Multi-periodic neural coding for adaptive information transfer

Yongseok Yoo[a], O. Ozan Koyluoglu[b], Sriram Vishwanath[c], Ila Fiete[d,*]

[a] *ETRI Honam Research Center, 176-11 Cheomdan Gwagi-ro, Buk-gu, Gwangju, 500-480, Korea*
[b] *Department of Electrical and Computer Engineering, The University of Arizona, 1230 E Speedway Blvd., Tucson, AZ 85721, U.S.A.*
[c] *Department of Electrical & Computer Engineering, The University of Texas at Austin, 1 University Station C7000, Austin, TX 78712, U.S.A.*
[d] *Center for Learning and Memory, The University of Texas at Austin, 1 University Station C0806, Austin, TX 78712, U.S.A.*

Abstract

Information processing in the presence of noise has been a key challenge in multiple disciplines including computer science, communications, and neuroscience. Among such noise-reduction mechanisms, the shift-map code represents an analog variable by its residues with respect to distinct moduli (that are chosen as geometric scalings of an integer). Motivated by the multi-periodic neural code in the entorhinal cortex, i.e., the coding mechanism of grid cells, this work extends the shift-map code by generalizing the choices of moduli. In particular, it is shown that using similarly sized moduli (for instance, evenly and closely spaced integers, which tend to have large co-prime factors) results in a code whose codewords are separated in an interleaving way such that when the decoder has side information regarding the source, then error control is significantly improved (compared to the original shift map code). This novel structure allows the system to dynamically adapt to the side information at the decoder, even if the encoder is not privy to the side information. A geometrical interpretation of the proposed coding scheme and a method to find such codes are detailed. As an extension, it is shown that this novel code also adapts to scenarios when only a fraction of codeword symbols is available at the decoder.

Keywords: multi-periodic neural codes, shift-map codes, grid cells, the entorhinal cortex, side information

1. Introduction

The brain represents, stores, and computes with analog variables (e.g., determining and storing the orientation of an edge in the visual world, estimating and representing the speed of motion of a target, comparing the hue of an item with that of another) in the presence of noise in the basic processes of synaptic communication and response generation [1, 2, 3]. In these neural computations, representing one variable with a large number of neurons reduces the effects of the noise. In many brain areas, the mean firing rates of neurons vary in characteristic ways with the value of the represented variable, and these functions are called

*Corresponding author
Email addresses: yyoo@etri.re.kr (Yongseok Yoo), ozan@email.arizona.edu (O. Ozan Koyluoglu), sriram@ece.utexas.edu (Sriram Vishwanath), ilafiete@mail.clm.utexas.edu (Ila Fiete)

tuning curves. Many neurons in the sensory and the motor cortices have unimodal (or single-bump) tuning curves peaked at a certain value of the represented variable. The peak values of different neurons tile the range of the variable. This redundant representation, known as a population code, enables noise tolerance [4, 5, 6, 7].

In this work, we investigate the features of a code that is believed to underlie navigational computations in the brain. Our focus is on the inherent neural diversity of the tuning curve functions, and the functional advantage of this diversity from a coding-theoretic viewpoint. The hippocampus, which has long been implicated in spatial learning and memory functions and with the writing of new episodic memories, exhibits largely unimodal tuning curves for the 2-dimensional coordinate that represents the animal's location in space [8]. Grid cells [9], the focus of the present work, reside in medial entorhinal cortex (mEC), a high-order association area that is the main cortical input to hippocampus. Individual grid cells represent animal location with an interesting multi-peaked firing pattern, in which the peaks are arranged on every vertex of a virtual triangular lattice that tiles the explored space, Figure 1 [9]. Grid cells are organized in a number of distinct functional modules: in each module, grid cells have a common spatial firing period and orientation, but a diversity of spatial phases (that is, tuning curves of different cells in a module are rigid shifts of a cannonical lattice pattern). Different modules exhibit different spatial periods. Neurophysiological experiments show $4-5$ (and no more than 10) distinct modules, with an approximately geometric progression of grid periods involving a non-integer scale factor close to 1, of size ≈ 1.42 [9, 10]. A multi-period representation, as seen in the firing activity of the grid cells, is shown to have excellent representational and error-control properties [11, 12] compared to unimodal tuning curves, assuming the existence of an appropriate decoder. More specifically, the grid cell code exhibits an exponential coding capacity with linearly many neurons in the presence of noise, a qualitatively different result from previously characterized population codes from the sensory and motor peripheries [12].

In this work, we generalize the conventional notion of shift-map codes to include the grid cell code as an instance [13, 14, 15], where the difference between the grid cell code and the conventional shift-map codes is the choice of moduli. In particular, by choosing relatively prime integers as moduli, we extend the shift-map code to a novel form that resembles the grid cell code in the brain, and thus obtain a self-interleaving code (in the coding space). Given that this use of relatively prime integers shares a strong connection with redundant residue number systems (RRNS) [16, 17, 18, 19, 11], we refer to our codes as RRNS-map codes. Among our findings is the increased robustness of the RRNS-map code against noise compared to the conventional shift-map codes when side information (about the variable encoded) is present. Furthermore, the code is shown to be adaptive against noise and the quality of the side information. The contributions of this work include the following:

- We generalize the notion of shift-map codes to a new construction of analog code using relatively prime integers, referred to as the RRNS-map code.
- We design a set of RRNS-map codes whose codewords are well-separated. This novel structure allows

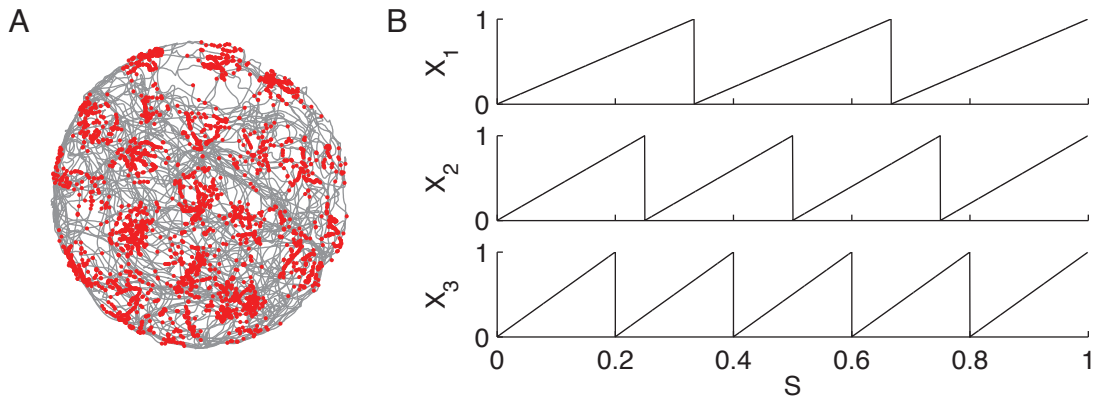

Figure 1: The response of a grid cell in the medial entorhinal cortex (mEC) is a periodic function of the animal location (A). Groups of grid cells (X_1, X_2, and X_3) encode the spatial location (S) with different periodicities (B).

the receiver to adaptively combine side information about the source for estimation, which offers an advantage over the traditional shift-map code.

- We offer a geometrical interpretation of the RRNS-map code.
- We provide a method to find RRNS-map codes with the desired properties. This method involves integer programming over relatively prime integers.
- As an extension, we show that the RRNS-map code is also adaptive for scenarios when only partial knowledge of the encoded variables is present.

The organization of the paper is as follows. Section 2 describes the system model. In Section 3, we briefly review the shift-map code and generalize it to include the RRNS-map code. In Section 4, the properties of the proposed construction are studied without side information. In Section 5, adaptive decoding of RRNS-map codes with side information is discussed and examples of good codes are provided. Extensions of the proposed code to different setups are discussed in Section 6. Concluding remarks are provided in Section 7.

2. System model

Figure 2 presents the system model. The continuous source S is uniformly distributed in the unit interval $[0, 1)$. This source is encoded by N real-valued variables $\mathbf{X} = (X_1, X_2, \ldots, X_N)$. This codeword is transmitted over additive Gaussian noise (AGN) channels with the constraint $0 \leq X_n < 1$ for $n = 1, 2, \ldots, N$[1]. The received signal $\mathbf{Y} = (Y_1, Y_2, \ldots, Y_N)$ is the sum of the transmitted codeword and noise

[1]With an appropriate offset, this constraint is equivalent to the power constraint $E\left[(X_n)^2\right] \leq \frac{1}{12}$ for $n = 1, 2, \ldots, N$.

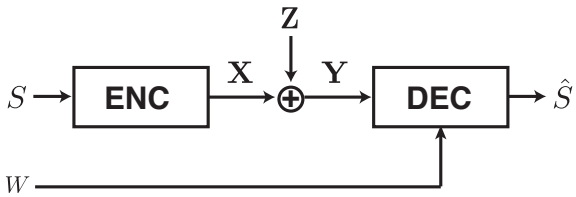

Figure 2: Schematic diagram of the system model. The encoder generates codewords $\mathbf{X}$ for a source $S \in [0,1)$, which are subsequently transmitted over AGN channels. In addition to the noisy observation $\mathbf{Y}$, the decoder has additional knowledge in the form of (*side information*) that the source S lies in a subinterval $W \subset [0,1]$ of the full source interval.

$\mathbf{Z} = (Z_1, Z_2, \ldots, Z_N)$. That is, for $n = 1, 2, \ldots, N$

$$Y_n = X_n + Z_n, \tag{1}$$

$$Z_n \sim \mathcal{N}(0, \sigma^2), \tag{2}$$

where the independent identically distributed (iid) noise terms Z_n are Gaussian with mean zero and variance σ^2. The receiver receives *side information*, that the source S lies in a subinterval $W \subset [0,1)$. This side information is assumed to be known to the receiver but not to the sender, in a setting analogous to the Wyner-Ziv source coding problem [20, 21]. The difference compared to [20, 21] is the channels, which in our model are AGN.

