Dynamic shift-map coding with side information at the decoder

Yongseok Yoo*†, O. Ozan Koyluoglu†, Sriram Vishwanath†, and Ila Fiete*
* Center for Learning and Memory, The University of Texas at Austin,
1 University Station C7000, Austin, TX 78712
† Department of Electrical & Computer Engineering, The University of Texas at Austin,
1 University Station C0806, Austin, TX 78712
Emails: ys.yoo@utexas.edu, ozan@mail.utexas.edu, sriram@ece.utexas.edu, ilafiete@clm.utexas.edu

***Abstract*—Shift-map codes have been studied as joint source-channel codes for continuous sources. These codes are useful in delay-limited scenarios and also provide better tolerance to deviations of the signal-to-noise ratio (SNR) from a target SNR, compared to separate source and channel coding. This paper defines a generalized family of shift-map codes that share a strong connection with redundant residue number systems (RRNS), and are henceforth called RRNS-map codes.**

In the proposed coding scheme, side information about the source allows the decoder to consider only a fraction of the codebook for decoding, with no change in the encoding process. With an appropriately designed RRNS-map code, in this fraction of the codebook, the codewords are much better separated than the original codebook. As a result, RRNS-map codes achieve the same distortion in the mean square error sense as conventional shift-map codes without side information, but significantly outperform shift-map codes when side information is provided to the decoder. This coding scheme is ideally suited to applications where a simple and fixed encoding scheme is desired at the encoder, while the decoder is given access to side information about the source.

I. Introduction

Analog codes, where both the input and output fields for the code correspond to points on the real line, have gained particular attention in recent years. Such codes differ from more conventional coding ensembles where typically one of the input or output sets is discrete-valued. Such analog codes have found applications in multiple domains (see, e.g., [1] and [2] for details). The two domains of particular interest in this paper are: 1) joint source channel coding of a continuous source over an additive Gaussian noise (AGN) channel, and 2) coding for spatial location in the brain, as studied in neuroscience [2]–[4].

The need for analog codes in joint source-channel coding applications is fairly well understood. Indeed, in many cases, it is known that uncoded transmission (uncoded corresponds to a trivial analog code) of continuous source(s) over classes of AGN channels outperforms separate source and channel coding [5], [6], and is sometimes optimal [7], [8]. Joint source-channel codes in general, and analog coding strategies in particular, are of interest for multi-terminal problems where separation of source and channel coding does not hold. However, even for the point-to-point case where source-channel separation holds, analog coding is of interest from other perspectives, such as simplicity, ease-of-use as well as coding under delay constraints and time-varying signal-to-noise ratio (SNR). For these reasons, as well as being a good starting point, we study analog codes for joint source-channel coding over point-to-point AGN channels in this paper.

Analog codes for error control are also of considerable interest from a neuroscience perspective. The brain often stores and computes with analog variables, such as the hue of an item, the speed of motion of a target, the orientation of an object, and so on. In addition, computation and communication in the brain are noisy processes [9], [10]. Recently, it was discovered that neurons called grid cells in the entorhinal cortex, part of the hippocampal circuit known to be the seat of learning and memory [11] and navigational computations [12], encode the location of animals in space using a particular analog code. N groups of neurons represent 2-d location as a set of N distinct 2-d modulo residues, given by the 2-d location phase relative to a periodic 2-d lattice [3]. This code was shown to have error-control properties [2], [4]. In particular, this code may be viewed as an example of a shift-map code [13]–[15], where the difference between this grid cell code and conventional shift-map codes is the specific choice of moduli.

Among analog coding schemes, the shift-map code has received particular attention in the field of communication and information theory. This is in part due to its applicability to joint source-channel coding for delay-limited applications as well as for channels with time-varying signal-to-noise ratio (SNR). The shift-map code's tolerance to SNR mismatch is an advantage over digital coding schemes where input is first quantized to discrete codewords and then transmitted by a channel code [13]–[15]. Separate source and channel coding is known to be optimal for a given SNR, but the performance of separation is not robust to SNR mismatch [13]. Joint source-channel coding with shift-map codes can render a degree of robustness to the system, and hence they have been analyzed in certain detail in existing literature.

In this paper, we generalize the notion of shift-map codes to include codes similar to the grid cell code, and show that this generalized family of codes enables the decoder to

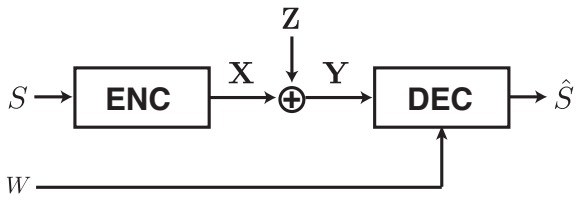

Fig. 1. Schematic diagram of the system model. The encoder generates codewords $\mathbf{X}$ for source $S \in [0,1]$, which are transmitted over an AGN channel. In addition to the noisy observation $\mathbf{Y}$, the decoder has additional knowledge (*side information*) that the source S lies in a subinterval $W = [S_l, S_u]$ which is contained in $[0,1]$.

exploit the side information available only at the decoder without modifying the encoding algorithm. This scenario corresponds to the Wyner-Ziv source coding problem over a point-to-point communication channel [16]–[19]. We find that our generalized shift-map codes can be used to enable robust joint source-channel coding over such a system. Given that our generalization of shift-map codes share a strong connection with redundant residue number systems (RRNS) [20], we refer to our codes as RRNS-map codes.

The organization of this paper is as follows. Section II describes the system model. In Section III, the shift-map code is reviewed and generalized to the RRNS-map code. The properties of the proposed construction are studied in Section IV. Dynamic decoding with side information and an example are provided in Section V. Finally, in Section VI, we conclude the paper by offering the implications of the new coding scheme and future research directions.

