provides an important benchmark for the performance of the more complex dirty paper schemes. For a dirty paper scheme to be interesting, it must surpass the performance of existing strategies for the associated dirty tape channel.

The Gaussian dirty-tape channel was examined by Erez, Shamai and Zamir [21]. They developed a simple strategy based on an "inflated-lattices" scheme for this channel. They also developed an information-theoretic expression for the optimal performance that can be achieved by their scheme (using random codes and maximum-likelihood decoding), at the asymptotic case of strong interference. At the limit of high SNR, this expression was shown to approach the capacity of the Gaussian dirty-tape channel. Caire *et al.* [9] and Erez and ten Brink [23] applied this scheme to achieve reliable transmission within 0.4dB of the "inflated lattices" information-theoretic limit, at a rate of approximately 0.667 bits per complex dimension.

In our discussion below, we show that like its Gaussian counterpart, the binary dirty-tape problem enjoys a simple practical coding scheme, which uses existing channel codes to approach capacity. We begin with the following lemma:

Lemma 3 *Consider the binary dirty tape channel, with Hamming-weight input constraint W and channel crossover probability p, and where the interference S is distributed as Bernoulli(1/2). The capacity of this channel is given by*

$$C = 2W(1 - h(p)) \tag{14}$$

Proof. The capacity formula for the general causal side-information channel was found by Shannon [43] and is given by

$$C = \sup_{P_U} I(U;Y) \tag{15}$$

The auxiliary variable U in (15) takes on values in the set $\mathcal{U}$ of memoryless functions (or "strategies") mapping the state S into the input X. Since S and X are binary, the set of possible strategies is $\mathcal{U} = \{\mathbf{0}, \mathbf{1}, \mathbf{id}, \mathbf{not}\}$, i.e., the identically *zero*, identically *one* functions, identity and negation, respectively.

Due to the concavity of capacity as a function of the input constraint W and to the fact that the input symbol $\mathbf{0}$ has associated cost zero, we have [49]

$$C \leq W \sup_{u \in \mathcal{U}} \frac{D(P_{Y|U=u} \| P_{Y|U=\mathbf{0}})}{E[d_H(u(S), 0)]} = 2W(1 - h(p)) \tag{16}$$

where the supremization is achieved by either $u = \mathbf{id}$ or by $u = \mathbf{not}$. Note that a similar approach, in the context of the Gaussian dirty-tape channel, was suggested by Hoesli [30]. In order to show that (16) is indeed the capacity, we must find an input probability assignment P_U such that $I(U;Y)$ equals (16). This is indeed achieved by $P_U(\mathbf{id}) = P_U(\mathbf{not}) = W$, $P_U(\mathbf{0}) = 1 - 2W$ and $P_U(\mathbf{1}) = 0$. □

A naive approach to designing dirty tape codes follows Shannon's achievability proof, and uses codes over the ternary alphabet $\{\mathbf{id}, \mathbf{not}, \mathbf{0}\}$. However, a simpler approach relies on the observation that the capacity function (14) is linear in W. Any point $(W, C(W))$ can be achieved by time-sharing between $(0,0)$ and $(1/2, 1 - h(p))$. The point $(0,0)$ corresponds to "silence". The point $(1/2, 1 - h(p))$ is easily achieved by using an unconstrained code for the BSC channel and cancelling the interference $\mathbf{s}$ by simply subtracting it. Thus, the binary dirty tape problem is reduced to conventional coding for the BSC with parameter p and multiplexing with a fixed zero symbol. Furthermore, it can be shown that when random-codes are used, this time-sharing approach renders a superior error-exponent in comparison to the naive approach [12].

Time-sharing is also preferable from a practical-implementation viewpoint. Binary LDPC codes are known [40] to approach the BSC capacity remarkably close. Thus, building on them, we can approach the binary dirty-tape capacity equally close.

V Design of Codes for Practical Implementation

The random-coding analysis of Sections II.2 and III.2 indicates the ability of the superposition coding scheme to achieve the capacities of the binary and Gaussian dirty-paper channels. However, the development in those sections assumes randomly generated codes and maximum-likelihood decoding. Such constructions cannot be applied in practice. In this section we discuss codes, encoders and decoders that allow practical implementation.

Our designs are guided by the analogy to the MAC channel that was established in Sections II.1 and III.1 and by the random-coding analysis of Sections II.2 and III.2.

V.1 Binary Dirty-Paper

We begin by specifying the selection of the quantization code $\mathcal{C}_0$ and the information-bearing code $\mathcal{C}_1$. This selection is determined by the requirements imposed on the encoder and the decoder.

The superposition-coding decoder, as presented in Section II.1, performs an operation that is identical to that of a decoder for the MAC channel. It would therefore appear to make sense to select $\mathcal{C}_0$ and $\mathcal{C}_1$ as the best known codes for the MAC channel. However, the encoder uses $\mathcal{C}_0$ to quantize $\mathbf{c}_1 + \mathbf{s}$. This produces a requirement that rules out LDPC and turbo-like codes for the role of $\mathcal{C}_0$, because at the present time no good quantization algorithm is known for these codes. We therefore make the next best choice, and design $\mathcal{C}_0$ as a convolutional code, with which we may use the Viterbi algorithm. We do select $\mathcal{C}_1$, however, to be an LDPC code.

We now turn to the design of the encoder and the decoder. The discussion in Section II.1 and the availability of the Viterbi algorithm prescribe a simple and straightforward implementation for the encoder. Building on the equivalence with the MAC decoding problem, we have turned to the existing literature to seek decoders for the MAC channel.

One possibility for the MAC decoder uses *successive cancellation* [53]. Namely, given the channel output $\mathbf{y}$ we first decode $\hat{\mathbf{c}}_0$ by treating $\hat{\mathbf{z}} + \mathbf{c}_1$ as noise and next decode $\hat{\mathbf{c}}_1$ from $\mathbf{y} - \hat{\mathbf{c}}_0$. However, achieving capacity with successive cancellation is only possible at vertex-points of the

MAC capacity region.

The capacity region of the MAC channel that is observed by the decoder, was derived in Lemma 2 and is depicted in Fig. 1. This region is a function of the input constraint q, which in Section II.1 also determined the random generation of $\mathcal{C}_1$. It is important to observe that unlike the standard MAC problem, where the input constraints are given, in our scheme q is a parameter that may be modified freely. We have seen in Section II.2 that any choice of $q \geq q^\star$ is sufficient in order that a point $(R_0, R_1 = C)$ (C is the binary dirty paper capacity) fall within this capacity region. A choice of $q = q^\star$ further results in a capacity region where $(R_0, R_1 = C)$ is achieved exactly at a vertex point. This choice thus enables the application of successive-cancellation.

Strictly speaking, decoding by successive-cancellation is sufficient to approach capacity on any MAC. However, this requires that the codes $\mathcal{C}_0$ and $\mathcal{C}_1$ be selected as capacity-approaching channel codes. As noted above, we are restricted in our selection of $\mathcal{C}_0$, and have selected it to be a trellis code rather than a capacity-approaching LDPC code. This substantially diminishes the performance of successive-cancellation. An alternative choice of MAC decoder (see Boutros and Caire [8] and references within), uses joint iterative decoding to decode $\mathcal{C}_0$ and $\mathcal{C}_1$. The joint-decoder iteratively transfers soft decisions back and forth between decoders for $\mathcal{C}_0$ and $\mathcal{C}_1$. It is thus a generalization of the successive-cancellation decoder, which transferred a single hard decision from the decoder of $\mathcal{C}_0$ to that of $\mathcal{C}_1$.

In [13, 10, 42], iterative multiuser decoding based on belief-propagation was examined [5]. The analysis was based on simulation results and density evolution of specific code ensembles. It was observed that when single-user codes are used (i.e., codes that were independently optimized for single-user channels), decoding performs close to the theoretical limits only if it is applied in the vicinity of a vertex-point of the MAC capacity region[6]. At other points, the first decoding iteration of $\mathcal{C}_0$ (before any soft information was received from $\mathcal{C}_1$) cannot produce meaningful information and decoding cannot bootstrap.

Note that since we used a suboptimal quantization code ($\mathcal{C}_0$) rather than a randomly-generated code, its Voronoi cells are *not* approximately spherical, and there is a substantial probability that the addition of noise and a codeword $\mathbf{c}_1$ would exceed the boundaries of the Voronoi cell of $\mathbf{c}_0$. This, in the context of successive cancellation could lead to poor performance. However, joint-iterative decoding is more robust to this problem. With joint-iterative decoding, all we need is

[5] Their results were focused on the Gaussian MAC channel, however our experiments indicate that they are equally valid with the binary MAC.

[6] Note, however, that this might not be the case if the user codes are jointly optimized for the belief-propagation decoder (see [1]).

that the first decoding of $\mathcal{C}_0$ to produce meaningful information (for the process to bootstrap), not a perfect decoding. Our simulations indicate that this indeed occurs.

Thus, for both successive cancellation and joint iterative decoding operating close to a vertex of the capacity region is of fundamental importance. This imposes a constraint on our selection of the code $\mathcal{C}_1$. Namely, the code needs to resemble a code randomly generated according to a Bernoulli(q) distribution. Roughly speaking, this is equivalent to requiring that the average codeword weight be $q \cdot n$. This prevents us from selecting $\mathcal{C}_1$ as a classic LDPC code since this would make the average codeword weight $1/2 \cdot n$.

To design $\mathcal{C}_1$, we used GQC-LDPC codes (Galois-field quantized-coset LDPC) as suggested in [4]. GQC-LDPC codes are obtained by taking LDPC codewords defined over an enlarged, nonbinary alphabet (for example $\mathbb{F}_4$) and mapping them to the channel alphabet. It was shown [4] that under maximum-likelihood decoding, GQC-LDPC codes are capable of achieving the capacity of any discrete-memoryless channel. Iterative decoding of GQC-LDPC codes is discussed in [4, 5].