The side information in our model can be made available by an additional sensory mechanism at the receiver. For instance, the sensory mechanism may provide a correlated information source $S' \sim \mathcal{N}(S, \sigma_s^2)$ to the decoder. Here, if a subregion $W \in [0,1)$ can be declared such that $S \in W$ with high probability, then W can readily serve as side information in our model. The decoder uses this side information W 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 the estimate of S:

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

The distortion generally arises from two causes: The first is due to translation of the decoded estimate into a codeword near the true codeword, which represents a nearby value of the source (Figure 3, blue). These errors occur frequently but the magnitudes are small. The second is due to large but infrequent errors, in which noise maps a codeword to another one that represents a distant point in the source (Figure 3, red). These errors, called *threshold errors* by Shannon [22], are rare if the noise variance is small. By considering the probability of each case and the corresponding mean square error, the distortion D in (3) is given by:

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

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

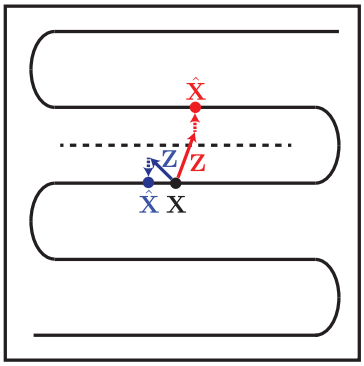

Figure 3: The notion of threshold error in dimension-expansion mappings. A source S of dimension 1 (e.g. the unit interval) is mapped to a codeword $\mathbf{X}$ of higher dimension. This mapping may be viewed as an embedding of a curve in a higher dimensional space. The length of the embedded curve is L and distant segments of the embedded curve are separated by a minimum distance d. A small noise (blue) is decoded as a point on the correct line segment, and the resulting error is small (local error). In contrast, a large noise (red) is decoded to the adjacent segment where the estimation error is large (threshold error).

3. The shift-map code and its generalization

We start by describing the shift-map code and then generalize it to a new construction.

3.1. The shift-map code

Given a continuous source S, the shift-map codeword of length N is defined as follows [13, 14, 15]:

Definition 1 (The shift-map code)**.** *For a source $S \in [0,1)$ and an integer $\alpha \geq 2$, the shift-map codeword of length N is given by $\mathbf{X}^{SM}(S) = [X_1^{SM}(S), X_2^{SM}(S), \ldots, X_N^{SM}(S)]$, where, for $n = 1, 2, \cdots, N$,*

$$X_n^{SM}(S) \quad = \quad \alpha^{(n-1)} S \bmod 1, \tag{5}$$

and $\alpha \geq 2$ is the design parameter that controls codeword scaling.

We denote the set of all the codewords by $\mathcal{X}^{SM}$ and the subset of codewords that represent values of the source in the subinterval $W \subset [0,1]$ as $\mathcal{X}^{SM}(W)$:

$$\mathcal{X}^{SM} \quad = \quad \left\{\mathbf{X}^{SM}(S) | S \in [0,1)\right\} \tag{6}$$

$$\mathcal{X}^{SM}(W) \quad = \quad \left\{\mathbf{X}^{SM}(S) | S \in W\right\}. \tag{7}$$

From a geometrical viewpoint, the shift-map codebook $\mathcal{X}^{SM}$ forms parallel line segments with direction $(1, \alpha, \alpha^2, \ldots, \alpha^{N-1})$ inside the unit hypercube (Figure 4A). The total arc-lengh over all these segments is called the stretch factor, denoted as $L^{SM}(\alpha)$, and the minimum distance between these segments is $d^{SM}(\alpha)$.

For a fixed N, there exists a trade-off between the stretch factor and the minimum distance, controlled by α: Increasing the total length of the coding line implies that a longer curve is packed into a fixed volume, and thus the minimum distance must decrease. Specifically, while $L^{SM}(\alpha)$ monotonically increases with α, the minimum distance $d^{SM}(\alpha)$ monotonically decreases. The following lemma quantifies the trade-off between the stretch factor and the minimum distance.

Lemma 1. *Given the shift-map code with the design parameter α, the stretch factor $L^{SM}(\alpha)$ and the minimum distance $d^{SM}(\alpha)$ are given by*

$$L^{SM}(\alpha) = \sqrt{1+\alpha^2+\cdots+\alpha^{2(N-1)}} \quad (8)$$

$$d^{SM}(\alpha) = \frac{1}{\alpha}\frac{\sqrt{(L^{SM}(\alpha))^2-1}}{L^{SM}(\alpha)}. \quad (9)$$

Proof. The stretch factor in (8) follows as the message interval of length 1 is *stretched* by α^{n-1} in the n'th dimension for each $n = 1, 2, \ldots N$. In order to calculate the minimum distance in (9), let us first consider distances from individual line segments to a reference line passing the origin. Finding intercepts of individual line segments simplifies the calculation as follows. Consider a non-zero codeword $\mathbf{X}$ with the n_o'th coordinate being zero for $1 < n_o \leq N$. By construction, $X_{n_o} = 0$ implies $X_n = 0$ for $n > n_o$. These intercepts correspond to the message $S = \frac{i}{\alpha^{(n_o-1)}}$ for a positive integer $0 < i < a_n$. Thus, other coordinates have the form of $X_n = \frac{\alpha^{(n-1)}i}{\alpha^{(n_o-1)}}$ for $n < n_o$. Among those points, $(\frac{1}{\alpha}, 0, \ldots, 0)$ attains the minimum distance in (9). □

For large α, the minimum distance scales inversely with α: $d \approx \frac{1}{\alpha}$. To be specific, we have the following corollary.

Corollary 1.

$$\frac{4}{5}\frac{1}{\alpha^2} \leq d^{SM}(\alpha)^2 < \frac{1}{\alpha^2}. \quad (10)$$

Proof. Equation (10) immediately follows from (9) by observing that

$$\frac{4}{5} \leq \frac{(L^{SM}(\alpha))^2-1}{(L^{SM}(\alpha))^2} = 1 - \frac{1}{(L^{SM}(\alpha))^2} < 1 \quad (11)$$

since $(L^{SM}(\alpha))^2 \geq 5$ for $\alpha \geq 2$. □

Importantly, the tradeoff between the stretch factor and the minimum distance results in a trade-off between the two terms in the distortion in (4). For a fixed SNR, as the stretch factor increases (by increasing α), the magnitude of each of the frequent small errors declines. However, as the minimum distance decreases, the frequency of threshold errors grows. Specifically, the first term in (4), scales as:

$$E\left[(\hat{S}-S)^2 | \mathbf{Z} \notin \mathcal{T}\right] \propto \frac{1}{L^2} \quad (12)$$

Therefore, the small or local errors decrease with increasing stretch factor and α. The second term in the distortion of (4), which is due to threshold errors, grows with increasing α because at a fixed SNR and more closely spaced coding line segments, there is a higher probability that the noise will map a codeword to a distant codeword.

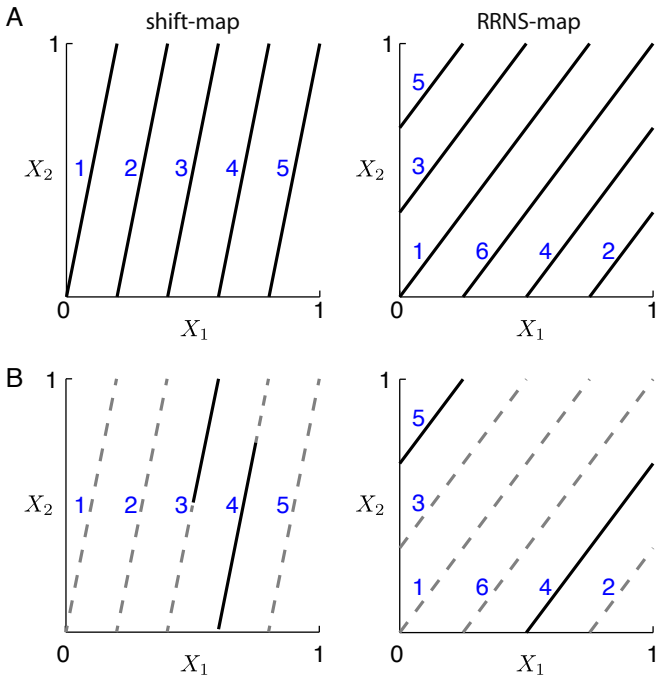

Figure 4: Examples of the shift map with $(a_1, a_2) = (1, 5)$ on the left and the *RRNS-map* with $(a_1, a_2) = (3, 4)$ on the right. Both codes are chosen to have similar stretch factors and minimum distances. Numbers next to individual segments in blue indicate the order of the mapping as the source increases from 0 to 1. A and B show codebooks without and with side information $W = [0.5, 0.75]$, respectively. Gray dashed lines correspond the codeword inconsistent with W.

3.2. A generalization of the shift-map code for the side information problem

The shift-map code is generalized by relaxing the conditions on the scaling coefficients in (5). In conventional shift-map codes, the source is multiplied by a geometric series $\{a^{n-1}\}$ followed by the modulo operation. In contrast, we propose it is possible to define generalized shift-map codes using arbitrary (non)integer modulii $\mathbf{a} = (a_1, a_2, \ldots, a_N)$for code generation. When the modulii are integer and co-prime, we will call that set of shift-map codes the RRNS-map codes.

Definition 2 (The RRNS-map code)**.** *For a source $S \in [0, 1)$ and a set of relatively prime integers $\mathbf{a} = (a_1, a_2, \ldots, a_N)$, the RRNS-map codeword is given by $\mathbf{X}^{RRNS}(S) = [X_1^{RRNS}(S), X_2^{RRNS}(S), \ldots, X_N^{RRNS}(S)]$, where, for $n = 1, 2, \ldots, N$,*

$$X_n^{RRNS}(S) = a_n S \mod 1, \tag{13}$$

$$gcd(a_n, a_m) = 1 \quad \text{for } n \neq m. \tag{14}$$

Here, $gcd(\cdot, \cdot)$ represents the greatest common divisor.