II. System Model

Fig. 1 presents our system model. The continuous source S is assumed to be uniformly distributed in the unit interval $[0,1]$. This source is encoded into N real-valued variables $\mathbf{X} = (X_1, X_2, \ldots, X_N)$. This codeword is transmitted over an additive Gaussian noise (AGN) channel with the power constraint $E\left[(X_i)^2\right] \leq 1$ for $i = 1, 2, \ldots, N$. The received signal $\mathbf{Y} = (Y_1, Y_2, \ldots, Y_N)$ is the sum of the transmitted codeword and noise $\mathbf{Z} = (Z_1, Z_2, \ldots, Z_N)$: For $i = 1, 2, \ldots, N$,

$$Y_i = X_i + Z_i \quad (1)$$
$$Z_i \sim \mathcal{N}(0, \sigma^2), \quad (2)$$

where the noise terms, i.e., Z_i's, are zero-mean Gaussian random variables with variance σ^2 and independent of one another and of the source.

The decoder has access to the *side information* that the source S lies in a subinterval $[S_l, S_u] \subseteq [0,1]$. This side information is assumed to be known to the decoder (receiver) but not the encoder (transmitter), in a setting analogous to a source coding with side-information problem [16], [17]. Here, the side information may come from another sensing modality, or the decoder could have kept track of the source to gain a priori knowledge about the support of the source.

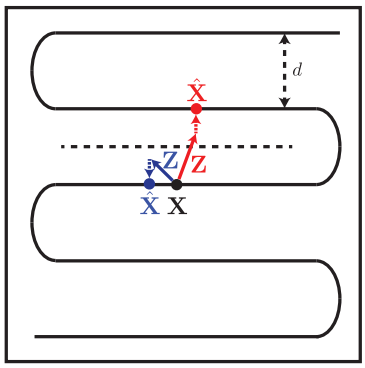

Fig. 2. The notion of threshold error in dimension-expansion mappings. A source S of dimension 1 (e.g., the unit interval) is mapped to a codeword $\mathbf{X}$ of higher dimension. This mapping may be viewed as an embedding of a curve in a higher dimensional space. The length of the embedded curve is L. If distant segments of the embedded curve are separated by a minimum distance d, then noise in a ball of radius $d/2$ (e.g., blue vector) is decoded as a point on the correct line segment, and the resulting error is a local error. In contrast, a larger noise (e.g., red vector) is decoded to the adjacent segment, and this phenomenon is referred to as the threshold error.

The decoder uses the side information and the channel output $\mathbf{Y}$ to produce an estimate of S, denoted as $\hat{S}$. The distortion D is defined as the mean square error in estimation:

$$D = E\left[\left(\hat{S} - S\right)^2\right]. \quad (3)$$

The distortion defined in (3) generally has two distinct components (Fig. 2). The first is due to codewords that are mapped by noise to other codewords that represent nearby points in the source. These errors occur frequently but the magnitudes are small. The second is due to noise that maps a codeword to another that represents a distant point in the source. These errors, called *threshold errors* by Shannon [21], are rare if the noise variance is small, but the magnitudes are large (Fig. 2). By considering the probability of each case and the corresponding mean square error, the distortion D in (3) is given by:

$$\begin{aligned} D &= P(\mathbf{Z} \notin \mathcal{T}) E\left[(\hat{S} - S)^2 | \mathbf{Z} \notin \mathcal{T}\right] \\ &\quad + P(\mathbf{Z} \in \mathcal{T}) E\left[(\hat{S} - S)^2 | \mathbf{Z} \in \mathcal{T}\right], \end{aligned} \quad (4)$$

where $\mathcal{T}$ represents the set of noise vectors that produce threshold errors.

III. The Shift-Map and Its Generalization

A. The shift-map code

In shift-map based coding schemes, the first element of the codeword X_1 is initialized by the source and the remaining elements X_i for $i > 1$ are generated by iterative scaling followed by modulo operation [22]:

$$X_i = \begin{cases} b_i S & \mod 1 \quad \text{if } i = 1 \\ b_i X_{i-1} & \mod 1 \quad \text{if } i > 1 \end{cases} \quad (5)$$

where b_i is a positive integer. As X_i ranges from 0 to 1, X_{i+1} sweeps the same interval b_i times. Thus, (5) is equivalent to the following non-iterative definition:

$$X_i = a_i S \bmod 1 \quad (6)$$

$$a_i = \prod_{j=1}^{i} b_j \quad (7)$$

A conventional way of choosing b_i is that $b_1 = 1$ and $b_j = \alpha$ for $j > 1$ where α is an integer greater than 1. Thus,

$$a_i = \alpha^{i-1}. \quad (8)$$

All the coding schemes based on the shift-map scheme as described above, called shift-map codes [13]–[15], fall into this category. In this paper, we start with the shift-map code as defined by (6) with a_i as in (8), then generalize this map to a new construction.

B. The distortion of the shift-map code

The set of codewords generated by the shift-map code forms parallel line segments with direction $(1, \alpha, \alpha^2, \ldots, \alpha^{N-1})$ inside the unit hypercube (Fig. 3 left panel). The total arc-lengh over all segments is L, given by

$$L(\alpha) = \sqrt{1 + \alpha^2 + \cdots + \alpha^{2(N-1)}}. \quad (9)$$

Thus, the unit source interval $[0, 1]$ is *stretched* along the ith dimension by the factor of α^{i-1}, and L^2 is called the *stretch factor* [23].

There is a tradeoff between the stretch factor and the minimum distance between the segments, which results in a tradeoff between the two distinct sources of distortion described above. For a given dimension N and a fixed noise variance σ^2, increasing $L(\alpha)$ by increasing α reduces the size of local distortions when there is no threshold error. This is because non-threshold errors correspond to decoding errors *along* the correct line segment, and a fixed length of coding line represents a smaller subinterval of the source with increasing $L(\alpha)$. Thus, local squared error, the first term in (4), scales as:

$$E\left[(\hat{S} - S)^2 | \mathbf{Z} \notin \mathcal{T}\right] \propto \frac{1}{L^2} \quad (10)$$

However, increasing the total length of the coding line means that the number of segments must increase and the distance d between them decreases. Specifically, the distance inversely scales with α [15],

$$d \approx \frac{1}{\alpha}. \quad (11)$$

Thus, the probability of threshold errors, which correspond to decoding on the wrong line segment, increases. As a result, the second term in the distortion of (4) grows with an increasing stretch factor.