In the simulations elaborated below, we considered the case of $q = 1/4$. We used GQC-LDPC codes constructed from LDPC codes defined over $\mathbb{F}_4$. We mapped one of the $\mathbb{F}_4$ alphabet symbols to the binary digit 1 and the rest to 0. Since the $\mathbb{F}_4$ symbols are approximately uniformly distributed in each $\mathbb{F}_4$ LDPC codeword (as shown in [4]), we obtained that the weight of each resulting GQC-LDPC codeword (produced by mapping to the binary alphabet) was approximately $1/4 \cdot n$, as desired.

In our simulations, we experimented with the constraint $\mathrm{W} = 0.3$ over a dirty paper channel with noise $\mathrm{p} = 0.1$ and uniformly distributed known interference S. The capacity of this channel is 0.41.

For the quantization code $\mathcal{C}_0$, we selected a convolutional code, as discussed above. The selection of the code parameters was guided by the input constraint, i.e., it was required to quantize a uniformly distributed source $(\mathbf{s} + \mathbf{c}_1)$ with distortion less than W. The rate-distortion function at $W = 0.3$ is 0.1187 and hence we sought low-rate convolutional codes at approximately this rate. We selected a code at rate 1/8 and constraint length $K = 11$ (i.e., $2^{10} = 1024$ states) provided in [28]. The code's generator polynomials are given by the octal digits $(2565, 2747, 3311, 3723, 2373, 2675, 3271, 2473)$. Simulation results indicate that the code is capable of quantization to a distortion of $W = 0.302$.

For the information-bearing code $\mathcal{C}_1$ we selected a GQC-LDPC code with an average codeword

Problem	Rate
Dirty tape capacity	0.318
Best dirty paper results	0.36
Dirty paper capacity	0.41

Table 1: Results for the binary dirty paper problem with $W = 0.3$ and $p = 0.1$.

weight of $q = 0.25$. This q satisfies $q \otimes p = W$, and thus equals $q^{\star}$ as defined in Section II.2 [7].

We used a joint iterative belief-propagation decoder to decode $\mathcal{C}_0$ and $\mathcal{C}_1$, using a BCJR decoder for $\mathcal{C}_0$ and a belief-propagation decoder for $\mathcal{C}_1$. A detailed description of this decoder is provided in Appendix D. The decoders alternated at a rate of at least 10 LDPC iterations per BCJR iteration (the ratio was changed throughout the decoding process).

We designed the edge distribution for $\mathcal{C}_1$ using a method that is based on Chung *et al.* [15]. The method requires "singleton" error probabilities, which in [15] are produced by density evolution. In our work we have instead used Monte Carlo simulations, because the analytic computation of the densities was not feasible. The edge distributions we obtained are $\lambda(2,3,4,5,6,7,8,11) = (0.49505,\ 0.26112,\ 0.0085769,\ 0.027705,\ 0.027272,\ 0.11694,\ 0.054428,\ 0.0089079)$ and $\rho(3,4) = (0.65,\ 0.35)$. Note that in the construction of $\mathcal{C}_1$ according to these edge distributions, we employed a non-random approach (discussed in Appendix D.3) which produced a substantial performance gain.

Simulation results for the above dirty paper scheme of rate 0.36 indicate a bit error rate of approximately $2.8 \cdot 10^{-5}$ at a block length of 10^5 (100 simulations). Decoding typically took 150-200 LDPC iterations and 10-15 BCJR iterations to converge.

For a channel characterized by the same W and p, we also experimented with a decoder that uses successive cancellation instead of joint iterative decoding. The details of our design and simulations are provided in Appendix E. Our best codes, optimized for this decoder, were capable of reliable transmission at a rate of only 0.25 bits per channel use. Thus, successive cancellation renders performance that is substantially inferior to that of joint iterative decoding.

As noted in Section IV, it is worthwhile to compare these results to the binary dirty *tape* capacity. The capacity of the dirty tape channel with the same parameters W and p as in our above discussed dirty paper scenario is 0.318. This capacity is thus surpassed by the above described dirty paper scheme of rate 0.36. The different rates are summarized in Table 1.

[7] We experimented with other values of q, and our results indicate that this choice is crucial.

V.2 Gaussian Dirty-Paper

Our construction for Gaussian dirty-paper parallels that of binary dirty-paper. We used non-binary, Ungerboeck Trellis codes instead of convolutional codes for $\mathcal{C}_0$.

The capacity region of the virtual MAC channel in the Gaussian case is a function of parameters α and Q. The discussion in Section III.2 prescribes $\alpha = P_X/(P_X + P_Z)$ and $Q \geq P_X^2/(P_X + P_Z)$. A selection of $Q = P_X^2/(P_X + P_Z)$ produces a MAC capacity region where the no-interference AWGN channel capacity is achieved at a vertex point. Once again, this point is interesting from a practical implementation viewpoint. To make $\mathcal{C}_1$ approximate random selection with a variance Q, we simply employed a binary LDPC code with the BPSK symbols $\pm\sqrt{Q}$. Note that the use of a binary code means that unlike the construction of Section III.2, the empirical distribution of $\mathcal{C}_1$'s codewords would not be approximately Gaussian. This choice for $\mathcal{C}_1$ is motivated by the experience that is available for the low-SNR regime with no-interference channels, as noted in Section III.3. In that section we also briefly discussed simple methods for shaping $\mathcal{C}_1$, that are applicable in the high-SNR regime. In this section, however, we focus on the low-SNR regime. The motivation for this choice will be explained shortly.

In our simulations, we experimented at a rates 0.25 and 0.5 bits per real dimension. We begin by describing our construction for rate 0.25. The dirty paper Costa (and Shannon) limit at this rate is -3.82 dB. We experimented at an SNR of -2.6dB.

We began by selecting $A = 2$ (this choice will be explained shortly). We selected the quantization code $\mathcal{C}_0$ as a Trellis code borrowed from Ungerboeck [48] of memory 9 [8]. The feedback polynomials are given by the octal digits $(1072, 0342)$, the output alphabet consisted of the 4-PAM signals [-0.75 -0.25 0.25 0.75]. Simulation results indicate that the code is capable of quantization of a source that is uniformly distributed in $[-A/2, A/2]$ with a mean square distortion of $P_X = 0.061$. The random-coding achievable distortion for rate 1 bit per channel use is approximately 0.0585 [9] (the gap from P_X is 0.18 dB), and hence $\mathcal{C}_0$ operates close to the limit. Note that since $\mathcal{C}_0$ is required to quantize a source that is uniformly distributed in a cube (rather than Gaussian i.i.d), shaping is not an issue in its design[10]. Thus, standard trellis codes, which do not

[8] Erez and ten Brink [23] used a memory-6 convolutional quantizer in their nested-lattice based scheme. However, their approach requires the application of the BCJR algorithm on a joint accumulator-quantizer code, rather than on the quantizer code alone, resulting in the same complexity as our scheme of memory 9.

[9] This figure was computed by solving $R \sim \log_2 A - 1/2 \log_2(2\pi e D)$ for $R = 1$.

[10] We use "shaping" in the context of the the region within which the codewords of $\mathcal{C}_0$ reside, as defined in [26]. This must not be confused with the definition that is used in nested-lattices literature (see e.g. [37]), which refers to the shape of the fundamental Voronoi cell of a code or lattice.

exhibit any shaping gain, are sufficient.

The noise variance in the simulations was thus set to $P_Z = 0.111$, in accordance with the above selected SNR of -2.6dB.

Note that we set $A = 2$ arbitrarily, and selected $\mathcal{C}_0$ without any regard for P_X. This value for A, along with the code alphabet could be scaled to produce any desired P_X. We selected a binary LDPC code for $\mathcal{C}_1$. We mapped the code bits to the BPSK signals ± 0.147, approximately corresponding to an energy of $Q = P_X^2/(P_X + P_Z) = 0.0216$.

We designed the edge-distributions of $\mathcal{C}_1$ using a modified version of the EXIT charts approach that was suggested by ten Brink *et al.* [47] for the design of LDPC codes for transmission over multiple-input, multiple-output (MIMO) fading channels[11]. Our approach replaces the function $I_{E,DET}(\cdot)$ of [47], which accounts for the effect of a MIMO detector, with $I_{E,\mathcal{C}_0}(I_A, SNR)$ which accounts for the effect of the convolutional code $\mathcal{C}_0$. Like $I_{E,DET}(\cdot)$ of [47], the value of $I_{E,\mathcal{C}_0}(I_A, SNR)$ is computed empirically, by simulations. We obtained the following edge distributions: $\lambda(2,3,5) = (0.45, 0.3, 0.25)$ and $\rho(2,3,4) = (0.108921, 0.048247, 0.842832)$. The rate of $\mathcal{C}_1$ is 0.25.

As with the decoder for the binary channel, we used a joint iterative belief-propagation decoder to decode $\mathcal{C}_0$ and $\mathcal{C}_1$ (the decoder is described in Appendix D). Unlike our simulations in Section V.1, a non-random construction of $\mathcal{C}_1$ (see Appendix D.3) was *not* useful (and not applied). We attribute this to the almost-regular left edge-distribution of the code we used.

Simulation results for the above dirty paper scheme of rate 0.25 indicate a bit error rate of approximately $5.6 \cdot 10^{-6}$ at a block length of $2 \cdot 10^5$ (50 simulations). As noted above, these results were obtained at an SNR approximately 1.2 dB away from the Shannon limit. The results are slightly better than the ones obtained in [23] using coset dilution.

As in our discussion of binary dirty paper, we now compare these results to results for the Gaussian dirty tape channel. The gap between the information-theoretic limit for the dirty tape "inflated lattices" scheme of [21] (discussed in Section IV) and the Shannon limit for the Gaussian dirty paper problem, is greatest at low SNR, and approaches approximately 4dB. This was the motivation for the preference of low SNR for the dirty paper simulations of [37] and in this paper, because this is the point where there is the most potential for gain by exploiting the non-causally known data. The"inflated lattices" limit at rate 0.25 bits per real dimension is -0.6 dB (SNR) (see [23]). This limit is thus surpassed by the above described dirty paper scheme at SNR -2.6dB.

[11] This EXIT chart approach assumes binary LDPC codes. It can be adapted to nonbinary codes (which are desirable at higher spectral efficiencies) using the methods suggested in [5].