We assume $1 < a_1 < \cdots < a_N$ for convenience and refer to these as the *scale factors*. Similar to the shift-map definition given above, the set of all the codewords generated by the RRNS code is denoted as

$\mathcal{X}^{RRNS}$ and the partial codebook for $S \in W \subset [0,1)$ $\mathcal{X}^{RRNS}(W)$:

$$\mathcal{X}^{RRNS} = \left\{ \mathbf{X}^{RRNS}(S) | S \in [0,1) \right\} \tag{15}$$

$$\mathcal{X}^{RRNS}(W) = \left\{ \mathbf{X}^{RRNS}(S) | S \in W \right\}. \tag{16}$$

All the shift-map codes, including the RRNS-map codes $\mathcal{X}^{RRNS}$, form parallel line segments with direction $(a_1, a_2, \ldots, a_N)$ inside the unit hypercube (Figure 4B). The stretch factor $L^{RRNS}(\mathbf{a})$ and the minimum distance $d^{RRNS}(\mathbf{a})$ are defined in the same way as for the shift-map code.

In the conventional shift-map codes, because the n'th coordinate of the codeword is related to the source S by the scale factor $\alpha^{(n-1)}$ with $\alpha > 1$, the coordinate with larger n encodes local changes of the source with a geometrically greater sensitivity. In contrast, in RRNS-map codes, one can choose a_n's to lie within a small range so that all the X_n's have similar sensitivity to variations in the source value.

We illustrate potential benefits of the RRNS-map code with the following example. Consider Figure 4A, which compares examples of a conventional shift-map (left) and RRNS-map (right) code. The parameters for these codes are $(a_1, a_2) = (1, 5)$ and $(a_1, a_2) = (3, 4)$, respectively. 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. The numbers in blue indicate the order of the encoding line segments, as the source point ranges from 0 to 1. In the conventional shift-map code, each component of codeword monotonically increases, and the resulting line segments are ordered sequentially. However, in the RRNS-map code, the order is 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 to reduce distortion.

In Figure 4B, the effects of side information on the shift-map (left) and RRNS-map (right) codes are demonstrated. Solid lines indicate codewords consistent with side information $W = [0.5, 0.75]$, corresponding to segments 3 and 4 for the shift-map code and segments 4 and 5 for the RRNS-map code. Gray dashed lines indicate codewords inconsistent with W. In the shift-map codes, the distance between candidate segments remains the same because they lie next to each other (Figure 4B left). In contrast, the distance between candidate segments of the RRNS-map code is much larger (Figure 4B right), resulting in a larger minimum distance and, consequently, a lower threshold error probability.

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 and minimum distance as the corresponding shift-map code, which guarantees the same distortion in the absence of side information. The second is to make the RRNS-map codebook interleave itself so that neighboring segments encode distant subintervals of the source. 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 the overall result is a smaller distortion without changing the encoding scheme.

4. RRNS-map codes

In this section, we study the structure of the RRNS-map codebook from a geometric perspective.

4.1. Geometric interpretation: "Cylinder packing" problem

We consider a geometrical perspective and discuss finding a good RRNS-map with maximum separation. A cylinder around the RRNS-map codebook with radius r is defined as follows:

$$\mathcal{C}(r) = \left\{\mathbf{x} \in [0,1)^N : |\mathbf{x} - \mathbf{x}^{RRNS}| < r,\ \forall \mathbf{x}^{RRNS} \in \mathcal{X}^{RRNS}\right\}, \tag{17}$$

where $|\cdot|$ is the Euclidean distance. The problem here is to find the direction $\mathbf{a} = (a_1, a_2, \ldots, a_N)$ of the cylinder with the maximum radius r under the constraint that $C(r)$ does not intersect itself. We call a problem of *cylinder packing*. Since $\mathbf{a}$ consists of relatively prime integers, cylinder packing is an integer programming problem with an unusual search domain, denoted as the following:

$$\begin{aligned} \mathcal{A} \quad = \quad & \{(a_1, a_2, \ldots, a_N) \mid \gcd(a_n, a_m) = 1 \text{ if } n \neq m, \\ & a_n < a_m \text{ if } n < m\}. \end{aligned} \tag{18}$$

The requirement of relatively prime coordinates might appear to be too stringent and the size of the search space $\mathcal{A}$ itself might appear small. However, as the dimension N grows, the probability of finding points with relatively prime coordinates among the N-dimensional rectangular lattice quickly approaches 1 [23], meaning that there exist many RRNS-map codes. To simplify the problem, we restrict the search domain to those vectors in $\mathcal{A}$ with coordinates that are not greater than a predetermined a_{max}, denoted as

$$\begin{aligned} \mathcal{A}_{a_{max}} \quad = \quad & \{(a_1, a_2, \ldots, a_N) \mid \gcd(a_n, a_m) = 1 \text{ if } n \neq m, \\ & a_n < a_m \text{ if } n < m,\ a_n \leq a_{max}\} \end{aligned} \tag{19}$$

This restriction avoids stretch factors that are too large and produce severe threshold errors with result in large distortions.

In order to find the tightest cylinder packing, let us consider the hyperplane that is orthogonal to the cylinder axis. The hyperplane that is orthogonal to $\mathbf{a}$ and passes through the center of the unit hypercube is denoted by $H(\mathbf{a})$. Algebraically, this hyperplane is

$$H(\mathbf{a}) = \left\{\mathbf{x} \in [0,1)^N \mid \mathbf{a} \cdot (\mathbf{x} - \mathbf{c}) = 0\right\}, \tag{20}$$

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 $\mathcal{X}^{RRNS}$ intersects with $H(\mathbf{a})$ at a point. We call the set of such intersections $\mathcal{X}^*$ the *discrete codebook*, denoted as:

$$\mathcal{X}^* = H(\mathbf{a}) \cap \mathcal{X}^{RRNS} \tag{21}$$

$$= \{\mathbf{a}S \bmod 1 \mid S \in [0,1), \mathbf{a} \cdot (\mathbf{a}S \mod 1) = \mathbf{a} \cdot \mathbf{c}\}. \tag{22}$$

Similarly, the discrete codebook of corresponding source in the subinterval W is defined as follows:

$$\mathcal{X}^*(W) = H(\mathbf{a}) \cap \mathcal{X}^{RRNS}(W) \tag{23}$$

$$= \{\mathbf{a}S \bmod 1 \mid S \in W, \mathbf{a} \cdot (\mathbf{a}S \mod 1) = \mathbf{a} \cdot \mathbf{c}\}. \tag{24}$$

Maximizing the radius of the cylinder is equivalent to maximizing the distance between points in the discrete codebook. The minimum distance of the RRNS-map code is equal to the minimum distance between distinct points in $\mathcal{X}^*$:

$$d^{RRNS}(\mathbf{a}) = \min_{x \neq x' \in \mathcal{X}^*} |\mathbf{x} - \mathbf{x}'| \tag{25}$$

Unfortunately, the minimum distance depends on $\mathbf{a}$ in a non-trivial way and the search domain $\mathcal{A}$ is not convex. Appendix A describes a way to identify all the points in $\mathcal{X}^*$ for a given $\mathbf{a}$ and to calculate the minimum distance $d^{RRNS}(\mathbf{a})$.

4.2. The trade-off between the minimum distance and the stretch factor

The generalized shift-map codes, including RRNS-map codes, also exhibit a fundamental tradeoff between minimum distance and stretch factor. $d^{RRNS}(\mathbf{a})$ and $L^{RRNS}(\mathbf{a})$ are related to the radius and the length along the axis of the cylinder, respectively, and the volume of the cylinder is upper-bounded by that of the unit hyper cube. Here, assuming evenly spaced coding lines, we have that the scaling of the optimal $d^{RRNS}(\mathbf{a})$ with $L^{RRNS}(\mathbf{a})$ and N will obey the following relationship:

$$d^{RRNS}(\mathbf{a}) = O(L^{RRNS}(\mathbf{a})^{-\frac{1}{N-1}}). \tag{26}$$

Figure 5 shows a concrete example of this trade-off. For each $\mathbf{a}$ in $\mathcal{A}_{50}$ with $N = 5$ and $W = [0, 1]$ (no side information), the minimum distance of the RRNS-map code $d^{RRNS}(\mathbf{a})$ is numerically computed and plotted as a function of the stretch factor $L^{RRNS}(\mathbf{a})$ in log-scale (black and gray dots). The dashed line provides a reference scaling, $\log d = -\frac{1}{N-1} \log L$, corresponding to the prediction in (26). Black dots highlight RRNS-map codes with large minimum distances, greater than 95% of the prediction in (26) while gray dots indicate RRNS-map codes with smaller minimum distances. The two red diamonds are for the shift-map codes with design parameter $\alpha = 2$ (left) and $\alpha = 3$ (right). There are 1156 such RRNS-map codes with large minimum distances among 15279 $\mathbf{a}$'s in $\mathcal{A}_{50}$. Those RRNS codes fill in the gaps between the shift-map codes with different α's. In this sense, the RRNS-map codes generalize shift-map codes.

Overall, good RRNS-map codes can be found (without considering side information), and can be used as candidates for good RRNS-map codes with side information. The RRNS-map codes found here are used in Section 5 to search for codes that satisfy an additional requirement: that the minimum distance should scale well as side information is revealed to the decoder.