The optimal choice of the stretch factor must therefore balance the two terms in (4): L should be as large as possible to reduce the common but small local errors, while remaining small enough to reduce the probability of very large threshold errors. Therefore, the parameter α of the shift-map code is chosen so that these two opposite effects are balanced and resulting L stays in an intermediate range.

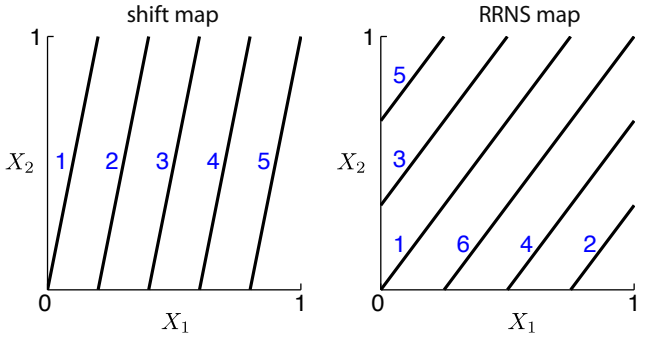

Fig. 3. An examples of the shift-map with $(a_1, a_2) = (1, 5)$ on the left and the *RRNS-map* with $(a_1, a_2) = (3, 4)$ on the right. Numbers next to individual segments in blue indicate the order of the mapping as the source marches from 0 to 1.

C. The RRNS-map: A generalization of the shift-map

We generalize the shift-map code by relaxing the conditions on the coefficients a_i. Instead of a_i being defined as the product of an integer b_i with the preceding coefficient a_{i-1} as in (7) or a geometric series as in (8), we choose each a_i independently from natural numbers. To be specific, a_i are chosen from any positive integers that are relatively prime: $\gcd(a_i, a_j) = 1$ if $i \neq j$, where gcd represents the greatest common divisor. The reason we have this primality constraint is as follows. If a_i and a_j $(i \neq j)$ share a common divisor other than 1, the mapping from S to $\mathbf{X}$ is not injective and the resulting map cannot be used for coding the full range of S. Representing a real number by its relationship to relatively prime numbers is, in principle, closely related to the redundant residue number system (RRNS) codes [20] in which an integer is encoded by their residues with respect to a set of relatively prime moduli.

In short, the *RRNS-map* is formulated as follows:

$$X_i = a_i S \mod 1 \quad (12)$$

$$\gcd(a_i, a_j) = 1 \quad \text{for } i \neq j, \quad (13)$$

where the scaling coefficients $1 < a_1 < \cdots < a_N$ are positive integers. We call the family of codes generated by this map *RRNS-map codes*.

In the shift-map codes, X_i is related to S by a multiplying factor $\alpha^{(i-1)}$, and X_i with larger i encodes local changes of the source with larger sensitivity. This could be a potential problem in the presence of noise: X_i with i larger than a certain value may not convey any information about the source due to the larger sensitivity to noise. On the other hand, in RRNS-map codes, one can choose a_i's within a small range so that all the X_i's contain non-vanishing information about the source.

Fig. 3 compares a shift-map code (left panel) and a RRNS-map code (right panel) with similar stretch factors. The

numbers in blue next to line segments indicate the order of the encoding line segments, as the source point moves from 0 toward 1. In the conventional shift-map code, X_i monotonically increases, and the resulting line segments are ordered sequentially. However, in the RRNS-map code, the order may be interleaved. As we will see next, this interleaving is a key advantage of the RRNS-map codes over conventional shift-map codes, because it allows the decoder to exploit side information by decreasing the probability of threshold errors while maintaining the size of local errors.

For example, let's compare the two cases shown in Fig. 3: The shift-map code has $(a_1, a_2) = (1, 5)$ (left panel) and the RRNS-map code has $(a_1, a_2) = (3, 4)$ (right panel). Both codes have essentially the same stretch factor ($L^2 = 1+5^2 \approx 3^2 + 4^2$) and minimum distances (0.20) between line segments. Therefore, without additional information about the source, decoders for both codes have the same performance. Now, suppose the decoders are informed that the source lies in the subinterval $[0.5, 0.75]$, corresponding to segments 3 and 4 for the shift-map code and segments 4 and 5 for the RRNS-map code. In the shift-map code, the distance between the candidate segments remains the same because they lie next to each other. In contrast, the distance between candidate segments is much larger in the RRNS-map code because they are non-consecutive, resulting in a much lower threshold error.

Thus, the design goal for the RRNS-map code is two-fold. The first is to achieve well-spaced segments with the same or approximately the same stretch factor as the shift-map counterpart, to obtain the same distortion in the absence of side information. The second is to interleave the order of the coding segments so that neighboring segments encode distant subintervals of the source and conversely, so that contiguous intervals of the source are encoded by distant coding segments. When this interleaving property is combined with the side information, the effective minimum distance between coding segments increases without a decrease in local stretch factor, and, consequently, the distortion decreases.

In the next section, we study the properties of the RRNS-map code in more detail. Decoding with side information at decoder is discussed in the section that follows.

IV. The RRNS-map code

A. Geometric interpretation: "Line packing" problem

From a geometrical perspective, finding a good RRNS-map code satisfying goals stated above is the problem of finding the direction $\mathbf{a} = (a_1, a_2, \ldots, a_N)$ of the line such that it winds around the N-dimensional torus (or hypercube with periodic boundary conditions), with maximal separation between segments as viewed in the hypercube. As explained earlier, choosing coordinates of $\mathbf{a}$ with relatively prime integers ensures that the line does not sweep the same point more than once. This line packing problem is a number theoretic problem that involves searching over the domain of relatively prime integers. We will deal with this problem numerically, after defining the search domain and the objective as follows.