Problem	SNR at rate 0.25	SNR at rate 0.5
Dirty tape “inflated lattices” limit	-0.6 dB	2.6 dB
Dirty paper simulations	-2.6 dB	1 dB
Dirty paper Shannon limit	-3.82 dB	0 dB

Table 2: Results for the Gaussian dirty paper problem at rates 0.25 and 0.5 bits per real dimension.

The different limits are summarized in Table 2.

We now discuss our simulations at rate 0.5 bits per real dimension. The dirty-paper Costa (and Shannon) limit at this rate is 0 dB. We experimented at an SNR of 1 dB, at a gap of 1 dB from the Shannon limit. We used the same value of A and the same trellis code as in the above construction at rate 0.25 bits per dimension. Consequently, P_X was 0.061 and P_Z was set at 0.0484 (corresponding to an SNR of 1 dB). The energy $Q = P_X^2/(P_X + P_Z) = 0.0339$. We designed a binary LDPC code using the above described EXIT chart method, and obtained the following edge distributions: $\lambda(2, 4) = (0.4, 0.6)$ and $\rho(1, 2, 6) = (0.006839, 0.007957, 0.985204)$ [12]. The rate of $\mathcal{C}_1$ is 0.5. We again used a joint iterative belief-propagation decoder to decode $\mathcal{C}_0$ and $\mathcal{C}_1$ constructed as in Appendix D.

Simulation results for this dirty paper scheme of rate 0.5 indicate a bit error rate of approximately $6 \cdot 10^{-5}$ at a block length of $2 \cdot 10^5$ (50 simulations). As noted above, these results were obtained at an SNR 1 dB away from the Shannon limit. The results are similar to the ones obtained in [23] using coset dilution. The dirty tape “inflated lattices” limit at 0.5 bits per real dimension is 2.6 dB. This limit is thus surpassed by our dirty-paper scheme at 1 dB.

VI Superposition Coding for General Gel’fand Pinsker Channels

In this section we propose an extension of superposition coding to general Gel’fand Pinsker (non-causal side-information) channels whose capacity is given by (4). Our goal is to provide motivation for the approach, and hence we focus on a random-coding analysis, while the practical implementation is the subject of current research.

Let $\mathcal{U}$ be the alphabet of the random variable U in (4), and let $F_U(x)$ be the distribution function of $p(u)$. We define a mapping $\delta : [-A/2, A/2] \to \mathcal{U}$ by $\delta(x) = F_U^{-1}(x/A + 1/2)$. With this choice, if X is uniformly distributed over $[-A/2, A/2]$, $U = \delta(X)$ is distributed as $p(u)$. We

[12] The use of check-nodes of degree 1 is equivalent to “doping”, because it forces certain bits of the LDPC code to be zero, thus revealing their value to the decoder. The use of doping was also observed to be beneficial in the coset-dilution simulations of [23].

also define two codes $\mathcal{C}_0$ and $\mathcal{C}_1$. We generate $\mathcal{C}_0$ by random i.i.d selection according to a uniform distribution in the range $[-A/2, A/2]$ ($A > 0$ being an arbitrary real number). $\mathcal{C}_1$ is generated by random i.i.d selection according to some distribution $p(v_1)$ which will be determined later. The superposition code is defined by $\mathcal{C} = \{\delta(\mathbf{c}_0 + \mathbf{c}_1 \bmod A) : \mathbf{c}_0 \in \mathcal{C}_0, \mathbf{c}_1 \in \mathcal{C}_1\}$ (δ being applied componentwise).

Encoding and decoding proceed as follows:

Encoder: The encoder selects a codeword $\mathbf{c}_1 \in \mathcal{C}_1$. Given $\mathbf{s}$, the side-information vector, it seeks a codeword $\mathbf{c}_0 \in \mathcal{C}_0$ such that the word $\mathbf{u} \triangleq \delta(\mathbf{c}_1 + \mathbf{c}_0 \bmod A)$ is jointly strongly ϵ-typical with $\mathbf{s}$. It then transmits $\mathbf{x} = f(\mathbf{s}, \mathbf{u})$ ($f(\cdot)$ being applied componentwise).

Decoder: The decoder receives $\mathbf{y}$ and decodes a pair of codewords $\hat{\mathbf{c}}_0$ and $\hat{\mathbf{c}}_1$ which are jointly strongly ϵ-typical with $\mathbf{y}$, assuming the MAC channel whose probability distribution is given by:

$$\Pr[y \mid x_0, x_1] \triangleq \sum_s \Pr[y \mid s, \delta(x_0 + x_1 \bmod A)] \Pr[s] \tag{17}$$

We define an *encoder error* and *decoder error* as in the binary and Gaussian cases.

Theorem 3 *Given the above construction, an encoder/decoder pair can be designed such that the following holds:*

1. *The probability of an encoder error approaches zero with the block length n, if R_0 satisfies*

$$R_0 > I(U; S) \tag{18}$$

2. *The probability of a decoder error approaches zero with n if the pair (R_0, R_1) lies in the interior of the following achievable region for the MAC channel whose transition probabilities are determined by (17).*

$$\begin{aligned} R_0 + R_1 &\leq I(U; Y) && (19) \\ R_1 &\leq I(U; Y) - I(V_0; Y) && (20) \end{aligned}$$

where $I(V_0; Y)$ is a term that depends on $p(v_1)$, whose precise definition is provided in Appendix F.1.

An outline of the proof of the theorem is provided in Appendix F.1. Note that the bounds in the theorem are similar to the bounds in Theorems 1 and 2.

The MAC achievable region is a function of the term $I(V_0; Y)$. This term is affected by the choice of $p(v_1)$, which determined the random generation of $\mathcal{C}_1$. It is our objective to design $p(v_1)$

such that the point $R_1 = I(U;Y) - I(U;S)$ (the achievable rate of the Gel'fand-Pinsker channel for a given choice of U) and $R_0 = I(U;S)$ lies within this achievable region. To avoid being bounded away from capacity by (20), we must select $p(v_1)$ such that $I(V_0;Y) \leq I(U;S)$. Several such choices for $p(v_1)$ are discussed in Appendix F.2.

So far, our discussion assumed randomly constructed codes. In a practical setting, we would typically select the code $\mathcal{C}_0$ as a convolutional code or trellis code, and $\mathcal{C}_1$ to be an LDPC code. Using the Viterbi algorithm, the encoder would seek the codeword $\mathbf{c}_0$ that maximizes $\Pr[\mathbf{s} \,|\, \delta(\mathbf{c}_1 + \mathbf{c}_0)] = \prod_{i=1}^{n} \Pr[S = s_i \mid U = \delta(c_{1,i} + c_{0,i})]$, where we have applied the above defined relation $p(s \mid u)$. The decoder, using joint BCJR and LDPC belief-propagation decoding, would seek a pair of codewords $\mathbf{c}_0$ and $\mathbf{c}_1$ such that $\Pr[\mathbf{y} \,|\, \delta(\mathbf{c}_0 + \mathbf{c}_1)] = \prod_{i=1}^{n} \Pr[y_i \,|\, \delta(c_{1,i} + c_{0,i})]$ is maximized.

VII Conclusion

Superposition coding provides a simple approach to constructing codes for the binary and Gaussian dirty-paper problems. This simplicity produces several advantages. For example, the component codes are not required to be building-blocks of a lattice, and hence are allowed to be nonlinear, adding an extra degree of freedom to their design. The simple structure of superposition-codes also lends itself to the simple design of codes and decoders, as demonstrated in Section V.

An important element of our analysis has been the analogy between our scheme's decoding problem and the decoding problem over a MAC channel. This analogy has enabled the utilization, in Section V, of a wide body of research that exists for the design of practical decoders for the MAC channel.

In Section V we have presented promising simulation results for the binary and Gaussian dirty paper channels. In particular, the results for the Gaussian dirty-paper channel indicate reliable transmission within 1.2 dB of the Shannon limit, at a rate of 0.25 bits per real dimension, and within 1 dB of the Shannon limit at a rate of 0.5 bits per real dimension. These results are the same as the best reported results with nested lattices [23]. Superposition coding was recently applied by Sun *et al.* [45] (using a stronger trellis code and a different design approach for the decoder) to obtain codes capable of reliable transmission within 0.88 dB of the Shannon limit at 0.25 bits per real dimension.

The approach presented in this paper assumes scalar interference channels. However, the approach may easily be applied to a vector MIMO setting by decomposing the vector channel into an array of scalar channels, exactly as performed by Telatar [46][Section 3.1] in the context of the no-interference channel[13]. A different method for extending our approach to the vector MIMO setting was recently suggested by Lin *et al.* [34].

Superposition-coding relies on a close relation between the dirty-paper channel and the multiple-access channel. Interestingly, a relationship between these two problems has recently been observed by Vishwanath *et al.* [50] who obtained a duality between the MIMO broadcast dirty-paper achievable region and the MIMO MAC capacity region.

Appendix

A Proof of Lemma 2

For any input product distribution $p(x_0, x_1) = p(x_0)p(x_1)$, define the region $\mathcal{R}(p(x_0)p(x_1))$ given by

$$\begin{aligned} R_0 &\leq I(X_0; Y|X_1) \\ R_1 &\leq I(X_1; Y|X_0) \\ R_0 + R_1 &\leq I(X_0, X_1; Y) \end{aligned} \tag{21}$$

The capacity region of the MAC (6) is given by the closure of the convex hull of the union of all regions $\mathcal{R}(p(x_0)p(x_1))$, for all $p(x_0)p(x_1)$ satisfying the input constraint, i.e., for which $p(x_1)$ satisfies

$$\mathbb{E}[d_H(X_1, 0)] \leq q$$

We observe that

$$I(X_1; Y|X_0) \leq h(p \otimes q) - h(p) \tag{22}$$

since the RHS in (22) is the capacity of the input-constrained BSC $Y = X_1 + Z$ (under conditioning with respect to X_0, the contribution of user 0 can be removed from the received signal). We observe also that

$$I(X_0; Y|X_1) \leq 1 - h(p) \tag{23}$$

since the RHS in (23) is the capacity of the BSC $Y = X_0 + Z$. Finally, we observe that

$$I(X_0, X_1; Y) = H(Y) - h(p) \leq 1 - h(p) \tag{24}$$

[13] The decomposition, as well as the allocation of power to the individual scalar channels, is performed as though no interference was present. After the decomposition, dirty-paper coding proceeds using a modified interference, resulting from a multiplication of the vector of interference components by a fixed unitary matrix.