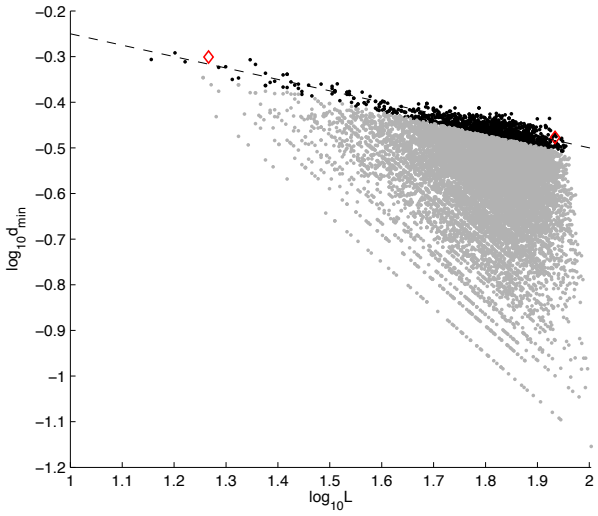

Figure 5: The trade-off between stretch factor and minimum distance for RRNS-map codes (black and gray dots), compared with conventional shift-map codes (red diamonds) for $\mathcal{A}_{50}$. The dashed line represents predicted scaling with slope $-\frac{1}{N-1}$ between stretch factor and minimum distance in (26). Black dots highlight RRNS-map codes with large minimum distances that are close to the optimal predicted scaling.

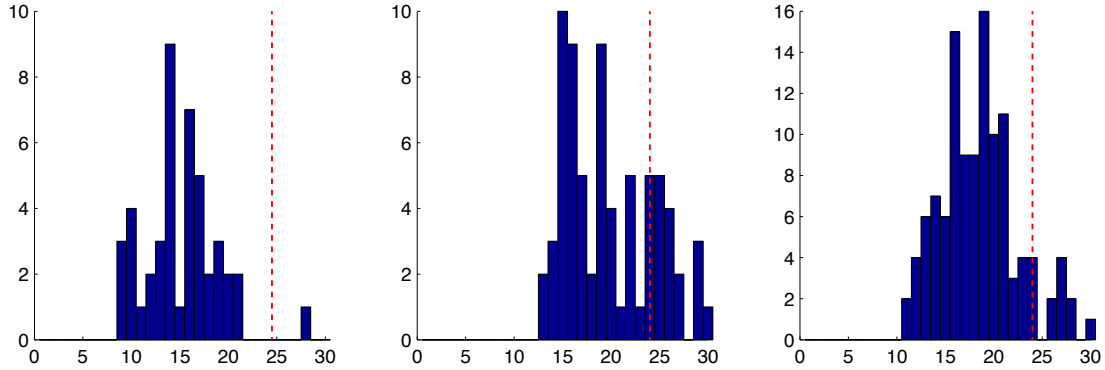

Figure 6: Histograms of the number of neighbors in RRNS-map codes with $\mathbf{a}_1 = (5, 7, 9, 11, 17)$ (left), $\mathbf{a}_2 = (5, 7, 13, 23, 27)$ (middle), and $\mathbf{a}_3 = (5, 11, 19, 37, 47)$ (right) in the projected codebook $\mathcal{X}^*$, which resides in an $(N-1) = 4$-dimensional space. Red dashed lines show the densest possible number of neighbors in a regular lattice in the same space.

4.3. Cylinder packing versus sphere packing

To further understand the cylinder packing problem, we address how densely $\mathcal{X}^*$ can be packed. As a reference, we compare the local structure of the RRNS-map code with that of lattice codes. Finding the lattice with the greatest density in a given dimension is equivalent to finding a way to pack as many as non-overlapping spheres with the same radii into a volume. The maximum number of spheres that can be packed around a central sphere without overlap (just touching at the boundaries) is called the kissing number. Finding the largest possible kissing number in a given dimension is *the kissing number problem* [24].

The centers of the spheres comprise a lattice codebook to encode a discrete source.

Cylinder packing and sphere packing are compared in the $N-1$ dimensional orthogonal hyperplane H. Note that the former controls the direction of the cylinders in N dimensional torus such that its projections are tightly packed points in H, while the latter concerns the tightest sphere packing directly in H. Thus, sphere packing in the $(N-1)$-dimensional perpendicular hyperplane puts an upper bound on the densest cylinder packing in the N-dimensional hypercube.

We report that good RRNS-map codes have well-spaced codewords that are close to lattices. Among the good RRNS codes with $N=5$ found in the previous subsection (black dots in Figure 5), three sets of parameters that satisfy constraints in the next section are chosen: $\mathbf{a}_1=(5,7,9,11,17)$, $\mathbf{a}_2=(5,7,13,23,27)$, and $\mathbf{a}_3=(5,11,19,37,47)$. For each parameter, the Voronoi regions of $\mathcal{X}^*$ and the corresponding numbers of neighbors are calculated by the quickhull algorithm [25] ported into Matlab [26]. The Voronoi region of a point $\mathbf{x}_c^*\in\mathcal{X}^*$ is defined as follows:

$$\begin{aligned} V_{\mathbf{a}}(\mathbf{x}_c^*) &= \{\mathbf{x}\in\mathcal{X}^* \,|\, |\mathbf{x}-\mathbf{x}_c^*|<|\mathbf{x}-\mathbf{x}'|, \\ &\qquad \mathbf{x}'\in\mathcal{X}^*, \mathbf{x}'\neq\mathbf{x}_c^*\}, \end{aligned} \tag{27}$$

where $|\cdot|$ is the Euclidean distance. The histograms of number of neighbors for these RRNS-map codes are shown in Figure 6. In the four dimensional space, the highest attainable kissing number is 24 [24], shown in red dashed lines in Figure 6. These RRNS-map codes have numbers of neighbors (15.2 ± 3.8, and 19.8 ± 4.6, and 18.5 ± 4); while not directly measuring the same quantity, it is nevertheless notable that these numbers are of similar size to the kissing number in four dimensions. This hints that good RRNS-map codes may indeed be well-spaced in the $(N-1)$ dimensional orthogonal hyperplane and close to lattice codes.

5. Adaptive decoding of RRNS-map codes with side information

In this section, we design the RRNS-map codebook so that side information at the decoder is efficiently combined, resulting in a lower distortion.

5.1. The RRNS-map codebook shrinks with side information at the decoder.

With side information W, the decoder needs only to use a subset of the codebook, corresponding to $\mathcal{X}^{RRNS}(W)$. The size of this partial codebook decreases as more side information is revealed to the decoder (equivalently, as $|W|$ decreases). The property we require for good RRNS-map codes is that this partial codebook is maximally separated in the codeword space and the minimum distance of $\mathcal{X}^{RRNS}(W)$ increases as $|W|$ decreases.

For example, Figure 7 illustrates the RRNS-map code with $\mathbf{a}=(3,5,7)$. In Figure 7A, we assume no additional information about S. Thus, the decoder must search the entire codebook $\mathcal{X}^{RRNS}$ in the left panel. The projected codebook (onto the perpendicular hyperplane H passing through the center of the hypercube), $\mathcal{X}^*([0,1])$, is shown in the right panel. The green area represents the Voronoi region of the center point.

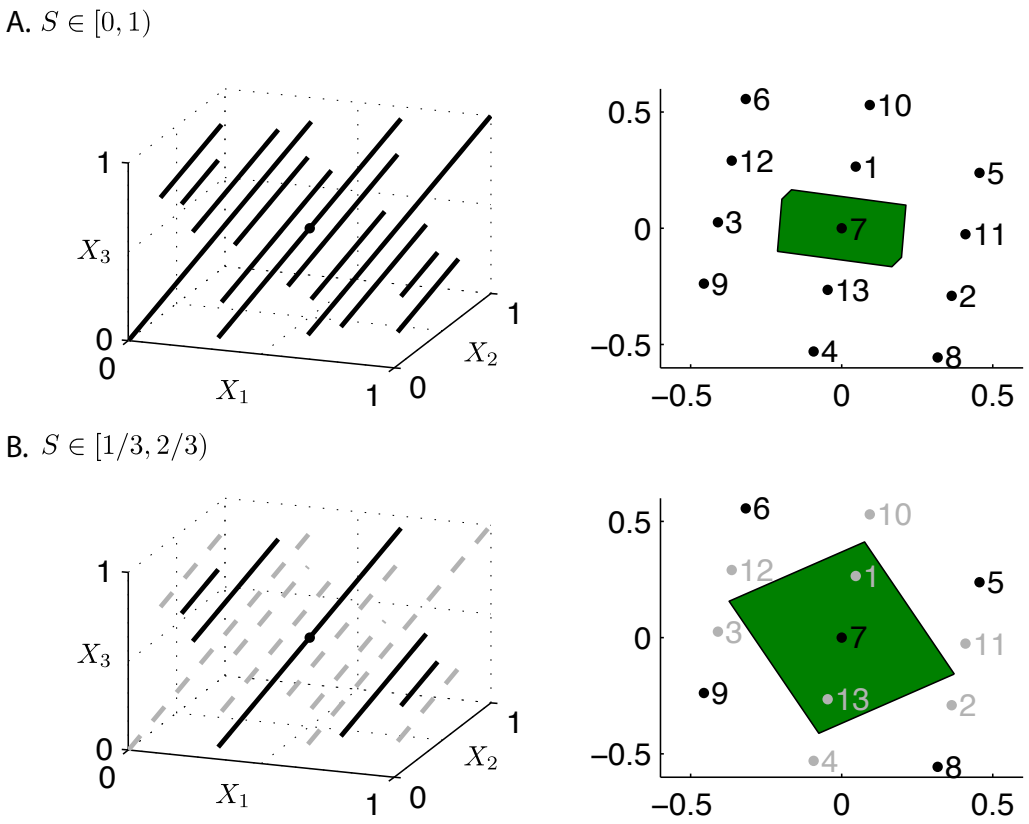

Figure 7: The RRNS-map codebook $\mathcal{X}^{RRNS}$ (left) and its projection to the orthogonal plane $\mathcal{X}^*$ (right) either without (A) and with (B) side information at decoder. Numbers in the right denote the order as the source increases from 0 and the green region is the Voronoi region of the center point.