The search domain is the set of vectors with relatively prime numbers as their coordinates, denoted as $\mathcal{A}$:

$$\begin{aligned}\mathcal{A} &= \{(a_1, a_2, \ldots, a_N) | \text{gcd}(a_i, a_j) = 1 \text{ if } i \neq j; \\ &\quad a_i < a_j \text{ if } i < j\}, \end{aligned} \tag{14}$$

where the coordinates of the vector in $\mathcal{A}$ are ordered by size. One might think that the requirement of relatively prime coordinates is too stringent and that the size of the search space $\mathcal{A}$ itself would be small. However, it turns out that this is not the case. As the dimension N grows, the ratio of the points with relatively prime coordinates to the points with integer coordinates quickly approaches to one [24]. This means that the size of this search domain is as large as set of all the integer points.

To simplify the problem, we restrict the search range to those elements in $\mathcal{A}$ with coordinates that are not greater than a_{max}, denoted as

$$\begin{aligned}\mathcal{A}_{a_{max}} &= \{(a_1, a_2, \ldots, a_N) \mid \text{gcd}(a_i, a_j) = 1 \text{ if } i \neq j; \\ &\quad a_i < a_j \text{ if } i < j;\ a_i \leq a_{max} \text{ for } \forall i\} \end{aligned} \tag{15}$$

This restriction keeps stretch factors from being too large and resulting in a very small distance between coding segments with severe threshold errors and large distortion.

Next, the objective of the line packing problem is to find the direction of the interleaving coding line, $\mathbf{a} \in A_{a_{max}}$, that maximizes the distance between neighboring lines. Let us call the set of all the codewords generated by $\mathbf{a}$ as $\mathcal{X}$, which is the set of parallel line segments inside of the unit hypercube pointing direction $\mathbf{a}$. To find the minimum distance, let's consider the hyperplane inside the unit cube that is orthogonal to $\mathbf{a}$ and passes through the center of the hypercube, denoted by H. Algebraically, this hyperplane is

$$H = \left\{\mathbf{x} \in [0,1]^N \mid \mathbf{a} \cdot (\mathbf{x} - \mathbf{c}) = 0\right\}, \tag{16}$$

where $\mathbf{c} = \left(\frac{1}{2}, \frac{1}{2}, \ldots, \frac{1}{2}\right)$ is the center of the unit hypercube and $\cdot$ represents the inner product. Then, each line segment in $\mathcal{X}$ intersects with H at a point and set of those intersecting points are denoted as $\mathcal{X}^*$. Considering the codeword generated as $\mathbf{x} = \mathbf{a}s \mod 1$ from (12) and (13), we have the set of such intersections $\mathcal{X}^*$ given by:

$$\begin{aligned}\mathcal{X}^* &= H \cap \mathcal{X} \\ &= \left\{\mathbf{x} \in [0,1]^N \mid \mathbf{a} \cdot (\mathbf{x} - \mathbf{c}) = 0;\right. \\ &\quad \left.\mathbf{x} = \mathbf{a}s \bmod 1; 0 \leq s < 1\right\}. \end{aligned} \tag{17}$$

Consequently, maximizing the distance between the coding line is equivalent to maximizing the minimum distances between points in $\mathcal{X}^*$ in the $(N-1)$ dimensional space.

B. The global structure of $\mathcal{X}^*$

We study the properties of the projected points $\mathcal{X}^*$ for a given $\mathbf{a}$. Due to the modulo operation in (17), the structure of $\mathcal{X}^*$ may change significantly even for small change in $\mathbf{a}$. Thus, instead of directly dealing with (17), let's first consider the intersections of $\mathcal{X}$ and the faces of the unit hypercube and

then project those onto H. This will provide the cardinality of $\mathcal{X}^*$ as well as a procedure to identify all the points in $\mathcal{X}^*$.

First, we analyze the number of points in $\mathcal{X}^*$ for a given $\mathbf{a}$. Let H_i be the hyperplane that is orthogonal to the ith axis and contains the origin:

$$H_i = \left\{ \mathbf{x} \in \mathbb{R}^N | \mathbf{e_i} \cdot \mathbf{x} = 0 \right\}, \tag{18}$$

where $\mathbf{e_i}$ is the unit vector with zero in all the coordinates other than the ith coordinate. Then, $\mathcal{X}$ intersects with the face of the unit hypercube on H_i at the origin and at additional $(a_i - 1)$ points. This is summarized in the following lemma.

Lemma 1. *$\mathcal{X}$ intersects with H_i at a_i points with corresponding source $S = \frac{j}{a_i}, j = 0, 1, \ldots, a_i - 1$:*

$$\begin{aligned} \mathcal{X}_i &= H_i \cap \mathcal{X} \\ &= \left\{ \mathbf{a}S \ mod \ 1 \,\middle|\, S = \frac{j}{a_i}, j = 0, 1, \ldots, a_i - 1 \right\} \end{aligned} \tag{19}$$

and

$$|\mathcal{X}_i| = a_i \tag{20}$$

Proof: By the definition of $\mathcal{X}_i = H_i \cap \mathcal{X}$, $\mathcal{X}_i$ is the set of points in $\mathcal{X}$ with the ith coordinate being zero, which is equivalent to

$$a_i S = 0 \pmod 1. \tag{21}$$

Since $0 \leq S < 1$, $0 \leq a_i S < a_i$. Thus, (21) has a_i solutions of S, namely $S = \frac{j}{a_i}, j = 0, 1, \ldots, a_i - 1$. ∎

Next, the relative primality of a_i's in (13) implies that the projections of $\mathcal{X}_i \setminus \mathbf{0}$ onto H are disjoint as shown in the next lemma.

Lemma 2. *If $i \neq j$,*

$$\mathcal{X}_i \cap \mathcal{X}_j = \mathbf{0}, \tag{22}$$

where $\mathbf{0}$ represents the vector corresponding to the origin $(0, 0, \ldots, 0)$.