By letting $p(x_1)$ be Bernoulli(q) and $p(x_0)$ be Bernoulli(1/2), the upper bounds (22), (23) and (24) are simultaneously achieved. Since the resulting region is closed and convex, no closure and convex hull operations are needed. □

B Proof of Theorem 2

In this proof, we consider the following channel model:

$$Y = X + S + Z \bmod A/\alpha \tag{25}$$

Y in this model corresponds to the channel output as in (9) after the modulo operation was performed, but without multiplication by α. Hence the argument to the modulo operation is A/α instead of A. For simplicity of our model, we encapsulate the random known dither into the interference S, and assume that the interference is uniformly distributed in the range $[-A/(2\alpha), A/(2\alpha)]$.

We begin by defining the following set of random variables. X is Gaussian with variance P_X. The distribution of U_1 will be defined later in this section. The variables S, X and U_1 are independent. We also define U_0 to satisfy the equation:

$$U_0 = \alpha S + X - U_1 \bmod A \tag{26}$$

Hence, U_0 is uniformly distributed in the range $[-A/2, A/2]$ and is dependent on S, X and U_1. Note that X is identical to a similar definition by Costa [17]. U_0 and U_1 replace Costa's auxiliary U. The code $\mathcal{C}_0$, as defined in Section III.1, corresponds to random i.i.d selection according to the distribution of U_0. The exact distribution for the random i.i.d generation of $\mathcal{C}_1$ in Section III.1 was left unspecified. We now define it to equal the distribution of U_1, which in turn will be specified later in this section.

To simplify our analysis, we consider an encoder/decoder pair that employs joint-typicality rather than a minimum-distance metric. We begin with the encoder.

Encoder: The encoder selects a codeword $\mathbf{c}_1 \in \mathcal{C}_1$, and seeks a word $\mathbf{c}_0 \in \mathcal{C}_0$ such that the pair $\mathbf{c}_0$ and $(\alpha\mathbf{s} - \mathbf{c}_1 \bmod A)$ are jointly strongly ϵ-typical with respect to the distribution of the random variables U_0 and $(\alpha S - U_1 \bmod A)$ (ϵ will be determined later). If no such $\mathbf{c}_0$ is found, the encoder declares an error. Otherwise, it transmits the sequence $\mathbf{x} = \mathbf{c}_0 + \mathbf{c}_1 - \alpha\mathbf{s} \bmod A$.

Note that the encoder requires *strong* typicality. The justification for this is similar to the one in the theoretical analysis of Gel'fand and Pinsker [29] and will be clarified later.

To bound the probability of an encoder error, we apply Lemma 13.6.2 of [18]. The probability of an encoder error approaches zero with n if R_0 satisfies

$$R_0 > I(U_0; \alpha S - U_1 \bmod A) + \epsilon_1 \tag{27}$$

where ϵ_1 is some value, dependent on ϵ that approaches 0 with ϵ.

$$\begin{aligned} I(U_0; \alpha S - U_1 \bmod A) &= h(U_0) - h(U_0 \mid \alpha S - U_1 \bmod A) \\ &= \log A - h(X \bmod A) \end{aligned}$$

The distribution of $X \bmod A$ approaches the distribution of X as A approaches infinity. Hence,

$$I(U_0; \alpha S - U_1 \bmod A) = \log A - \frac{1}{2}\log(2\pi e P_X) + \delta_1$$

where $\delta_1 \to 0$ as $A \to \infty$. Thus, if (11) is satisfied, then for large enough A and small enough ϵ, (27) is satisfied. This completes the proof of Part 1 of the theorem.

We now define $\hat{Y} = \alpha Y$ and combine (25) and (26) to obtain,

$$\begin{aligned} \hat{Y} &= U_0 + U_1 - (1-\alpha)X + \alpha Z \bmod A \\ &= U_0 + U_1 + \hat{Z} \bmod A \end{aligned} \tag{28}$$

where $\hat{Z} \triangleq -(1-\alpha)X + \alpha Z$. $\hat{Z}$ is distributed as a Gaussian variable with variance $P_{\hat{Z}} \triangleq (1-\alpha)^2 P_X + \alpha^2 P_Z$.

X and Z are independent of U_0 and U_1, and hence $\hat{Z}$ is also independent of U_0 and U_1, thus overcoming the obstacle in (9). Since $\mathcal{C}_0$ and $\mathcal{C}_1$ were constructed according to U_0 and U_1, we would expect the probability of error to approach zero if (R_0, R_1) lie within the capacity region of the MAC channel as defined in Lemma 14.3.1 of [18].

The proof, however, is slightly more involved than the proof of [18]. This is because the channel output $\hat{\mathbf{y}}$ was not generated according to the true MAC channel model (28). Specifically, the self-noise element $\mathbf{x}$ of $\hat{\mathbf{z}}$ was not generated by random selection according to X. We begin by replacing the decoder of [18] with a decoder that requires strong (rather than weak) typicality.

Decoder: The decoder seeks $\hat{\mathbf{c}}_0 \in \mathcal{C}_0$ and $\hat{\mathbf{c}}_1 \in \mathcal{C}_1$ such that the triplet $(\hat{\mathbf{c}}_0, \hat{\mathbf{c}}_1, \mathbf{y})$ are jointly strongly ϵ-typical with respect to the distribution of (U_0, U_1, Y).

We now consider the probability of a decoder error.

1. We start by examining the probability that the channel output $\mathbf{y}$ is not strongly ϵ-typical with the transmitted $\mathbf{c}_0$ and $\mathbf{c}_1$. To determine the probability of this event, we examine some intermediate values.

We first examine the probability that the triplet $(\mathbf{c}_0, \mathbf{c}_1, \alpha\mathbf{s} - \mathbf{c}_1)$ are not jointly strongly ϵ-typical. The random variable U_1 is independent of $(\alpha S - U_1 \bmod A)$ and U_0 by the fact that αS is uniformly distributed in the range $[-A/2, A/2]$.

Similarly, $\mathbf{c}_1$ is independent of $(\alpha\mathbf{s} - \mathbf{c}_1 \bmod A)$. The codebook $\mathcal{C}_0$ was generated independently of $\mathbf{c}_1$, and the selected codeword $\mathbf{c}_0$ is a function of $(\alpha\mathbf{s} - \mathbf{c}_1 \bmod A)$. Hence the codeword $\mathbf{c}_1$ is also independent of the pair $\mathbf{c}_0$ and $(\alpha\mathbf{s} - \mathbf{c}_1 \bmod A)$. Thus, by the weak law of large numbers, we obtain that the probability that $\mathbf{c}_0$, $\mathbf{c}_1$ and $(\alpha\mathbf{s} - \mathbf{c}_1 \bmod A)$ are jointly strongly ϵ-typical approaches one with n.

If the triplet $(\mathbf{c}_0,\ \mathbf{c}_1,\ \alpha\mathbf{s} - \mathbf{c}_1 \bmod A)$ are jointly strongly ϵ-typical, then so are the sequences $(\mathbf{x},\ \mathbf{s},\ \mathbf{c}_0,\ \mathbf{c}_1)$, because $\mathbf{x}$ and $\mathbf{s}$ are obtained from the original triplet by applying deterministic operations. Finally, in a manner similar to the analysis of Gel'fand and Pinsker [29], we observe that $(\mathbf{U}_0, \mathbf{U}_1)$, $(\mathbf{X}, \mathbf{S})$ and $\mathbf{Y}$ form a Markov chain and hence by the Markov lemma (see e.g. [18][Lemma 14.8.1]), $\mathbf{y}$ is jointly strongly ϵ-typical with $\mathbf{c}_0$ and $\mathbf{c}_1$ with probability that approaches 1 with n.

2. We now examine the event that there exist codewords $\mathbf{c}_0' \in \mathcal{C}_0$ and $\mathbf{c}_1' \in \mathcal{C}_1$, other than the transmitted $\mathbf{c}_0$ and $\mathbf{c}_1$, that are jointly strongly ϵ-typical with the channel output $\mathbf{y}$.

   Lemma 4, which is provided in Appendix C extends the achievability proof of the MAC capacity region from [18] to our setting using the fact that $\hat{\mathbf{y}}$ has been shown above to be strongly typical (with probability approaching 1) to the MAC model (28)[14]. All that remains is therefore to evaluate the MAC capacity region.

   We replace Y of Lemma 4 by $\hat{Y}$ of our discussion, as defined in (28).

$$\begin{aligned} I(U_0; \hat{Y} \mid U_1) &= h(\hat{Y} \mid U_1) - h(\hat{Y} \mid U_0, U_1) \\ &= h(U_0 + \hat{Z} \bmod A) - h(\hat{Z} \bmod A) \end{aligned}$$

   The first element of the sum is maximized by U_0 that is uniformly distributed in the range $[-A/2, A/2]$, matching our choice above. Hence $h(U_0 + \hat{Z} \bmod A) = \log A$. As A approaches infinity, $\hat{Z} \bmod A$ approaches the distribution of $\hat{Z}$. Hence

$$I(U_0; \hat{Y} \mid U_1) = \log A - \frac{1}{2}\log 2\pi e P_{\hat{Z}} + \delta_2 \qquad (29)$$

   where $\delta_2 \to 0$ as $A \to \infty$. We proceed to examine,

$$I(U_1; \hat{Y} \mid U_0) = h(U_1 + \hat{Z} \bmod A) - h(\hat{Z} \bmod A)$$

[14] Note that since $\hat{\mathbf{y}}$ is obtained from $\mathbf{y}$ using a deterministic invertible operation, strong typicality with $\mathbf{y}$ is equivalent to strong typicality with $\hat{\mathbf{y}}$.