In Figure 7B, side information $S \in W = [1/3, 2/3]$ is provided to the decoder. Thus, the decoder will only consider a subset of the codebook, $\mathcal{X}^{RRNS}([1/3, 2/3])$, which contains five segments shown in black in the left panel. The number of segments needed for decoding decreases from 12 to 5. The right panel shows the projection of those five active segments in black; gray points are not considered. Because the segments are well-interleaved, the side information results in an increased distance between neighbors. Consequently, the Voronoi region (in green) of the center point increases.

Next, we quantify this shrinkage of the code book as a function $|W|$. The cardinality of the discrete codebook, $|\mathcal{X}^*(W)|$, linearly scales with $|W|$, as summarized in the following theorem.

Theorem 1. *Given an RRNS-map code with parameter* **a** *satisfying (14), the cardinality of* $\mathcal{X}^*(W)$ *— the intersection set between the active codebook* $\mathcal{X}^{RRNS}$ *with* W *of length* $|W|$ *and the orthogonal hyperplane* H *in (20) — is*

$$1 + \sum_{i=1}^{N} (\lfloor a_n |W| \rfloor - 1) \leq |\mathcal{X}^*(|W|)| \leq 1 + \sum_{i=1}^{N} (\lfloor a_n |W| \rfloor) \tag{28}$$

Proof. The theorem follows from the same argument as in Lemma 2 in Appendix A, with the slight modifi-

cation that the support of the source considered is reduced by a factor of $|W|$. $\mathcal{X}^{RRNS}(|W|)$ intersects with $H_n(\mathbf{a})$ when the corresponding source is $S = \frac{i}{a_n} \in W$ for some integer i. Thus, there are $\lfloor a_n|W| \rfloor$ integers for i. If $i = 0$ is included, it should be excluded to avoid multiple counting. Summing up for all a_n's, and subtracting redundant counts for $S = 0$, we have (28). □

For a large N, one can ignore the first term in the right side of (28) and have the following corollary.

Corollary 2. *The number of active points scales with the length of the side information $|W|$.*

$$|\mathcal{X}^*(W)| \approx |\mathcal{X}^*||W|. \tag{29}$$

5.2. The minimum distance of the RRNS-map codebook increases with side information at the decoder.

As the number of coding points in H decreases, the minimum distance between them will tend to increase. An upper bound of the minimum distance $\tilde{d}$ is derived as a function of $|W|$. The radius of cylinder packing is bounded from above by that of optimal sphere packing. Thus, the volume of a $(N-1)$-dimensional ball with radius $\tilde{d}/2$ multiplied by the number of points in $\mathcal{X}^*(W)$ is bounded from above by the volume of H, which is a constant:

$$|\mathcal{X}^*(W)| \frac{\pi^{\frac{N-1}{2}}}{(\frac{N-1}{2})!} \left(\frac{\tilde{d}}{2}\right)^{(N-1)} \leq \mathrm{Vol}(H), \tag{30}$$

where the formula for $(N-1)$-dimensional ball is used when N is an odd and $\mathrm{Vol}(H)$ represents the volume of H. Using the approximation in (29) for the number of balls and applying Stirling's formula $(\frac{N-1}{2})! \approx (\frac{N-1}{2})^{(\frac{N-1}{2})} e^{-(\frac{N-1}{2})}$, we have

$$\tilde{d} \leq 2\sqrt{\frac{N-1}{2\pi e}} \left(\frac{\mathrm{Vol}(H)}{|\mathcal{X}^*||W|}\right)^{1/(N-1)}. \tag{31}$$

Thus, for a fixed $\mathbf{a}$, the exponent of $\tilde{d}$ scales with $|W|$ as follows:

$$\log \tilde{d} = O\left(\frac{1}{N-1} \log \frac{1}{|W|}\right). \tag{32}$$

5.3. Examples of good RRNS-map codes for $N = 5$

To find good RRNS-map codes among the candidates found in Section 4, we add the constraint that the partial codebook $\mathcal{X}^*(W)$ must have large minimum distances for *a range of* $|W|$. In particular, the minimum distance for $|W| < 1$ should be larger than the minimum distance for $|W| = 1$ (no side-information), and should scale similarly as the minimum distance upper-bound of (32). The number of active points should scale linearly with $|W|$ as predicted in (29).

L^{RRNS} and d_{min}^{RRNS} and are calculated for each $\mathbf{a} \in \mathcal{A}_{50}$ with $N = 5, a_{max} = 50$, and varying $W < 1$. For each $\mathbf{a} \in \mathcal{A}_{50}$, W is varied such that its boundaries are aligned to points among $\{0.1, 0.2, \ldots 1\}$. The number of points in $\mathcal{X}^*(W)$ and the minimum distance are averaged over intervals of the same length and, therefore, are functions of $|W|$.

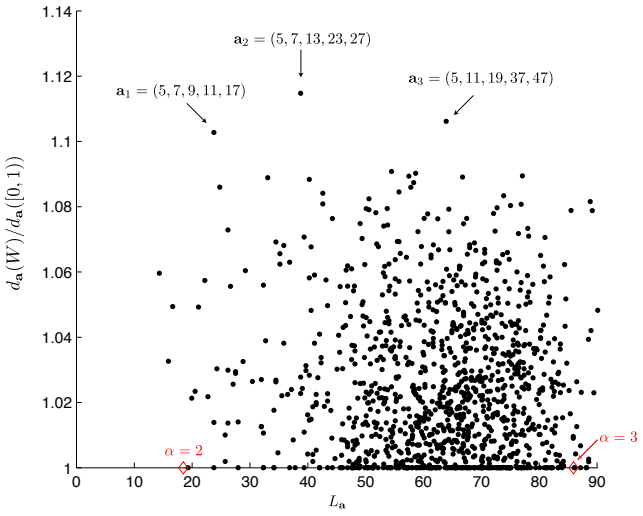

Figure 8: The minimum distance of the RRNS-map code increases with additional side information. For good RRNS-map codes identified in Figure 5, each black dot represents the ratio of the minimum distance with side information W to the minimum distance of the whole codebook ($d_{\mathbf{a}}(W)/d_{\mathbf{a}}([0,1))$), averaged over a uniformly sampled range of $|W|$ ($|W|$ ranges from 0.1 to 1 in steps of 0.1). Three parameters for the RRNS-map with three largest average increase in minimum distance with side information are $\mathbf{a}_1 = (5, 7, 9, 11, 17), \mathbf{a}_2 = (5, 7, 13, 23, 27)$, and $\mathbf{a}_3 = (5, 11, 19, 37, 47)$. For comparison, shift-map codes with $\alpha = 2, 3$ are shown in red diamonds.

The choices $\mathbf{a}_1 = (5, 7, 9, 11, 17), \mathbf{a}_2 = (5, 7, 13, 23, 27)$, and $\mathbf{a}_3 = (5, 11, 19, 37, 47)$ produce the three largest minimum distances (Figure 8). The stretch factors of those codes are between those of shift-map codes with $\alpha = 2$ and 3 (shown in red diamonds). The left panel of Figure 9 shows the size of the partial codebook $\mathcal{X}^*(W)$ as a function of $|W|$. Circles in the figure are numerical calculations and dashed lines are analytical predictions from (29). Consistent with the prediction, the partial codebook size $|\mathcal{X}^*(W)|$ linearly scales with $|W|$. The right panel of Figure 9 shows the minimum distances as a function of $|W|$. Circles in the figure are numerical results and dashed lines are least-mean-square fits of the numerical results. The slopes of the fits are -0.22, -0.23, and -0.22, which are close to the analytical prediction $-\frac{1}{N-1} = -0.25$ from (32). Thus, the size of the partial codebook and resulting minimum distance for these RRNS-map codes match the analytical predictions, and thus have the desired properties.

5.4. Probability of threshold error and distortion of RRNS-map codes decrease with side information at the decoder.

The threshold error probability of good RRNS-map codes decreases as more side information is revealed to the decoder, because of the increase in the minimum distance. For the RRNS-map codes with parameters $\mathbf{a}_1 = (5, 7, 9, 11, 17)$, $\mathbf{a}_2 = (5, 7, 13, 23, 27)$, and $\mathbf{a}_3 = (5, 11, 19, 37, 47)$, upper bounds on the probability of threshold error (P_{th}) are calculated using the minimum distances and numbers of neighbors (See Appendix B). In Figure 10 (left), the probabilities of threshold errors for the RRNS-map codes (triangles) and the

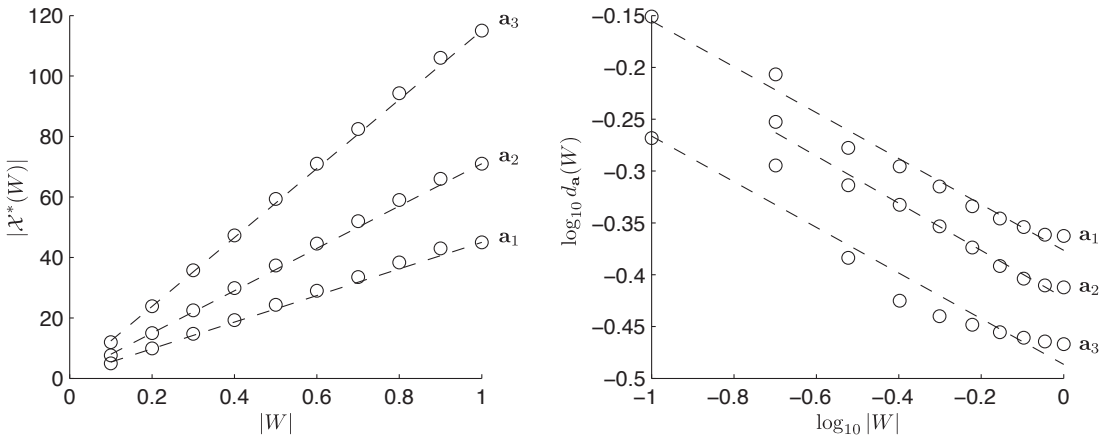

Figure 9: How the number of active points and minimal distances scale with side information in good RRNS-map codes. Numbers of active points (left) and minimum distances (right) of good RRNS-map codes with $\mathbf{a}_1 = (5, 7, 9, 11, 17)$, $\mathbf{a}_2 = (5, 7, 13, 23, 27)$, and $\mathbf{a}_3 = (5, 11, 19, 37, 47)$ are plotted as a function of $|W|$. Circles indicate numerically found values, which agree well with analytical predictions (29) and (32), shown as dashed lines.

shift-map codes (red dashed lines) are compared when $\sigma = 0.05$. As $|W|$ decreases (more side information), the upper bounds on P_{th} of the RRNS-map codes decrease, while lower bounds on P_{th} for conventional shift-map codes remain constant.