Proof: Trivially, $\mathbf{0} \in X_i$ for all $i = 1, 2, \ldots, N$. Next, it is shown by contradiction that there is no other element in the intersection. Suppose that there is an element other than $\mathbf{0}$ in the intersection: $\mathbf{x} \in \mathcal{X}_i \cap \mathcal{X}_j$ such that $\mathbf{x} \neq \mathbf{0}$. This implies that there exists $s \in (0, 1)$ satisfying (21) for both a_i and a_j. In other words, $a_i s = m$ and $a_j s = n$ with integers $0 \leq m < a_i$ and $0 \leq n < a_j$. Since a_i and a_j are relatively prime from (13), this can happen only when $s = 0$, which contradicts the assumption that $\mathbf{x} \neq \mathbf{0}$. ∎

Lemma 1 and 2 lead to the following theorem about the number of points in $\mathcal{X}^*$.

Theorem 1. *Given an RRNS-map code satisfying (13), the cardinality of $\mathcal{X}^*$, the intersection between the coding segments $\mathcal{X}$ and the orthogonal hyperplane H, is*

$$|\mathcal{X}^*| = 1 + \sum_{i=1}^{N} (a_i - 1) \tag{23}$$

Proof: From Lemma 1, $|\mathcal{X}_i^*| = a_i$. From Lemma 2, $\mathcal{X}_i^*$ are disjoint for distinct i except $\mathbf{0}$. Thus, the number of points in $|\mathcal{X}^*|$ other than the origin is given by adding the cardinality of non-origin points of $|\mathcal{X}_i^*|$ which is $(a_i - 1)$. Thus, (23) immediately follows by adding one (for the origin) to the sum. ∎

Similarly, all the points in $\mathcal{X}^*$ for a given $\mathbf{a}$ are identified by repeating the same procedure (first identifying $\mathcal{X}_i$ and then projecting those onto H) algebraically. Let $\mathbf{x}_{ij}$ be an element in X_i with corresponding source $S = \frac{j}{a_i}$, $j \neq 0$ from (19):

$$\mathbf{x}_{ij} = \left(\frac{a_1}{a_i} j, \ldots, \frac{a_{i-1}}{a_i} j, 0, \frac{a_{i+1}}{a_i} j, \ldots, \frac{a_N}{a_i} j \right) \bmod 1, \tag{24}$$

where $i = 1, 2, \ldots N$ and $j = 1, \ldots a_i - 1$. To project this point onto H, we find the orthogonal basis of H by finding the null space of $\mathbf{a}$. Considering $\mathbf{a}^T$ as a $(1 \times N)$ matrix and performing the singular value decomposition, we obtain

$$\mathbf{a}^T = \begin{bmatrix} a_1 & a_2 & \ldots & a_N \end{bmatrix} = \mathbf{U} \mathbf{\Sigma} \mathbf{V}^T, \tag{25}$$

where $\mathbf{\Sigma}$ contains only one non-zero singular value at the first row and all the other singular values are zero. Thus, the first column of $\mathbf{V}$ corresponds to the non-zero singular value and the remaining $(N-1)$ columns of $\mathbf{V}$ are the orthogonal basis of the null space of $\mathbf{a}$, denoted as a $N \times (N-1)$ matrix $\mathbf{B}$. Therefore, the projection of $\mathbf{x}_{ij}$ onto H is

$$\mathbf{x}_{ij}^* = \mathbf{B}^T (\mathbf{x}_{ij} - \mathbf{c}), \tag{26}$$

where $\mathbf{c} = \left(\frac{1}{2}, \frac{1}{2}, \ldots, \frac{1}{2} \right)$ is the center of the hypercube.

C. The local structure of $\mathcal{X}^*$ determines the probability of threshold error.

After all the points in $\mathcal{X}^*$ are identified from (24) and (26), the local structure between neighboring points in $\mathcal{X}^*$ determines the probability of threshold error. The Voronoi regions of point $\mathbf{x}_{ij}^* \in \mathcal{X}^*$ is defined as

$$\begin{aligned} V(\mathbf{x}_{ij}^*) &= \left\{ \mathbf{x} \in \mathcal{X}^* \,\middle|\, ||\mathbf{x} - \mathbf{x}_{ij}^*||_2 < ||\mathbf{x} - \mathbf{x}'||_2, \right. \\ &\qquad \left. \mathbf{x}' \in \mathcal{X}^*, \mathbf{x}' \neq \mathbf{x}_{ij}^* \right\}, \end{aligned} \tag{27}$$

where $|| \cdot ||_2$ is the L_2 norm. When the noise $\mathbf{Z}$ falls outside of this Voronoi region, a threshold error occurs. Thus, the probability of threshold error is given by the probability of the noise falling outside of the Voronoi region, which is an $(N-1)$ dimensional polytope.

The probability of threshold error can be computed based on the observation that the region outside of the Voronoi region is the union of the half spaces defined by the boundaries of the Voronoi region. Using the union bound, the probability of threshold error is bounded from above by the sum of individual probabilities. It is shown that this union bound is very tight in a high-dimensional space and used as an approximation of the probability of threshold error, called union bound estimate (UBE) [25]. In addition, this calculation involves only the number of neighbors and distances between them.

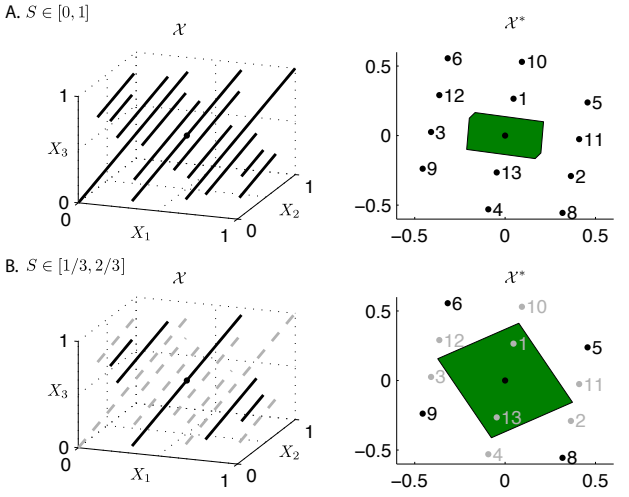

Fig. 4. Codeword $\mathcal{X}$ and its projection to the orthogonal plane $\mathcal{X}^*$ either without (A) and with (B) side information at decoder. Numbers in the right denote the order as the source increases from 0 and the green regions are the Voronoi regions of the center point.