For large A, the distribution for U_1 that maximizes the first element of the above sum, under the restriction that the variance does not exceed Q, approaches a Gaussian random variable $\mathcal{N}(0,Q)$. We now define the distribution of U_1 to match this maximizing distribution. Thus,

$$\begin{aligned} I(U_1;\hat{Y}\mid U_0) &= \frac{1}{2}\log 2\pi e(Q+P_{\hat{Z}}) - \frac{1}{2}\log 2\pi e P_{\hat{Z}} + \delta_3 \\ &= \frac{1}{2}\log\left(1+\frac{Q}{P_{\hat{Z}}}\right) + \delta_3 \end{aligned} \tag{30}$$

where $\delta_3 \to 0$ as $A \to \infty$. Using similar arguments,

$$\begin{aligned} I(U_0,U_1;\hat{Y}) &= h(U_0+U_1+\hat{Z} \bmod A) - h(\hat{Z}\text{mod } A) \\ &= \log A - \frac{1}{2}\log 2\pi e P_{\hat{Z}} + \delta_2 \end{aligned} \tag{31}$$

Finally, combining (29), (30), and (31), recalling Lemma 4, we obtain the capacity region of (11) and (12), thus completing the proof of the theorem.

□

C Statement and Proof of Lemma 4

The following lemma extends achievability proof of the MAC capacity region from [18]. It differs from the achievability proof of Theorem 14.3.1 of [18] in that it only assumes that the channel output $\mathbf{y}$ is typical to the MAC, and does not require that the output be truly generated by the MAC. Also, it assumes that the MAC decoder tests for strong-typicality, as in Appendices B and F.1 rather than weak-typicality as in [18].

Lemma 4 *Let $p(y\mid u_0,u_1)$ be the transition probabilities of a MAC. Let $\mathcal{U}_0$ and $\mathcal{U}_1$ be two random codes generated by i.i.d selection according to distributions $p(u_0)$ and $p(u_1)$, at rates R_0 and R_1 (respectively). Let $\mathbf{u}_0$ and $\mathbf{u}_1$ be the transmitted codewords and $\mathbf{y}$ the corresponding channel output. Assume a MAC decoder that tests for joint strong ϵ-typicality. If $(\mathbf{u}_0,\mathbf{u}_1,\mathbf{y})$ are jointly strongly ϵ-typical, then for small enough ϵ, the ensemble probability of decoding error approaches zero with the block length n if the following inequalities hold:*

$$\begin{aligned} R_0 &< I(U_0;Y\mid U_1) \\ R_1 &< I(U_1;Y\mid U_0) \\ R_0+R_1 &< I(U_0,U_1;Y) \end{aligned} \tag{32}$$

Proof. An error occurs if there exists any other pair of codewords $(\mathbf{u}_0', \mathbf{u}_1') \neq (\mathbf{u}_0, \mathbf{u}_1)$ such that $(\mathbf{u}_0', \mathbf{u}_1', \mathbf{y})$ are jointly strongly ϵ-typical.

We first examine the probability of a codeword $\mathbf{u}_0' \neq \mathbf{u}_0$ being jointly strongly ϵ-typical with $(\mathbf{u}_1, \mathbf{y})$. Let $\mathbf{U}_0$ denote a random sequence, generated by i.i.d selection according to $p(u_0)$, independently of $\mathbf{u}_1$ and $\mathbf{y}$. Invoking Lemma 13.6.2 of [18] we obtain,

$$Pr[(\mathbf{U}_0, \mathbf{u}_1, \mathbf{y}) \in A_\epsilon^*] < 2^{-n[I(U_0;U_1,Y)-\epsilon_1]}$$

where A_ϵ^* denotes the set of strongly ϵ-typical sequences and ϵ_1 approaches zero as $\epsilon \to 0$ and $n \to \infty$. Examining $I(U_0; U_1, Y)$ we have,

$$\begin{aligned} I(U_0; U_1, Y) &= h(U_1, Y) - h(U_1, Y \mid U_0) = h(Y \mid U_1) + h(U_1) - [h(Y \mid U_0, U_1) + h(U_1 \mid U_0)] \\ &= h(Y \mid U_1) - h(Y \mid U_1, U_0)] = I(U_0; Y \mid U_1) \end{aligned}$$

The third equality follows from the fact that U_0 and U_1 are independent, and hence $h(U_1 \mid U_0) = h(U_1)$. Using a union bound, we obtain,

$$\Pr[\exists \mathbf{u}_0' \in \mathcal{U}_0 : \mathbf{u}_0' \neq \mathbf{u}_0, (\mathbf{u}_0', \mathbf{u}_1, \mathbf{y}) \in A_\epsilon^*] < 2^{nR_0} \cdot 2^{-n[I(U_0;Y \mid U_1)-\epsilon_1]} = 2^{n[R_0 - I(U_0;Y \mid U_1)+\epsilon_1]}$$

For small enough ϵ, the probability of this error event approaches zero with n if the first inequality in (32) is satisfied. Similarly, the probability of a codeword $\mathbf{u}_1' \neq \mathbf{u}_1$ being jointly strongly ϵ-typical with $(\mathbf{u}_0, \mathbf{y})$ approaches zero if the second inequality in (32) is satisfied.

Finally we have,

$$Pr[(\mathbf{U}_0, \mathbf{U}_1, \mathbf{y}) \in A_\epsilon^*] < 2^{-n[I(U_0,U_1;Y)-\epsilon_1]}$$

and hence the probability of two codewords $\mathbf{u}_0' \neq \mathbf{u}_1$ and $\mathbf{u}_1' \neq \mathbf{u}_1$ being jointly strongly ϵ-typical with $\mathbf{y}$ approaches zero if the last inequality in (32) is satisfied. □

The application of Lemma 4 in our context of side-information channels requires a discussion of the following fine point. An underlying assumption of the lemma is that the codewords of $\mathcal{C}_0$ and $\mathcal{C}_1$, except for the transmitted $\mathbf{c}_0$ and $\mathbf{c}_1$, have been generated by a random selection process independently of $\mathbf{y}$. This assumption is true in the context of the regular MAC channel. The reasoning is that the production of the transmitted codewords $\mathbf{c}_0$ and $\mathbf{c}_1$ and subsequently the generation of $\mathbf{y}$, has no relation to the random events that produced the other codewords $\mathbf{c}_0^{(2)}, ..., \mathbf{c}_0^{(M_0)}$ and $\mathbf{c}_1^{(2)}, ..., \mathbf{c}_1^{(M_1)}$ of both codes.

In our case, we assume that the encoder examines the codewords of $\mathcal{C}_0$ by some order, and stops at the ith codeword if it is strongly typical with $\mathbf{c}_1$ and $\mathbf{s}$. Hence, given that the encoder selected $\mathbf{c}_0^{(i)}$, it cannot be argued that the production of codewords $\mathbf{c}_0^{(1)}, ..., \mathbf{c}_0^{(i-1)}$ is independent of $\mathbf{s}$ and $\mathbf{c}_1$. Each of the codewords is known to be not typical with $\mathbf{s}$ and $\mathbf{c}_1$. Consequently, they are not in general independent of $\mathbf{y}$.

The solution to this problem is simple: Our random-coding analysis holds for codewords $\mathbf{c}_0^{(i+1)}, ..., \mathbf{c}_0^{(M_0)}$, since they were produced independently of $\mathbf{y}$. Codewords $\mathbf{c}_0^{(1)}, ..., \mathbf{c}_0^{(i-1)}$ are randomly selected among words that are *not* typical with $\mathbf{c}_1$ and $\mathbf{s}$. For large enough n, the probability that they are jointly typical with $\mathbf{y}$ is bounded by a multiplicative constant factor times the probability assuming independent selection. Examining the proof of Lemma 4, it is easy to verify this is sufficient for our needs.

To see this, let $\mathbf{U}_0$ be a randomly selected sequence distributed i.i.d as $p(u_0)$. We first examine the probability that $\mathbf{U}_0$ would be jointly typical with the transmitted $\mathbf{c}_1$ and $\mathbf{y}$, given that it was *not* typical with $\mathbf{s}$ and $\mathbf{c}_1$.

$$\begin{aligned}
\Pr[(\mathbf{U}_0, \mathbf{c}_1, \mathbf{y}) \in A_\epsilon^* \mid (\mathbf{U}_0, \mathbf{c}_1, \mathbf{s}) \notin A_\epsilon^*] &= \frac{\Pr[(\mathbf{U}_0, \mathbf{c}_1, \mathbf{y}) \in A_\epsilon^*, (\mathbf{U}_0, \mathbf{c}_1, \mathbf{s}) \notin A_\epsilon^*]}{\Pr[(\mathbf{U}_0, \mathbf{c}_1, \mathbf{s}) \notin A_\epsilon^*]} \\
&\leq \frac{\Pr[(\mathbf{U}_0, \mathbf{c}_1, \mathbf{y}) \in A_\epsilon^*]}{(1 - 2^{-n(I(U_0;U_1,S)-\epsilon_1)})}
\end{aligned}$$

For large enough n, the denominator is greater than 1/2. Hence we obtain,

$$\Pr[(\mathbf{U}_0, \mathbf{c}_1, \mathbf{y}) \in A_\epsilon^* \mid (\mathbf{U}_0, \mathbf{c}_1, \mathbf{s}) \notin A_\epsilon^*] \quad \leq \quad 2 \cdot \Pr[(\mathbf{U}_0, \mathbf{c}_1, \mathbf{y}) \in A_\epsilon^*]$$

Let $\mathbf{U}_1$ be randomly i.i.d distributed as $p(u_1)$, we can analyze the probability that $\mathbf{U}_0$ would be jointly typical with some other codeword $\mathbf{U}_1$ of $\mathcal{C}_1$ and $\mathbf{y}$, given that it was *not* typical with $\mathbf{s}$ and the transmitted $\mathbf{c}_1$. This case is handled in exactly the same manner as the previous one we have just examined, and is therefore omitted.

The last error event of the MAC channel involves the probability that a codeword $\mathbf{c}_1'$, other than the transmitted $\mathbf{c}_1$, would be jointly typical with $\mathbf{c}_0$ and $\mathbf{y}$. Unlike the case of $\mathcal{C}_0$, the independence of $\mathbf{c}_1$ from the other codewords of its codebook *does* carry over from the MAC model. Hence, this case does not require any special attention.