Consequently, the distortion of the RRNS-map decreases with increasing side information at the decoder. In Figure 10 (right), distortion in the good RRNS-map codes are compared to shift-map, tent-map, and repetition codes. Upper bounds on the distortion of the RRNS-map codes (triangles) decreases as $|W|$ decreases. However, lower bounds on the distortions of the shift-map (red dashed lines) stay constant, as does the average distortion of the repetition code (green): $\frac{\sigma^2}{N}$. Those lower bounds are derived similarly to [15] (Appendix C). The red cross represents the distortion of the tent-map code [13, Eq. (16a) and (16b)]. Without side information ($|W| = 1$), the upper bounds of the distortion of the RRNS-map codes with $\mathbf{a}_1 = (5, 7, 9, 11, 17)$, $\mathbf{a}_2 = (5, 7, 13, 23, 27)$ (from bottom to up) are similar to the lower bounds of the shift-map codes with $\alpha = 2$ and 3, respectively. As $|W|$ decreases (and thus side information increases), distortion in the RRNS-map codes decreases while that in the conventional shift-map codes stays constant. Interestingly, the RRNS-map code with $\mathbf{a}_3 = (5, 11, 19, 37, 47)$ has larger distortion than the other two RRNS-map codes shown in this plot when $|W| = 1$, but the distortion crosses over and becomes smaller than in the other RRNS-map codes when $|W|$ approaches 0.

The differing reductions in distortion with side information for the different RRNS-map codes suggests that choosing code parameters based on prior knowledge about the distribution of $|W|$ could result in lower average distortion. For instance, if $|W|$ were uniformly distributed $\in [0, 1]$, then $\mathbf{a}_1$ and $\mathbf{a}_2$ would produce lower average distortions than would $\mathbf{a}_3$. On the other hand, if $|W|$ were concentrated around a smaller value and is large only occasionally, the RRNS-map code with $\mathbf{a}_3$ would achieve a lower average distortion. Thus,

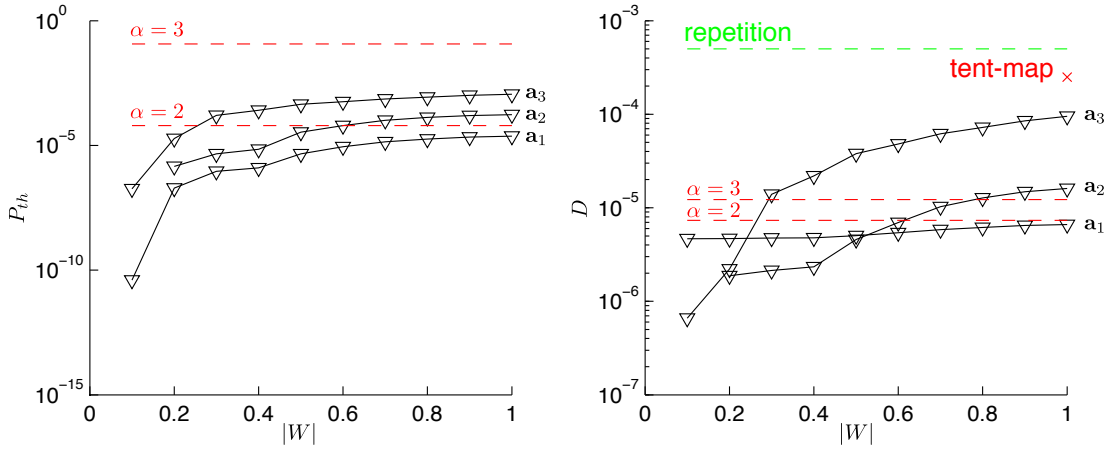

Figure 10: The probability of threshold error and distortion in the RRNS-map code. Left: The probability of threshold error for the RRNS-map code decreases with increasing side information at the decoder (decreasing $|W|$). Right: The distortion in some good RRNS-map codes compared to that in other analog codes. Distortion in RRNS-map codes without side information ($|W| = 1$) is similar to that in shift-map codes with similar stretch factors ($\alpha = 2, 3$). As more side information is revealed to the decoder, distortion in the RRNS-map code decreases without changes in the encoding scheme. This is in contrast to other analog codes, in which distortion is independent of side information when the encoder is ignorant of the side information.

specific RRNS-map parameters may be preferable for a specific distribution on the amount of available side information, an area for future research.

6. Extensions

In the previous section, we focused on the advantage the RRNS-map coding presents, in increasing the robustness of the source estimate to noise when side information is available at the decoder. In the next subsection, we study the cost of this side information. In the second subsection, we apply the RRNS-map code for a slightly different scenario, where we show that the RRNS-map code outperforms the shift-map code when only parts of the codeword are available at the decoder. (Showing that the RRNS-map code is robust to adversarial erasures in the channel.)

6.1. The case when side information is not free

We consider the cost of transmitting any side information from the transmitter to the receiver. For simplicity, we assume that the interval is divided into 2^Q bins of width $|W| = 2^{-Q}$ each (where Q is any integer), and the side information involves conveying the number Q, which specifies how many bins there are (this number also automatically specifies the bin boundaries within the interval $[0, 1]$), and the index k of the specific bin in which the source is contained. Specifying Q requires $\log_2(Q)$ bits, and specifying k involves $\log_2(k)$ bits, where k ranges from 1 to 2^Q. Thus, $\log_2(k) \leq Q$, and the total number of bits in the side information is bounded above by $\log_2(Q) + Q$. Substituting $Q = \log_2(1/|W|)$, we have that the side information channel uses an additional B bits with $B = \log_2(1/|W|) + \log_2(\log_2(1/|W|))$.

Next, the number of additional channel uses, ΔN, to transmit B bits is calculated by considering the channel capacity of the AWGN channel with SNR γ:

$$B < \frac{\Delta N}{2} \log_2 (1 + \gamma) . \tag{33}$$

Therefore, we obtain

$$\Delta N > \frac{2[\log_2(1/|W|) + \log_2(\log_2(1/|W|))]}{\log_2 (1 + \gamma)}. \tag{34}$$

Thus for a performance comparison, the RRNS-map codes utilized over N channel uses with side information W might more fairly be compared against conventional shift-map codes that are allowed $N + \Delta N$ channel uses (and no side information).

Any performance gains in the conventional shift-map codes with the added ΔN channel uses depend on how they are allocated. One possibility is to keep with the conventional shift-map coding scheme, and thus to hold the design parameter α fixed while adding registers with higher powers of α, so that $\vec{a} = (1, \alpha, \alpha^2, \ldots, \alpha^N, \ldots, \alpha^{N+\Delta N-1})$ $(1 < \alpha \in \mathcal{Z}^+)$. In this scheme, because the additional registers represent finer-scale information about the source S, the gains from ΔN will only occur at very high SNR. At lower SNR, noise will swamp any information transmitted in the channels with larger scale factors and eliminate any gains of a non-zero ΔN. This is in fact a general limitation of conventional shift-map codes, because for any given SNR, there is a cutoff in n beyond which registers α^n convey no information about the source.

On the other hand, the additional ΔN may be used to build a hybrid shift-map and repetition code. Instead of a single geometric series with powers of α from 0 to $N + \Delta N - 1$, registers with lower exponents of α could be repeated. Thus, the code parameters would be $(1, \alpha, \alpha^2, \ldots, \alpha^{(N-1)}, 1, \alpha, \alpha^2, \ldots, \alpha^{(\Delta N-1)})$, $(1 < \alpha \in \mathcal{Z}^+)$. When $N = \Delta N$, such a scheme results in a doubling of the SNR and a 3dB decrease in distortion.

Figure 11 quantifies the cost of side information and the potential decrease in the distortion of conventional shift-map codes if we take into account the extra side information cost of the RRNS-map codes. According to Equation 34, ΔN increases to 2 as $|W|$ decreases down to 0.1 with $\sigma = 0.05$ (Figure 11 left). By the hybrid shift-map repetition code scheme, with ΔN transmissions used to repeat the smallest scale-factor registers, the effective SNR increases and distortion decreases, relative to the case when the sender was allowed only N transmissions (no repetitions), Figure 11 right. Note that compared to the corresponding orders-of-magnitude decrease in distortion in the RRNS-map code when side information is provided, Figure 10, the decrease in distortion that results by granting extra channel uses to the conventional shift-map code is limited and modest.

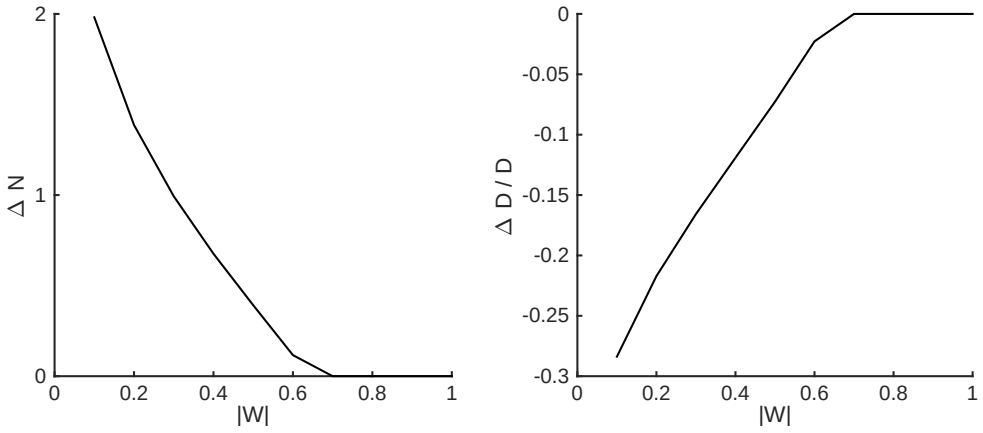

Figure 11: **The cost of side information.** Left: The cost of side information is quantified in terms channel uses. With the same parameters as in Figure 10 ($N = 5, \sigma = 0.05$), ΔN increases up to 2 as $|W|$ decreases. Right: The potential improvement in distortion if the additional channel uses were granted to the conventional shift-map code. If the additional ΔN channel uses corresponding to different values of $|W|$ are allocated to repeating some entries of the conventional shift-map code, distortion at the decoder decreases by the factor of $\frac{N}{N+\Delta N}$. Consequently, the distortion decreases to 70% (right).