To be specific, the union bound estimate of the probability of threshold error is calculated as follows. Suppose that true codeword $\mathbf{X}$ is in the ith segment in $\mathcal{X}$ and let E_j be the event that $\mathbf{X}+\mathbf{Z}$ is closer to another segment j than to segment i with probability P_j. Then the probability of threshold error is the probability that any of such event happens:

$$P(\mathcal{T}) \quad = \quad P\left(\bigcup_{j\neq i} E_j\right) = P\left(\bigcup_{j\in N(i)} E_j\right), \qquad (28)$$

where $N(i)$ represents the neighbors of i excluding i. Considering the union bound for (28), we have an upper bound on $P(\mathcal{T})$ with the union replaced by the summation of corresponding probabilities:

$$P(\mathcal{T}) \quad \leq \quad \sum_{j\in N(i)} P_j \qquad (29)$$

$$= \quad \sum_{j\in N(i)} \left(1-\Phi\left(\frac{d_j}{2\sigma}\right)\right) \qquad (30)$$

$$\equiv \quad P_{th}^U, \qquad (31)$$

where d_j is the distance to the jth neighbor and $\Phi(x)$ is the cumulative distribution function of the standard normal variable defined by $\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{x} e^{-\frac{t^2}{2}}dt$. In addition, if the distances to neighboring points are close to one another $d \approx d_j, j \in N(i)$, the union bound in (31) is further simplified to

$$P_{th}^U \quad \approx \quad K\left(1-\Phi\left(\frac{d}{2\sigma}\right)\right), \qquad (32)$$

where K is the number of neighbors.

V. Dynamic decoding with side information

Now, let's turn our attention to the role of the side information about the source S at decoder. Suppose that the decoder is informed that $S \in [S_l, S_u]$ with length $\Delta S = S_u - S_l \leq 1$. With this additional information about the source, the number of line segments considered for decoding decreases as ΔS decreases and, as a result, the effective minimum distance increases. Fig. 4 illustrates how the side information is used in decoding with an example of the 3-dimensional RRNS-map code with $(a_1, a_2, a_3) = (3, 5, 7)$. In Fig. 4A, there is no additional information about S. Thus, decoder should search for all the segments in $\mathcal{X}$ in the left panel. Corresponding $\mathcal{X}^*$ is shown in the right panel with the Voronoi region of the center point in green. In contrast, in Fig. 4B, additional information $S \in [1/3, 2/3]$ is provided to the decoder. Then, the decoder only needs to consider five segments shown in black in the left panel. The number of candidate segments for decoding decreases from 12 to 5. The right panel shows the projection of those five segments in black while gray points do not need to be considered any more. Because the segments are well interleaved, the reduced number of active points in $\mathcal{X}^*$ results in increased distance between each neighbor. Consequently, the Voronoi region of the center point expands. We refer to this phenomenon as the *dynamic* decoding with side information.

A. With side information, the minimum distance increases and the number of active points decreases.

Next, we quantify this scaling behavior as a function ΔS, the reduced range of input due to the side information at the decoder.

The most important property is the distance to the neighboring points because this determines the exponent of the probability of threshold error. We assume that the points in $\mathcal{X}_s^*$ are equally-spaced with minimum distance d in the $(N-1)$ dimensional hyperplane H. Under this condition, the volume of the Voronoi region multiplied by the number of the points is equal to the volume of H.

$$|\mathcal{X}_s^*|\text{Vol}(V_s) = \text{Vol}(H), \qquad (33)$$

where $|\mathcal{X}_s^*|$ is the number of active points, $\text{Vol}(V_s)$ is the volume of the Voronoi region, $\text{Vol}(H)$ is the volume of the orthogonal hyper plane H inside of the unit hypercube. Since the length of the source interval scales down by a factor of ΔS, the remaining active points also scales with the same factor:

$$|X_s^*| = |X^*|\Delta S. \qquad (34)$$

The volume of the Voronoi region, which is a polytope in the $(N-1)$ dimensional space, scales with $O(d^{(N-1)})$ while the volume of H is approximately constant [2]. Thus, we have the following scaling of d with respect to ΔS for a fixed dimension N:

$$d^{(N-1)} = \Omega(1/\Delta S). \qquad (35)$$

B. Finding RRNS-map codes with desired properties

We report that there exists an RRNS-map code with the minimum distance scaling as predicted in (35).

We performed an exhaustive search for all $\mathbf{a}$ with odd integers in $\mathcal{A}_{50}$ from (15) with $N = 5, a_{max} = 50$. Only the odd numbers are considered for a_i to fix a reference point from which the minimum distance and the number of neighbors are calculated, which is described as follows. The side information about the source interval is varied such that the length of the interval ΔS changes while its center remains at 0.5: $S \in [0.5 - \Delta S/2, 0.5 + \Delta S/2]$ with ΔS varying from 0.1 to 1 by increments of 0.1. Since all a_i's are odd integers, $S = 0.5$ always corresponds to the center of the code space $\mathbf{c} = (0.5, 0.5, \ldots, 0.5)$. For each $\mathbf{a}$ and ΔS, points in $\mathcal{X}^*$ are found from (24). The active points $\mathcal{X}_s^*$ corresponding the intervals are identified and the Voronoi region of the reference point $\mathbf{c}$ is calculated by an efficient algorithm in [26] ported into Matlab [27]:

$$V_s(\mathbf{c}) = \{\mathbf{x} \in \mathcal{X}_s^* \,|\, ||\mathbf{x} - \mathbf{c}||_2 < ||\mathbf{x} - \mathbf{x}'||_2, \; \mathbf{x}' \in \mathcal{X}_s^*, \mathbf{x}' \neq \mathbf{c}\}, \quad (36)$$

where $||\cdot||_2$ is the L_2 norm. Among $\mathcal{A}_{50}$ with odd integers and $N = 5$, $\mathbf{a} = (9, 11, 13, 17, 23)$ with $L = 34.5$ produces the largest minimum distance averaged over the range of ΔS considered.