D Joint Iterative Decoding for Superposition Codes

The joint iterative decoder used in this paper is based on the concepts of Boutros and Caire [8]. We use the terminology of factor graphs, which was introduced by Kschischang *et al.* [31].

Fig. 3 is a schematic diagram of the factor graph which forms the basis of the decoding process. The gray-filled circles represent channel observation nodes. The black squares are MAC factor

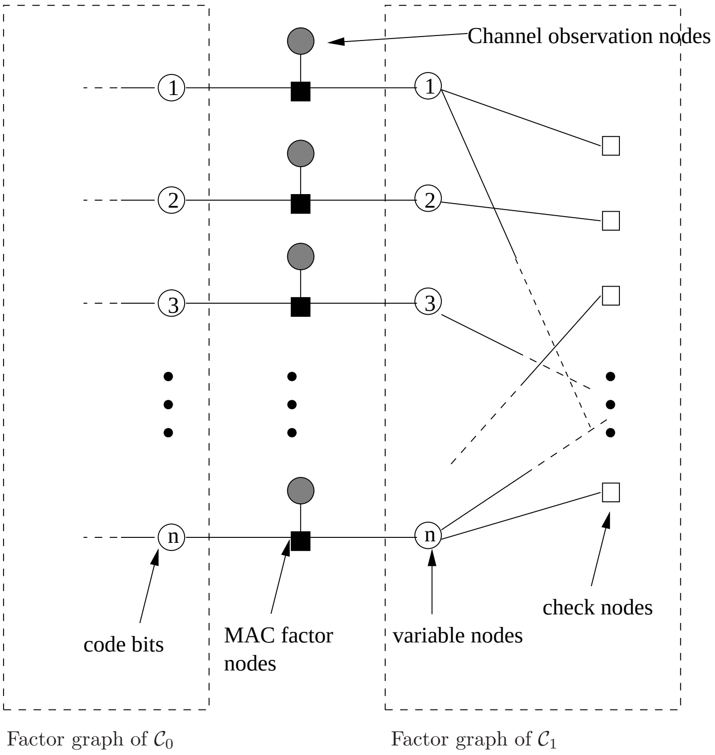
Channel observation nodes
1
2
3
n
1
2
3
n
code bits
MAC factor nodes
variable nodes
check nodes
Factor graph of $\mathcal{C}_0$
Factor graph of $\mathcal{C}_1$

Figure 3: Factor graph for joint iterative decoding for binary dirty-paper.

nodes that process the channel observations and the information obtained from one code and communicate them to the other code. The left hand side of the graph contains the factor graph of $\mathcal{C}_0$. For simplicity, we have only drawn nodes corresponding to bits of the convolutional code (with binary dirty-paper) or symbols of the trellis code (with Gaussian dirty-paper). The right hand side of the graph contains the factor (Tanner) graph of $\mathcal{C}_1$.

D.1 Binary Dirty-Paper

We first consider the binary dirty-paper problem. For this problem, $\mathcal{C}_1$ is designed to be a GQC-LDPC code. As mentioned in Section V.1, GQC-LDPC codes are obtained from LDPC codes defined over Galois fields[15] $\mathbb{F}_q$ where $q \geq 2$. The non-binary symbols of the code are mapped to a binary channel alphabet using a mapping function, $\delta : \mathbb{F}_q \to \{0, 1\}$. The $\mathcal{C}_1$ portion of the factor-graph is involved in recovering the nonbinary codeword of the $\mathbb{F}_q$-LDPC code (from which the GQC-LDPC code was constructed). Thus, the nodes of this portion of the factor graph exchange *vector* messages among themselves and with the MAC factor nodes. $\mathcal{C}_0$, on the other hand, is a binary convolutional codes. Its nodes therefore exchange scalar messages.

The MAC factor nodes perform the translation between the two types of message. For example, given a vector message $\mathbf{v} = [v_0, ..., v_{q-1}]$ from an LDPC variable node (with some abuse of notation, we let the indices $0, ..., q-1$ represent the elements of $\mathbb{F}_q$), the following scalar message s is transferred to the convolutional code:

$$s = \frac{\sum_{i=0}^{q-1} \Pr[y \mid \delta(i) + 1] \cdot v_i}{\sum_{i=0}^{q-1} \Pr[y \mid \delta(i) + 1] \cdot v_i + \sum_{i=0}^{q-1} \Pr[y \mid \delta(i)] \cdot v_i} \tag{33}$$

where y denotes the channel output observed by the MAC factor node, $\delta(i) + 1$ is computed over $\mathbb{F}_2$ (the channel alphabet), and $\Pr[y \mid \delta(i) + 1]$ is evaluated using the virtual MAC model, (6).

When decoding begins, the variable nodes of the LDPC factor-graph transmit messages $\mathbf{v} = [1/q, ..., 1/q]$ to the MAC factor node, indicating zero knowledge. The factor nodes apply (33) to obtain scalar messages that are transferred to the factor graph of $\mathcal{C}_0$. It is important to observe that by applying (33), the MAC factor nodes implicitly benefit from the knowledge of the mapping $\delta(\cdot)$ and through it, of the expected weight of GQC-LDPC codewords.

The convolutional code uses the scalar messages from the MAC factor nodes to perform a BCJR iteration. The BCJR algorithm is applied using the approach developed by Berrou and

[15] In the definition of [4], GQC-LDPC codes are obtained from *coset*-$\mathbb{F}_q$-LDPC codes rather than $\mathbb{F}_q$-LDPC codes. For simplicity of our discussion, we neglect the addition of a coset vector to the LDPC code. This addition can easily be accounted for by modifying expressions (33),(34), (35) and (36).

Glavieux [6] [16]. With this approach, the message produced at each of the convolutional bit nodes, and directed to a MAC factor node, does not rely on the information that was previously obtained from that factor node. Note that unlike the algorithm of [6], no distinction is made between systematic and non-systematic bits, and outgoing messages are computed for all.

The MAC factor nodes translate each scalar message s into a vector message $\mathbf{v}$ using the following expression:

$$v_i = \alpha \cdot (\Pr[y \mid \delta(i) + 1] \cdot s + \Pr[y \mid \delta(i)] \cdot (1 - s)) \tag{34}$$

where α is a normalization constant selected so that the sum of the $\mathbf{v}$ elements is one.

Decoding proceeds by performing decoding GQC-LDPC iterations, replacing the initial messages with the ones obtained from the MAC factor nodes. Typically, a number of LDPC decoding iterations are performed before information is transferred leftward toward the MAC factor nodes (this will be elaborated shortly). When the last such decoding iteration is performed, each variable-node computes a message to the corresponding MAC factor node in the following way. The expression for rightbound LDPC messages is applied, using *all* incoming leftbound messages, and excluding the initial message.

The MAC factor nodes transfer the information leftward toward the factor graph of the convolutional code. After a BCJR iteration, new information is send rightward through the MAC factor nodes and to the LDPC code. The subsequent LDPC iterations proceed as before. At the first such rightbound iteration, the leftbound messages that were generated prior to the BCJR iteration are employed.

Note that BCJR iterations are more costly (in terms of execution time) than LDPC decoding iterations. In our experiments, we have found that a more cost-effective approach executes approximately 10 LDPC decoding iterations before executing a BCJR iteration.

D.2 Gaussian Dirty-Paper

Now consider decoding for the Gaussian dirty-paper problem. The decoding process is similar to the one described above for binary dirty-paper. We focus only on the differences between the two.

With Gaussian dirty-paper, $\mathcal{C}_1$ is designed to be a standard binary LDPC code (assuming low SNR as discussed in Section V.2), while $\mathcal{C}_0$ is a nonbinary trellis code. Thus the messages exchanged between the MAC factor nodes and $\mathcal{C}_1$ nodes are scalar, and the messages exchanged with the nodes of $\mathcal{C}_0$ are vectors (the exact opposite of the situation with binary dirty-paper).

[16] In our work, we have found it more convenient to use plain-likelihood messages rather than log-likelihood (LLR) messages as in [6]. The translation one from to the other is immediate

Expression (33) is replaced by,

$$v_i = \alpha \cdot \Big(f_\sigma((y - \delta(i) - \sqrt{Q}) \bmod A) \cdot s + f_\sigma((y - \delta(i) + \sqrt{Q}) \bmod A) \cdot (1 - s) \Big) \tag{35}$$

where s is the incoming scalar message from the LDPC code, $\mathbf{v} = [v_0, ..., v_{2^b-1}]$, is the outgoing vector message. The indices $i = 0, ..., 2^b - 1$ represent binary subsequences that are produced at each encoding step of the trellis code, and $\delta(i)$ is their mapping to channel signals. f_σ is the density function of a Gaussian random variable $\mathcal{N}(0, \sigma^2)$, where σ^2 is the variance of the effective noise $\hat{Z}$ in (10). $\pm\sqrt{Q}$ are the BPSK symbols of the LDPC code $\mathcal{C}_1$, as discussed in Section V.2.

Similarly, (34) is replaced by,

$$s = \frac{\sum_{i=0}^{2^b-1} f_\sigma((y - \delta(i) - \sqrt{Q}) \bmod A) \cdot v_i}{\sum_{i=0}^{2^b-1} f_\sigma((y - \delta(i) - \sqrt{Q}) \bmod A) \cdot v_i + \sum_{i=0}^{2^b-1} f_\sigma((y - \delta(i) + \sqrt{Q}) \bmod A) \cdot v_i} \tag{36}$$

where $\mathbf{v}$ is the incoming message from the trellis code, and s is the outgoing scalar message to the LDPC code.

The BCJR algorithm is easily extended from binary convolutional codes to nonbinary trellis codes (see e.g. [41]). We use the extended BCJR algorithm for the portion of the decoding that involved $\mathcal{C}_0$. Since $\mathcal{C}_1$ is a standard binary LDPC code, we use standard LDPC decoding iterations for its portion of the decoding process.