6.2. Robustness of RRNS-map codes to erasure

In communication networks and biological neural networks, some connections may completely fail and only a partial measurements may be available at the decoder. Here, we show that the RRNS-map code can be robust to erasure of codeword symbols.

Let us assume that only N_o out of N measurements are available to the decoder. In the conventional shift-map codes, if the registers corresponding to the smallest scale-factors are erased, the distortion will be large; loss of the largest scale-factors will lead to only small increases in distortion. By contrast, in RRNS-map codes, because the scale factors are relatively prime and all roughly of the same size, the increase distortion will not be large for any erasure.

This effect is shown in Figure 12. The stretch factors of the RRNS-map and shift-map codes shown there are very close (9.1 and 9.5, respectively), hence they exhibit similar mean square errors (MSEs) in the absence of any erasures. When the first component is erased from both codewords, the mean square error (MSE) of the shift-map code is significantly larger than that of the RRNS-map code (Figure 12C, second pair of bars). Erasures of either of the other two components, corresponding in the shift-map code to larger scale-factors, results in similar but slightly smaller distortion in the shift-map than the RRNS-map code. The net result is that the RRNS-map code outperforms the shift-map code when all symbol erasure patterns are equally likely, at the SNR considered here.

7. Conclusions

In summary, we have generalized the shift-map code family of analog codes to a larger family that does not involve a geometric series of a single design parameter. Instead, the codewords are generated by modulo

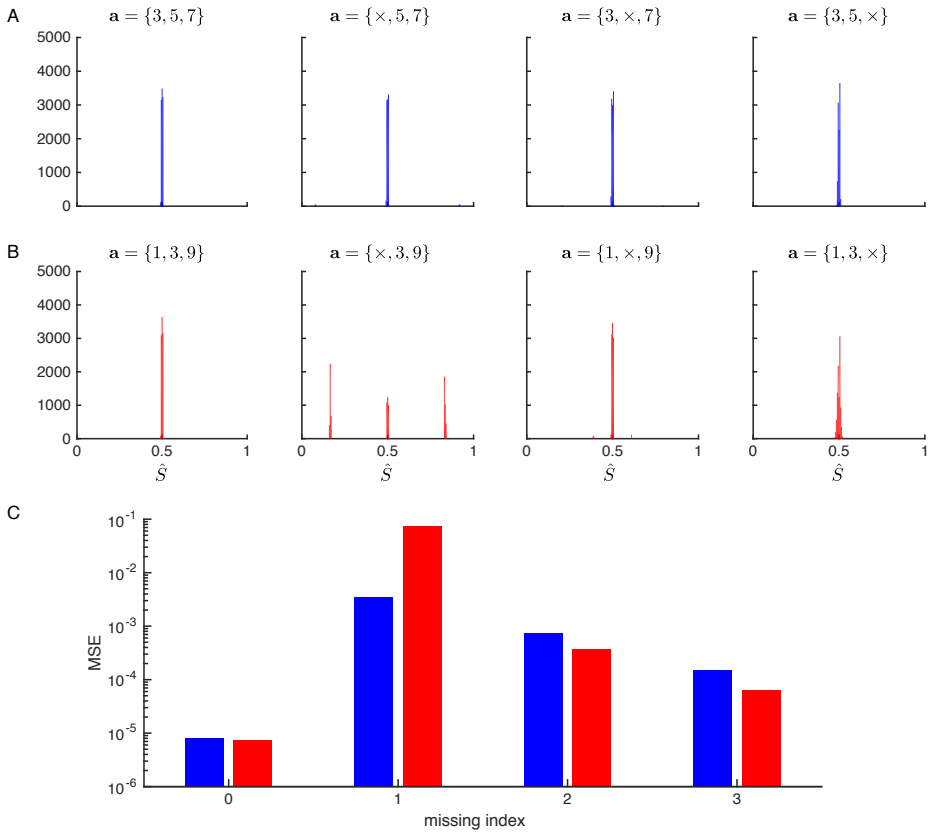

Figure 12: **Effects of erasure on RRNS-map and shift-map codes**. (A-B) Histograms of the decoded estimate of the source $\hat{S}$ without (leftmost column) or with (right three columns) single erasures in the RRNS-map and shift-map codes. The single erased register is designated by '×'. (A) The RRNS-map code with $\mathbf{a} = \{3, 5, 7\}$, and (B) the shift-map code with $\alpha = 3$ (B). 10^4 samples are generated to quantify decoding performance. The stretch factors of these codes are similar (9.1 and 9.5, respectively) and thus they exhibit similar mean square errors (MSEs) without puncturing (C, leftmost).

remainder after multiplying the source with a set of (non)integer scale-factors $\mathbf{a} = (a_1, a_2, \ldots, a_N)$. We identify a subset of these generalized shift-map codes, defined as involving co-prime integer scale-factors, and call these codes RRNS-map codes.

Good RRNS-map codes can, thanks to their interleaving structure, allow the decoder to use side information to produce an estimate of the source that is less prone to threshold errors, and thus has lower distortion. The interleaving structure is generated by carefully chosen relatively prime integer parameters for the code. Without side information, good RRNS-map codes achieve a comparable performance as corresponding conventional shift-map codes. With side information, RRNS-map codes can outperform the shift-map code.

Designing an RRNS-map code was formulated as an integer program over relatively prime integers, which has a geometrical interpretation of cylinder packing. We numerically solved this problem to find example RRNS-map codes that have well-spaced partial codebooks. These codes result in excellent performance when varying degrees of side information are made available to the receiver, and are moreover somewhat tolerant

to erasure of components of the codeword.

It will be interesting as part of future work to consider the complexity of decoding of RRNS-map codes, and to consider whether these codes can be decoded with similar complexity as the conventional shift-map codes.

Acknowledgment

YY thanks Sae-Young Chung and Goo Jun for helpful discussions and comments.

Appendices

A. Identifying the discrete codebook in $N-1$ dimensional space.

For a given direction $\mathbf{a}$, points in $\mathcal{X}^*$ are calculated as follows. We first consider the intersections of $\mathcal{X}^{RRNS}$ and the faces of the unit cube and then project those intercepts onto H. Let H_n be the hyperplane that is orthogonal to the n'th axis and contains the origin:

$$H_n = \left\{ \mathbf{x} \in \mathbb{R}^N \mid \mathbf{e_n} \cdot \mathbf{x} = 0 \right\}, \tag{35}$$

where $\mathbf{e_n}$ is the unit vector with zero in all the coordinates other than the n'th coordinate. Then, $\mathcal{X}^{RRNS}$ intersects with the face of the unit hypercube on H_n at the origin and the other $(a_n - 1)$ points. This is summarized in the following lemma.

Lemma 2. *$\mathcal{X}^{RRNS}$ intersects with $H_n(\mathbf{a})$ at a_n number of points with corresponding source $S = \frac{i}{a_n}, i = 0, 1, \ldots, a_n - 1$: Let*

$$\begin{aligned} \mathcal{X}_n(\mathbf{a}) &\equiv H_n \cap \mathcal{X}^{RRNS} && (36) \\ &= \left\{ \mathbf{a}S \ mod\ 1 \;\middle|\; S = \frac{i}{a_n}, i = 0, 1, \ldots, a_n - 1 \right\}, . && (37) \end{aligned}$$

Then,

$$|\mathcal{X}_n(\mathbf{a})| = a_n \tag{38}$$

Proof. By the definition, $\mathcal{X}_n$ is the set of points in $\mathcal{X}$ with the nth coordinate being zero, which is equivalent to

$$a_n S = 0 \pmod 1. \tag{39}$$

Since $0 \le S < 1$, $0 \le a_n S < a_n$. Thus, (39) has a_n solutions of S, namely $S = \frac{i}{a_n}, i = 0, 1, \ldots, a_n - 1$. □

Next, the primality of a_n's in the construction of the RRNS-map code implies that the projections of $\mathcal{X}_n \setminus \mathbf{0}$ onto H are disjoint as stated in the following lemma.

Lemma 3. *If $n \neq m$,*

$$\mathcal{X}_n(\mathbf{a}) \cap \mathcal{X}_m(\mathbf{a}) = \mathbf{0}, \tag{40}$$

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

Proof. Trivially, $\mathbf{0} \in X_n$ for all $n = 1, 2, \ldots, N$. Next, it is shown by contradiction that there is no other element in the intersection. Suppose that there exists an element other than $\mathbf{0}$ in the intersection: $\mathbf{x} \in \mathcal{X}_n \cap \mathcal{X}_j$ and $\mathbf{x} \neq \mathbf{0}$. This implies that there exists $s \in (0, 1)$ satisfies (39) for both a_n and a_m. In other words, $a_n S = m$ and $a_m S = n$ with integers $0 < m < a_n - 1$ and $0 < n < a_m - 1$. Since a_n and a_m are relatively prime by construction, this can happen only when $S = 0$, which contradicts the assumptions that $\mathbf{x} \neq \mathbf{0}$. □

Lemma 2 and 3 lead to the following theorem showing that the number of points in $\mathcal{X}^*$ is related to the sum of a_n's.