Fig. 5 shows the minimum distance (left panel) and the number of neighbors (right panel) as functions of ΔS. Blue circles in the figure are numerical calculations and black lines are the analytical predictions from (35) and (34). In Fig. 5 left panel, the minimum distance increases stepwise with decreasing ΔS and overall scaling agrees well with the prediction from (35). In Fig. 5 right panel, even though there is a quite large fluctuation in the number of neighborhoods, the overall trend shows that $|\mathcal{X}_s^*|$ decreases as ΔS decreases from 1 towards 0. Shift-map codes with similar stretch factors, shown in red squares ($L = 18.5$ with $\alpha = 2$) and diamonds ($L = 86.0$ with $\alpha = 3$), have comparable minimum distances and the numbers of neighbors to those of the RRNS-map code without side information ($\Delta S = 1$). However, the minimum distances of the shift-map codes remain constant.

It is also interesting to compare the number of neighbors produced by the RRNS-map codes to that by regular lattices in the same dimension. Finding the regular lattice with the largest density in a given dimension is equivalent to finding a way to arrange as many spheres with the same radii as possible so that they barely touch ("kiss") with one another. The number of such spheres touching the one at the center is called the kissing number and finding the largest possible kissing number in a given dimension is *the kissing number problem* [28]. In the four dimensional space, the highest attainable kissing humber that is known is 24 with lattice D_4 [28], shown as cyan star in Fig 5 right panel. Interestingly, the RRNS-map code with this choice of $\mathbf{a}$ produces the number of neighbors K close to the kissing number of D_4 when $\Delta S = 1$ (without side information). This hints that the RRNS-map code with $\mathbf{a} = (9, 11, 13, 17, 23)$ is indeed well-spaced in the $(N-1)$ dimensional orthogonal hyperplane.

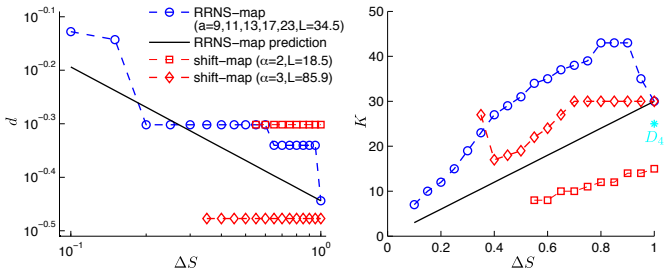

Fig. 5. Minimum distance (left) and the number of neighbors (right) of active points $\mathcal{X}_s^*$ for $\mathbf{a} = (9, 11, 13, 17, 23)$ as function of the interval length of side information. Blue circles indicate numerically found values and black lines indicate analytical predictions from (35) for left and (34) for right. Red squares and diamonds are for shift-map codes with similar stretch factors. The asterisk in cyan represents the highest value attainable by regular lattice in $(N-1) = 4$ dimensional space, which is 24 by lattice D_4 [28].

In the left panel of Fig. 6, the probabilities of threshold error for the RRNS-map code (blue circles) and the shift-map codes (red squares and diamonds) are shown for $\sigma = 0.05$. Those are union bounds calculated from (31) using numerically found minimum distances and the numbers of neighbors shown in Fig. 5. The black line shows the union bound for RRNS-map code from the analysis of the average number of neighbors and the minimum distance using (32). The analysis agrees well with the numerical results and shows that the probability of threshold error for the RRNS-map code drops as ΔS decreases. In contrast, the probability of threshold error stays constant for the shift-map codes.

Finally, the right panel of Fig. 6 shows that the RRNS-map code outperforms other analog codes when side information is provided to the decoder. The green dashed line in Fig. 6 right panel shows the average distortion of the repetition code: $\frac{\sigma^2}{N}$ with $N = 5$. The black cross represents the distortion of the tent-map code [13, Eq. (16a) and (16b)]. Lower bounds on the distortions of the shift-map codes are derived similarly to [15, Appendix B], which remain constant regardless of the amount of the side information at the decoder. Without the side information ($\Delta S = 1$), the upper bound of the distortion of the RRNS-code with $\mathbf{a} = (9, 11, 13, 17, 23)$ is close to the lower bounds of those shift-map codes with similar stretch factors. However, with additional side information ($\Delta S < 1$), the former becomes smaller than the latter; the RRNS-map code outperforms the shift-map code.

VI. Conclusions

Motivated by the way spatial location is encoded in the brain as well as the robust joint source channel problem with side information, we generalize shift-map codes in this paper to a new family, called RRNS-map codes. In RRNS-map codes, scaling coefficients are chosen as relatively prime numbers rather than a geometric series determined by a fixed integer. The new coding scheme based on the RRNS-map allows the decoder to exploit side information about

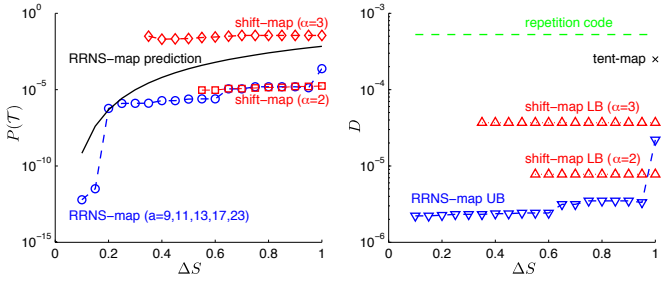

Fig. 6. Probability of threshold error for the RRNS-map code decreases as the decoder has access to more side information (decreasing ΔS) (left). The distortion of RRNS-map is compares to those of other analog codes (right). The distortion of RRNS-map code without any side information ($\Delta S = 1$) is similar to those of shift-map codes with similar stretch factor ($\alpha = 2, 3$). But, as more side information is revealed to the decoder (as ΔS decreases), the distortion of the RRNS-code decreases. This contrasts the other analog codes whose distortions remain constant regardless of side information.

the source to produce a lower threshold error while the encoder does not change its encoding policy. The threshold error of the RRNS-map code is lower than the corresponding conventional shift-map code, because the probability of threshold error drops with side information, and the distortion due to local error remains unchanged. Consequently, lower distortion is achieved with the knowledge of side information at the decoder.