D.3 Non-Random Construction of $\mathcal{C}_1$

A significant improvement in the decoder's performance was obtained by applying the following design rule. Consider the construction of $\mathcal{C}_1$ according to a given edge distributions λ and ρ and at a block length of n. For each left degree i, the parameters λ, ρ and n prescribe the *number* n_i of variable-nodes of degree i. They do not specify the *identity* of the variable-nodes. One option is to design nodes at consecutive indices to have the same degree. For example, in Fig. 3, nodes 1, 2, 3,...,n_2 would have degree 2, nodes $n_2 + 1, ..., n_2 + n_3$ would have degree 3, and so forth. An alternative option would spread same-degree variable-nodes randomly among the indices. Typically, in a standard point-to-point channel, the random construction of edges means that both options produce similar performance. In the context of our decoder for the MAC channel, we have found that a *non-random* assignment produces superior performance.

The explanation for this begins with the observation that the higher the degree of a variable node, the greater the reliability of the information it transmits (via the MAC factor node) to its corresponding $\mathcal{C}_0$ bit node. At the following BCJR iteration, the reliability of the message produced at each bit node, is a function of the reliability of the messages at the other bit-nodes (this is the result of the "extrinsic information" rule). Furthermore, the BCJR decoder is characterized by a “window” phenomenon, by which the value produced at a bit node is influenced mainly by bits at nearby indices, and is weakly affected by bits that are far away.

In Fig. 3, $\mathcal{C}_0$ bit nodes of consecutive indices are connected, via the MAC factor-nodes, to $\mathcal{C}_1$ variable nodes at consecutive indices. In a *random* assignment of LDPC variable-node degrees, neighboring $\mathcal{C}_0$ bit-nodes would receive information of a varying degree of reliability from the corresponding $\mathcal{C}_1$ variable-nodes. In a *non-random* assignment, neighboring bit nodes would typically receive similarly reliable information from their neighbors. Fortunate bit nodes would receive consistently better information from their neighbors than unfortunate ones. This would result in irregularity in the BCJR decoder (a desirable quality in iterative soft-decoders), thus significantly improving the performance of the joint decoder.

E Decoding by Successive Cancellation for the Binary Dirty-Paper Channel

In this section we describe our simulations for the binary dirty-paper channel, using a decoder based on successive cancellation, instead of iterative multiuser detection as in Section V.1. The channel parameters $W = 0.3$ and $p = 0.1$ were the same as the ones used in Section V.1.

For the quantization code $\mathcal{C}_0$, we used the same convolutional code as in Section V.1. As noted in that section, this code has excellent quantization capabilities. Its channel coding abilities, however, are less favorable. Simulation results indicate that the code is able to correct a bit error rate of approximately 0.25, instead of 0.295 as would be expected of a random code of the same rate. Moreover, it produces a bit error of approximately 0.008. This has two implications for the design of the information-bearing code $\mathcal{C}_1$. First, the code's weight constraint q' must satisfy $q' \otimes p = 0.25$, in order that the level of noise at the input to the convolutional decoder does not exceed 0.25. This yields $q' = 0.1875 = 3/16$. Second, the noise level at the input to the auxiliary code must account for the noise produced by the convolutional code, i.e. $p' = 0.1 \otimes 0.008 = 0.1064$.

The information-bearing code was thus designed for a BSC channel with error p' and the codeword weight constraint q'. We selected a GQC-LDPC code as discussed in Section V.1, but over an enlarged alphabet $\mathbb{F}_{16}$ instead of $\mathbb{F}_4$. We mapped 3 elements of the $\mathbb{F}_{16}$ alphabet to the binary digit 1 and the rest to 0. Thus, the normalized weight of each resulting GQC-LDPC codeword was approximately 3/16, as desired. The random coding capacity of the above BSC (q, p') channel is 0.32. Using the methods of [4] and [5], we obtained a GQC-LDPC code at rate 0.25. The edge distributions for this code are given by $\lambda(2, 3, 4, 5, 9, 10, 21) =$

$(0.55253, 0.17636, 0.03867, 0.07163, 0.092431, 0.017174, 0.051208)$ and $\rho(2,3) = (0.1, 0.9)$.

Finally, simulation results for the above dirty paper scheme of rate 0.25 indicate a bit error rate of 10^{-5} at a block length of $5 \cdot 10^4$ (100 simulations).

F Results for Section VI

F.1 Outline of Proof for Theorem 3

Let V_0 be uniformly distributed in the range $[-A/2, A/2]$, V_1 distributed as $p(v_1)$, $V \overset{\Delta}{=} V_0 + V_1 \mathrm{mod} A$ and $U \overset{\Delta}{=} \delta(V)$. U is thus distributed as $p(u)$. The side information S is related to U by $p(s\,|\,u)$. The transmitted signal X is related to S and U by $X = f(S,U)$ (f having been defined above). The channel output Y, is related to S and X by the channel transition function, $p(y\,|\,s,x)$. The relation between the random variables is given by the Markov chain $V_0, V_1 \longleftrightarrow V \longleftrightarrow U \longleftrightarrow S, X \longleftrightarrow Y$.

We first examine the probability of an encoder error. As in the Gaussian dirty paper case, with probability that approaches 1 with n, $(\mathbf{c}_0, \mathbf{c}_1, \mathbf{s})$ are jointly strongly ϵ-typical if

$$R_0 > I(V_0; V_1, S) + \epsilon_1 \tag{37}$$

It can be shown that

$$I(V_0; V_1, S) = I(U, S)$$

(using the fact that V_0 and V are identically distributed so that $h(V) = h(V_0)$, the fact that V and S are independent of V_1, so that $h(V \mid V_1, S) = h(V \mid S)$, and using the Markov relation $V \longleftrightarrow U \longleftrightarrow S$ and the relation $I(S; U \mid V) \leq h(U \mid V) = 0$) Thus, given (18), for small enough ϵ, (37) is satisfied and the probability of an encoder error approaches zero.

We now examine the probability of a decoder error.

1. As in the Gaussian dirty paper case, with a probability that approaches 1 with n, $\mathbf{y}$ will be jointly strongly ϵ-typical with $\mathbf{c}_0$ and $\mathbf{c}_1$.

2. We proceed to examine the event that there exist codewords $\mathbf{c}'_0 \in \mathcal{C}_0$ and $\mathbf{c}'_1 \in \mathcal{C}_1$, other than the transmitted $\mathbf{c}_0$ and $\mathbf{c}_1$, that are jointly strongly ϵ-typical with the channel output $\mathbf{y}$. As in the Gaussian case, we are faced with a multiple access channel from V_0, V_1 to Y. We can again apply Lemma 4 and obtain that for small enough ϵ, the probability of a decoder error approaches 0 with n if (R_0, R_1) fall within the achievable region specified by the inequalities (32) (replacing U_0 and U_1 with V_0 and V_1). It can be shown that,

$$I(V_0; Y \mid V_1) = I(U;Y), \quad I(V_1; Y \mid V_0) = I(U;Y) - I(V_0;Y), \quad I(V_0, V_1; Y) = I(U;Y) \tag{38}$$

(to derive the first equality we use the fact that Y is independent of V_1, and therefore $h(Y \mid V_1) = h(Y)$). Thus the achievable region coincides with (19) and (20). □

F.2 Review of Choices for $p(v_1)$ that render $I(V_0;Y) \leq I(U;S)$

One option that renders $I(V_0;Y) \leq I(U;S)$ is to make V_1 uniformly distributed in $[-A/2, A/2]$. With this choice, $I(V_0;Y) = 0 < I(U;S)$ as desired.

Another option is to make V_1 Gaussian distributed with variance Q, such that $I(V_0;Y) = I(U;S)$. To see that this is possible, observe that if $Q = 0$ then V_1 is deterministic and hence independent of V_0 and Y. Therefore,

$$I(V_0;Y) = I(V_0;Y \mid V_1) = I(U;Y)$$

The last equality follows from (38). The Gel'fand-Pinsker capacity is given by $I(U;Y) - I(U;S)$. We assume this capacity to be positive, and hence we have $I(V_0;Y) = I(U;Y) > I(U;S)$ under the assumption of $Q = 0$. As $Q \to \infty$, we approach the above uniformly distributed case, i.e. $I(V_0;Y) < I(U;S)$. Therefore, by a continuity argument there must exist some Q such that $I(V_0;Y) = I(U;S)$. Such a selection would produce a MAC achievable region where the Gel'fand-Pinsker capacity is achieved at a vertex point.

Acknowledgment

The authors would like to thank the anonymous reviewers and the associate editor for their comments and help.

References

[1] A. Amraoui, S. Dusad and R. Urbanke, "Achieving general points in the 2-user gaussian mac without time-sharing or rate-splitting by means of iterative coding," In *Proc. Int. Symp. Information Theory*, Lausanne, Switzerland, June 30–July 5 2002, pp. 334.

[2] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal Decoding of Linear Codes for Minimizing Symbol Error Rate," *IEEE Trans. on Inform. Theory*, Vol. 20, No. 2, March, 1974.

[3] R. J. Barron, B. Chen, and G. W. Wornell, "On the duality between information embedding and source coding with side information and some applications," in *Proc. Int. Symp. Information Theory,*Washington DC, June 2001, pp. 300.

[4] A. Bennatan and D. Burshtein, "On the Application of LDPC Codes to Arbitrary Discrete-Memoryless Channels", *IEEE Trans. on Inform. Theory*, Vol. 50, No. 3, March, 2004.

[5] A. Bennatan and D. Burshtein, “Design and Analysis of Nonbinary LDPC Codes for Arbitrary Discrete-Memoryless Channels”, to appear in *IEEE Trans. on Inform. Theory,* February 2006.

[6] C. Berrou, A. Glavieux and P. Thitimajshima, “Near Shannon limit error-correcting coding and decoding: Turbo-codes,” *IEEE Intern. Conf. on Commun. ICC ’93*, pp. 1064-1070, Geneva, Switzerland, May 1993.

[7] R.S. Blum, Z. Tu and J. Li, “How Optimal is Algebraic Binning Approach: A Case Study of the Turbo-Binning Scheme With Uniform and Nonuniform Sources,” *38th Annual Conference on Information Sciences and System* (CISS 2004), March 14–19, 2004, Princeton, N.J., USA.