Theorem 2. *Given a RRNS-map code with* $\mathbf{a}$ *satisfying (14), the cardinality of the discrete codebook* $\mathcal{X}^*$*, the intersections between the codebook* $\mathcal{X}^{RRNS}$ *and the orthogonal hyperplane* H *in (20), is as follows:*

$$|\mathcal{X}^*| \quad = \quad 1 + \sum_{n=1}^{N} (a_n - 1) \tag{41}$$

Following the same procedure, one can identify all the points in $\mathcal{X}^*$ by first finding $\mathcal{X}_n(\mathbf{a})$ and then projecting those points and the origin onto H. Let $\mathbf{x}_{ni}$ be an element in $X_n(\mathbf{a})$ with corresponding source $S = \frac{i}{a_n}$, $i \neq 0$ from (37):

$$\mathbf{x}_{ni} = \left(\frac{a_1}{a_n} i, \ldots, \frac{a_{n-1}}{a_n} i, 0, \frac{a_{n+1}}{a_n} i, \ldots, \frac{a_N}{a_n} j \right) \bmod 1, \tag{42}$$

where $n = 1, 2, \ldots N$ and $i = 1, \ldots a_n - 1$. In order to project this point onto H, one needs to 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,

$$\mathbf{a}^T = \begin{bmatrix} a_1 & a_2 & \ldots & a_N \end{bmatrix} = \mathbf{U} \mathbf{\Sigma} \mathbf{V}^T. \tag{43}$$

$\mathbf{\Sigma}$ 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 $\mathbf{x}_{ni}$ onto H is

$$\mathbf{x}^*_{ni} = \mathbf{B}^T (\mathbf{x}_{ni} - \mathbf{c}), \; n = 1, 2, \ldots, N, i = 1, 2, \ldots, a_n - 1 \tag{44}$$

where $\mathbf{c} = \left(\frac{1}{2}, \frac{1}{2}, \ldots, \frac{1}{2} \right)$ is the center of the hypercube. From (44), we have the full description of individual points in $\mathcal{X}^*$, which allows us to numerically calculate the minimum distance for a given $\mathbf{a}$.

B. Calculating the probability of threshold error using the union bound

In order to calculate the probability of threshold error, probability of threshold error to a neighboring segment is calculated. Suppose that true codeword $\mathbf{X}$ is in the i'th segment and let E_{ij} be the event that this codeword is decodeed to another segment j by noise. Corresponding probability P_{ij} is as follows:

$$P_{ij} \quad = \quad 1 - \Phi\left(\frac{d_{ij}}{2\sigma}\right), \tag{45}$$

where d_{ij} is the distance between segments i and j and $\Phi(x)$ is the cumulative distribution of the standard normal distribution defined by $\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{x} e^{-\frac{t^2}{2}}dt$.

Then, the threshold error event is the union of E_{ij} with j being neighbors of i. Thus, the probability of threshold error is

$$P_{th} \quad = \quad P\left(\bigcup_{j\in N(i)} E_{ij}\right), \tag{46}$$

where $N(i)$ represents the neighbors of i excluding i. Considering the union bound of (46), we have an upper bound on P_{th} with the union replaced by the summation of corresponding probabilities:

$$P_{th} \quad \leq \quad \sum_{j\in N(i)} P_{ij} \leq K\left(1 - \Phi\left(\frac{d_{min}}{2\sigma}\right)\right) \tag{47}$$

where K is the number of neighbors and d_{min} is the minimum distance. When dimension is high, this upper bound is known to be a tight approximation of the threshold error [27].

C. The lower bound of distortion of the shift-map codes for a given α.

A lower bound of the distortion of the shift-map code with a given scaling parameter α is calculated as follows. Since the first encoded variable $X_1 = S$ is most sensitive to the additive noise Z_1, let's consider the case where a large error occurs in the estimate of S because the observation $\mathbf{Y}$ is pushed toward to a different segment of the code segments in the first dimension. To be specific, when $|Z_1| > \frac{1}{2\alpha}$ and $Z_n = 0$ for $n > 1$, the estimation error is $\frac{1}{\alpha}$. In addition, if $|Z_1| > \frac{1}{2\alpha}$ and $Z_n (n > 1)$ have the same sign as Z_1, the estimation error only increases. Let's denote the set of such $\mathbf{Z}$ as $\mathcal{T}_1 \subset \mathcal{T}$. The distortion considering only $\mathbf{Z} \in \mathcal{T}_1$ is a lower bound of the distortion considering all $\mathbf{Z} \in \mathcal{T}$. Therefore, the second term in (4) is greater than $\text{erfc}\left(\frac{1}{2\sqrt{2}\alpha\sigma}\right)\left(\frac{1}{2}\right)^{(N-2)}\frac{1}{\alpha^2}$, where erfc is the complementary error function. Consequently, the distortion of the shift-map code with α is bounded from below by:

$$D \geq \left(1 - P_{th}^{U}\right)\frac{\sigma^2}{(L^{SM}(\alpha))^2} + \text{erfc}\left(\frac{1}{2\alpha\sigma}\right)\left(\frac{1}{2}\right)^{(N-2)}\frac{1}{\alpha^2}, \tag{48}$$

where P_{th}^{U} is the union bound of the probability of threshold error calculated similarly to (47), $L^{SM}(\alpha)$ is the stretch factor defined in (8).

References

[1] W. Softky, C. Koch, The highly irregular firing of cortical cells is inconsistent with temporal integration of random EPSPs, The Journal of Neuroscience 13 (1) (1993) 334–350.

[2] M. N. Shadlen, W. T. Newsome, The variable discharge of cortical neurons: Implications for connectivity, computation, and information coding, The Journal of Neuroscience 18 (10) (1998) 3870–3896.

[3] C. F. Stevens, Y. Wang, Changes in reliability of synaptic function as a mechanism for plasticity, Nature 371 (6499) (1994) 704–707.

[4] M. A. Paradiso, A theory for the use of visual orientation information which exploits the columnar structure of striate cortex, Biological Cybernetics 58 (1) (1988) 35–49.

[5] A. Georgopoulos, A. Schwartz, R. Kettner, Neuronal population coding of movement direction, Science 233 (4771) (1986) 1416–1419.

[6] H. S. Seung, H. Sompolinsky, Simple models for reading neuronal population codes, Proceedings of the National Academy of Sciences of the United States of America 90 (22) (1993) 10749–10753.

[7] P. Dayan, L. F. Abbott, Theoretical neuroscience : computational and mathematical modeling of neural systems, Massachusetts Institute of Technology Press, Cambridge, Mass., 2001.

[8] J. O'Keefe, L. Nadel, The hippocampus as a cognitive map, Behavioral and Brain Sciences 2 (04) (1979) 487–494.

[9] T. Hafting, M. Fyhn, S. Molden, M.-B. Moser, E. I. Moser, Microstructure of a spatial map in the entorhinal cortex, Nature 436 (2005) 801–806.

[10] H. Stensola, T. Stensola, T. Solstad, K. Frøland, M.-B. Moser, E. I. Moser, The entorhinal grid map is discretized, Nature 492 (7427) (2012) 72–78.

[11] I. R. Fiete, Y. Burak, T. Brookings, What grid cells convey about rat location, Journal of Neuroscience 28 (27) (2008) 6858–6871.

[12] S. Sreenivasan, I. Fiete, Grid cells generate an analog error-correcting code for singularly precise neural computation, Nature Neuroscience (2011) 1330–1337.

[13] B. Chen, G. Wornell, Analog error-correcting codes based on chaotic dynamical systems, Communications, IEEE Transactions on 46 (7) (1998) 881 –890.

[14] V. Vaishampayan, S. Costa, Curves on a sphere, shift-map dynamics, and error control for continuous alphabet sources, Information Theory, IEEE Transactions on 49 (7) (2003) 1658 – 1672.

[15] M. Taherzadeh, A. Khandani, Single-sample robust joint source-channel coding: Achieving asymptotically optimum scaling of SDR versus SNR, Information Theory, IEEE Transactions on 58 (3) (2012) 1565 –1577.

[16] R. Watson, C. Hastings, Self-checked computation using residue arithmetic, Proceedings of the IEEE 54 (12) (1966) 1920 – 1931.

[17] S.-S. Yau, Y.-C. Liu, Error correction in redundant residue number systems, Computers, IEEE Transactions on C-22 (1) (1973) 5 – 11.

[18] H. Krishna, K.-Y. Lin, J.-D. Sun, A coding theory approach to error control in redundant residue number systems. I. theory and single error correction, Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on 39 (1) (1992) 8 –17.

[19] M. A. Soderstrand, W. K. Jenkins, G. A. Jullien, F. J. Taylor (Eds.), Residue number system arithmetic: modern applications in digital signal processing, IEEE Press, Piscataway, NJ, USA, 1986.

[20] A. Wyner, On source coding with side information at the decoder, Information Theory, IEEE Transactions on 21 (3) (1975) 294 – 300.

[21] A. Wyner, J. Ziv, The rate-distortion function for source coding with side information at the decoder, Information Theory, IEEE Transactions on 22 (1) (1976) 1 – 10.

[22] C. Shannon, Communication in the presence of noise, Proceedings of the IRE 37 (1) (1949) 10–21.

[23] M. Baake, U. Grimm, D. H. Warrington, Some remarks on the visible points of a lattice, Journal of Physics A: Mathematical and General 27 (8).

[24] J. Conway, N. Sloane, E. Bannai, Sphere Packings, Lattices, and Groups, 3rd Edition, Springer, 1999.

[25] C. B. Barber, D. P. Dobkin, H. Huhdanpaa, The quickhull algorithm for convex hulls, ACM Transactions on Mathematical Software 22 (4) (1996) 469–483.

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

[27] J. Forney, G.D., G. Ungerboeck, Modulation and coding for linear gaussian channels, Information Theory, IEEE Transactions on 44 (6) (1998) 2384 –2415.