Finding a good RRNS-map code is a line packing problem, which translates to solving a mixed integer program. We numerically solve this problem and find an example RRNS-map code that possesses a homogeneous structure that resembles a lattice. This code results in a superior performance when varying degrees of side information is made available to the decoder.

Acknowledgment

The authors would like to thank Dr. Goo Jun for helpful discussions and comments.

References

[1] T. Goblick, "Theoretical limitations on the transmission of data from analog sources," *Information Theory, IEEE Transactions on*, vol. 11, no. 4, pp. 558–567, Oct. 1965.

[2] S. Sreenivasan and I. Fiete, "Grid cells generate an analog error-correcting code for singularly precise neural computation," *Nature Neuroscience*, pp. 1330–1337, 2011.

[3] T. Hafting, M. Fyhn, S. Molden, M.-B. Moser, and E. I. Moser, "Microstructure of a spatial map in the entorhinal cortex," *Nature*, vol. 436, pp. 801–806, 2005.

[4] I. R. Fiete, Y. Burak, and T. Brookings, "What grid cells convey about rat location," *Journal of Neuroscience*, vol. 28, no. 27, pp. 6858–6871, 2008.

[5] T. Cover, A. Gamal, and M. Salehi, "Multiple access channels with arbitrarily correlated sources," *Information Theory, IEEE Transactions on*, vol. 26, no. 6, pp. 648–657, Nov. 1980.

[6] M. Gastpar, B. Rimoldi, and M. Vetterli, "To code, or not to code: lossy source-channel communication revisited," *Information Theory, IEEE Transactions on*, vol. 49, no. 5, pp. 1147–1158, May 2003.

[7] M. Gastpar, "Uncoded transmission is exactly optimal for a simple gaussian sensor network," *Information Theory, IEEE Transactions on*, vol. 54, no. 11, pp. 5247–5251, Nov. 2008.

[8] R. Soundararajan and S. Vishwanath, "Communicating the difference of correlated gaussian sources over a MAC," in *Data Compression Conference, 2009*, Mar. 2009, pp. 282–291.

[9] W. Softky and C. Koch, "The highly irregular firing of cortical cells is inconsistent with temporal integration of random epsps," *The Journal of Neuroscience*, vol. 13, no. 1, pp. 334–350, 1993.

[10] M. N. Shadlen and W. T. Newsome, "The variable discharge of cortical neurons: Implications for connectivity, computation, and information coding," *The Journal of Neuroscience*, vol. 18, no. 10, pp. 3870–3896, 1998.

[11] P. Andersen, Ed., *The Hippocampus Book*. Oxford University Press, 2007.

[12] J. O'Keefe and L. Nadel, "The hippocampus as a cognitive map," *Behavioral and Brain Sciences*, vol. 2, no. 4, pp. 487–494, 1979.

[13] B. Chen and G. Wornell, "Analog error-correcting codes based on chaotic dynamical systems," *Communications, IEEE Transactions on*, vol. 46, no. 7, pp. 881–890, July 1998.

[14] V. Vaishampayan and S. Costa, "Curves on a sphere, shift-map dynamics, and error control for continuous alphabet sources," *Information Theory, IEEE Transactions on*, vol. 49, no. 7, pp. 1658–1672, July 2003.

[15] M. Taherzadeh and A. Khandani, "Single-sample robust joint source: Channel coding: Achieving asymptotically optimum scaling of SDR versus SNR," *Information Theory, IEEE Transactions on*, vol. 58, no. 3, pp. 1565–1577, Mar. 2012.

[16] A. Wyner, "On source coding with side information at the decoder," *Information Theory, IEEE Transactions on*, vol. 21, no. 3, pp. 294–300, May 1975.

[17] A. Wyner and J. Ziv, "The rate-distortion function for source coding with side information at the decoder," *Information Theory, IEEE Transactions on*, vol. 22, no. 1, pp. 1–10, Jan. 1976.

[18] N. Merhav and S. Shamai, "On joint source-channel coding for the Wyner-Ziv source and the Gel'fand-Pinsker channel," *Information Theory, IEEE Transactions on*, vol. 49, no. 11, pp. 2844–2855, Nov. 2003.

[19] Y. Kochman and R. Zamir, "Joint Wyner-Ziv/dirty-paper coding by modulo-lattice modulation," *Information Theory, IEEE Transactions on*, vol. 55, no. 11, pp. 4878–4889, Nov. 2009.

[20] M. Soderstrand, F. Taylor, W. Jenkins, and G. Jullien, *Residue Number System Arithmetic: Modern Applications in Digital Signal Processing*, ser. IEEE PRESS Reprint Series. IEEE, 1986.

[21] C. E. Shannon, "Communication in the presence of noise," *Proceedings of the IRE*, vol. 37, no. 1, pp. 10–21, Jan. 1949.

[22] S. Strogatz, *Nonlinear Dynamics And Chaos: With Applications To Physics, Biology, Chemistry, And Engineering*. Westview Press, 1994.

[23] J. Ziv, "The behavior of analog communication systems," *Information Theory, IEEE Transactions on*, vol. 16, no. 5, pp. 587–594, Sep. 1970.

[24] M. Baake, U. Grimm, and D. H. Warrington, "Some remarks on the visible points of a lattice," *Journal of Physics A: Mathematical and General*, vol. 27, no. 8, 1994.

[25] J. Forney, G.D. and G. Ungerboeck, "Modulation and coding for linear gaussian channels," *Information Theory, IEEE Transactions on*, vol. 44, no. 6, pp. 2384 –2415, Oct. 1998.

[26] C. B. Barber, D. P. Dobkin, and H. Huhdanpaa, "The quickhull algorithm for convex hulls," *ACM Transactions on Mathematical Software*, vol. 22, no. 4, pp. 469–483, 1996.

[27] MATLAB, *version 7.14.0.739 (R2012a)*. The MathWorks Inc., 2012.

[28] J. H. Conway, N. J. A. Sloane, and E. Bannai, *Sphere Packings, Lattices, and Groups*, 3rd ed. Springer, 1999.