[8] J. Boutros and G. Caire “Iterative Multiuser Joint Decoding: United Framework and Asymptotic Analysis,” *IEEE Trans. on Inform. Theory,* Vol. 48, No. 7, July 2002.

[9] G. Caire, D. Burshtein and S. Shamai, “LDPC Coding for Interference Mitigation at the Transmitter”, in *40th Annual Allerton Conf. on Commun., Cont. and Comp.*, Monticello, IL, October 2002.

[10] G. Caire, R. Müller and T. Tanaka, “Iterative multiuser joint decoding: Optimal power allocation and low-complexity implementation,” *IEEE Trans. on Inform. Theory*, September 2004.

[11] G. Caire, S. Shamai, “On the achievable throughput of a multiantenna Gaussian broadcast channel,” *IEEE Trans. Inform, Theory*, Vol. 49, No. 7, pp. 1691–1706, July 2003.

[12] G. Caire, A. Bennatan, D. Burshtein and S. Shamai, “Coding Schemes for the Binary-Symmertic Channel with Known Interference”, *The 41st Annual Allerton Conference on Commun., Control and Computing*, Monticello, IL, Oct. 1-3, 2003.

[13] N. Chayat, S. Shamai, “Convergence properties of iterative soft onion peeling” *Proc. IEE Information Theory and Communications Workshop*, 1999.

[14] B. Chen, G. W. Wornell, “Achievable performance of digital watermarking systems,” In *Proc. of the IEEE Intl. Conference on Multimedia Computing and Systems (ICMCS 99)*, vol. 1, pp. 13-18, Florence, Italy, June 1999.

[15] S.-Y. Chung, J. G. D. Forney, T. Richardson and R. Urbanke, “On the design of low-density parity-check codes within 0.0045 dB of the Shannon limit”, *IEEE Commun. Lett.*, vol. 5, pp. 58–60, February 2001.

[16] A. S. Cohen and A. Lapidoth, “The Gaussian Watermarking Game” *IEEE Trans. on Inform. Theory* vol. 48, no. 6, June 2002.

[17] M. Costa, “Writing on dirty paper,” *IEEE Trans. on Inform. Theory*, vol. 29, no. 3, pp. 439–441, May 1983.

[18] T.M. Cover and J.A. Thomas, *Elements of Information Theory*, John Wiley and Sons, 1991.

[19] I. J. Cox, M. L. Miller, A. L. McKellips, "Watermarking as Communications with Side Information," In *Proc. of the IEEE, Special Issue on Identification and Protection of Multimedia Information*, pp. 1127-1141, July 1999.

[20] R. Dobrushin, "Asymptotic optimality of group and systematic codes for some channels," *Theor. Probab. Appl.,* Vol. 8, pp. 52-66, 1963.

[21] U. Erez, S. Shamai and R. Zamir, "Capacity and Lattice-Strategies for Cancelling Known Interference", *IEEE Trans. on Inform. Theory*, November 2005.

[22] U. Erez, S. Litsyn and R. Zamir, "Lattices which are good for (almost) everything", submitted for publication *IEEE Trans. on Inform. Theory*, October 2005.

[23] U. Erez and S. ten Brink, "A close-to-capacity dirty paper coding scheme," *IEEE Trans. on Inform. Theory*, October 2005.

[24] U. Erez and R. Zamir, "Lattice Decoding Can Achieve $1/2\log(1+\mathrm{SNR})$ on the AWGN Channel Using Nested Codes," *ISIT'01*, p. 125, Washington, DC., June 24–29, 2001.

[25] R.F.H. Fischer, R. Tzschoppe and R. Bauml, "Lattice Costa Scheme using Subspace Projection for Digital Watermarking," European Transactions on Telecommunications (ETT), Vol. 15, No. 4, pp. 351-361, July/August 2004.

[26] G. D. Forney, Jr. and G. Ungerboeck, "Modulation and Coding for Linear Gaussian Channels", *IEEE Trans. on Inform. Theory*, vol. 44, pp. 2384–2415, October 1998.

[27] C. Fragouli, R. D. Wesel, D. Sommer, and G. Fettweis, "Turbo codes with non-uniform QAM constellations". In *Proc. IEEE ICC*, vol. 1, pp. 70–73. Helsinki, Finland, June 2001.

[28] P. Frenger, P. Orten, T. Ottosson, and A. Svensson, "Multirate convolutional codes," Tech. Rep. 21, Dept. of Signals and Systems, Communication Systems Group, Chalmers University of Technology, Goteborg, Sweden, Apr. 1998.

[29] S. Gel'fand and M. Pinsker, "Coding for channel with random parameters," *Problems of Control and Information Theory*, vol. 9, no. 1, pp. 19–31, January 1980.

[30] D. Hoesli, "On the Capacity per Unit Cost of the Dirty Tape Channel," Winter School on coding and Information Theory, Monte Verita, Switzerland, Feb. 24-27, 2003.

[31] F. R. Kschischang, B. J. Frey and H.-A. Loeliger, "Factor graphs and the sum-product algorithm", *IEEE Trans. on Inform. Theory*, vol. 47, pp. 498–519, February 2001.

[32] A. Kusnetsov and B. Tsybakov, "Coding in a memory with defective cells," *Probl. Pered. Inform.*, vol. 10, no. 2, pp. 52–60, 1974.

[33] A. Lapidoth, "Mismatched Encoding in Rate Distortion Theory," *1994 IEEE-IMS Workshop on Information Theory and Statistics*, 27–29, October 1994, p. 67.

[34] S-C Lin and H-J Su, “Vector Superposition Dirty Paper Coding for MIMO Gaussian Braodcast Channels”, *2005 Conference on Information Sciences and Systems (CISS 2005)*, The Johns Hopkins University, March 16-18, 2005.

[35] T. Liu and P. Viswanath, “Opportunistic Orthogonal Writing on Dirty Paper,” submitted to *IEEE Trans. on Inform. Theory.*

[36] P. Moulin, J. A. O’Sullivan, “Information-theoretic analysis of information hiding,” in *IEEE Transactions on Information Theory*, vol. 49, pp. 563-593, March 2003.

[37] T. Philosof, “Precoding for Interference Cancellation at Low SNR” , MSc. dissertation, Jan 2003.

[38] T. Philosof, U. Erez and R. Zamir “Combined Shaping and Precoding for Interference Cancellation at Low SNR” *Proc. 2003 IEEE International Symposium on Information Theory*, July 2003.

[39] S. Pradhan, J. Chou and K. Ramchandran, “Duality between source coding and channel coding and its extension to the side information case,” *IEEE Trans. on Inform. Theory*, vol.49, No. 5, pp. 1181–1203, May 2003.

[40] T. Richardson, A. Shokrollahi and R. Urbanke, “Design of capacity-approaching irregular low-density parity-check codes,” *IEEE Trans. on Inform. Theory*, vol. 47, pp. 619–637, February 2001.

[41] P. Robertson and T. Wörts, “Bandwidth-efficient Turbo trellis-coded modulation using punctured component codes”, *IEEE J. Select. Areas Commun.* , vol. 16, pp. 206–218, February 1998.

[42] A. Sanderovich, M. Peleg and S. Shamai, “LDPC Coded MIMO Multiple Access Communications,” The 2004 International Zurich Seminar on Communications (IZS) ETH Zurich, Switzerland, February 18–20, 2004, pp. 106–110.

[43] C. Shannon, “Channels with side information at the transmitter,” *IBM J. Res. & Dev.*, pp. 289–293, 1958.

[44] F-W. Sun and H. C. A. van Tilborg, “Approaching Capacity by Equiprobable Signaling on the Gaussian Channel,” *IEEE Trans. Inform. Theory*, vol. IT-39, pp. 1714-1716, September 1993.

[45] Y. Sun, A.D. Liveris, V. Stankovic and Z. Xiong, “Near-Capacity Dirty-Paper Code Design: A Source-Channel Coding Approach,” *2005 Conference on Information Sciences and Systems, (CISS2005)*, The John Hopkins University, March 16-18, 2005.

[46] E. Telatar, “Capacity of multi-antenna Gaussian channels,” *Eur. Trans. Telecomm. ETT*, vol. 10, no. 6, pp. 585-596, Nov. 1999.

[47] S. ten Brink, G. Kramer, and A. Ashikhmin,, “Design of Low-Density Parity-Check Codes for Modulation and Detection,” *IEEE Trans. Communications*, vol. 52, pp. 670-678, April 2004.

[48] G. Ungerboeck, “Channel coding with multilevel/phase signals,” *IEEE Trans. on Inform. Theory*, vol. 28, no. 1, pp. 55–67, January 1982 .

[49] S. Verdú, "On Channel Capacity per Unit Cost," *IEEE Trans. on Inform. Theory,* Vol. 36, No. 5, pp. 1019–1030, September 1990.

[50] S. Vishwanath, N. Jindal, A. Goldsmith, "Duality, achievable rates, and sum-rate capacity of Gaussian MIMO broadcast channels," *IEEE Trans. on Inform. Theory*, vol. 49, pp. 2658–2668 Oct. 2003.

[51] H. Wang and P. Viswanath, "Fixed Binning Schemes for Channel and Source Coding Problems An Operational Duality," *38th Annual Conference on Information Sciences and System* (CISS 2004), March 14–19, 2004, Princeton, N.J., USA.

[52] H. Weingarten, Y. Steinberg and S. Shamai (Shitz), "The Capacity Region of the Gaussian MIMO Broadcast Channel," *38th Annual Conference on Information Sciences and System* (CISS 2004), March 14–19, 2004, Princeton, N.J., USA.

[53] A. D. Wyner, "Recent results in the Shannon theory," *IEEE Trans. Inform. Theory*, vol. IT-20, pp. 2-10, Jan. 1974.

[54] R. Zamir, S. Shamai, and U. Erez, "Nested linear/lattice codes for structured multiterminal binning," *IEEE Trans. on Inform. Theory*, vol. 48, no. 6, pp. 1250–1276, June 2002.