Efficient Channel Coding for Analog Sources using Chaotic Systems

Brian Chen *Gregory W. Wornell*
Department of Electrical Engineering and Computer Science, and
Research Laboratory of Electronics
Massachusetts Institute of Technology
Cambridge, MA 02139

ABSTRACT

We explore the application of chaotic sequences for encoding and transmission of analog sources over channels with unknown or multiple signal-to-noise ratios, as occur in broadcast and fading scenarios. Lower bounds on the mean-square distortion are derived for codes based on so-called tent-map dynamics, and are compared with those of other codes. For additive white Gaussian noise channels, we show there always exists a power-bandwidth regime in which this code yields lower distortion than any digital (i.e., finite-alphabet) code. We also develop and evaluate three practical decoding algorithms for efficiently exploiting these new codes on intersymbol interference channels.

1. INTRODUCTION

Among many other interesting properties, chaotic systems possess the sensitivity to initial condition property, i.e., state trajectories corresponding to nearby initial states diverge exponentially fast. Although prediction of the future states of a chaotic system is thus fundamentally difficult, this sensitivity is actually advantageous in the estimation of past states [1]. Hence, if one embeds information in the initial state of a chaotic system, the resulting state sequence forms a natural error correction code, where code sequences corresponding to nearby initial states eventually separate. It is this concept that we explore in this paper.

In particular, we explore the potential application of chaotic sequences for joint source-channel coding of analog, discrete-time data. A traditional digital approach to this communication problem has been to quantize the source data and encode the quantized data using some suitable channel code so that the quantized data can be recovered with arbitrarily low probability of error. Although such approaches can be optimal over additive white Gaussian noise (AWGN) channels with known signal-to-noise ratios (SNRs), this separation of lossy source coding (quantization) and channel coding is suboptimal when the SNR is unknown or when there are multiple SNRs. Indeed, the performance of these digital approaches depend crucially on being able to choose the proper number of quantization levels, which in turn depends on the SNR. See, e.g., Trott [2] for a discussion of several aspects of the suboptimality of separate source and channel coding for the broadcast channel.

This work has been supported in part by ARPA monitored by ONR under Contract No. N00014-93-1-0686, the Office of Naval Research under Grant No. N00014-96-1-0930, and a National Defense Science and Engineering Graduate Fellowship.

0-7803-3336-5/96 $5.00 © 1996 IEEE

There are a wide range of scenarios in which the need to transmit data over a channel with unknown or multiple SNRs is encountered. For example, in a broadcast context, SNR typically varies from receiver to receiver. Another application involves low-delay communication over a time-selective fading channel, where the SNR fluctuates over time in a manner that is unknown *a priori.* Motivated by the knowledge that digital channel coding of quantized data can be suboptimal in these situations, in this paper we explore the use of chaotic systems to implement analog codes for the transmission of analog source data over channels with unknown or, equivalently, multiple SNRs.

We consider first the coding of a uniformly distributed source for the AWGN channel. For simplicity of exposition, we restrict our attention to real-valued baseband channels; extensions to more typical complex equivalent baseband channels are straightforward. Fig. 1 illustrates the problem considered. The source letter x_0 has a uniform density on the interval $[-1, 1]$ and is mapped into a sequence $x[n]$ of length N, i.e., we constrain the encoder to expand the bandwidth by a factor of N. We also constrain the average signal energy per dimension to be P:

$$P = \frac{1}{N} \sum_{n=0}^{N-1} E\left\{x^2[n]\right\}. \qquad (1)$$

This sequence passes through an AWGN channel where the noise $w[n]$ has zero mean and variance σ_w^2. Then, the SNR is, therefore,

$$\mathrm{SNR} = \frac{P}{\sigma_w^2}. \qquad (2)$$

Finally, the decoder estimates x_0 from the channel output $y[0], \cdots, y[N-1]$. The source-channel coding problem we consider is that of finding a code with small distortion for a given SNR and bandwidth, where the distortion measure of interest is mean-square error, $E\{(\hat{x}_0 - x_0)^2\}$.

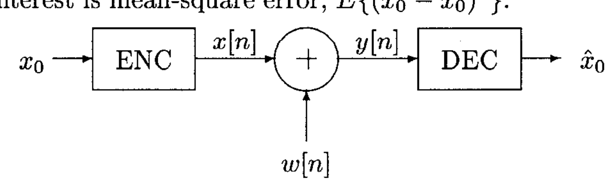

Figure 1. *Joint Source-Channel Coding of a Uniform Source over an AWGN Channel*

2. THE TENT MAP CODE

The proposed encoder maps each analog source "letter" to a chaotic sequence generated from a nonlinear dynamical system with the source letter as an initial value. Although any of a variety of discrete-time chaotic systems is suitable for this purpose—and all lead to the same qualitative behavior—we will focus on the system whose dynamics are governed by the symmetric tent map, for which efficient signal processing algorithms are readily available. With these systems, the resulting chaotic sequences obey the following one-dimensional dynamics

$$x[n] = F(x[n-1]), \tag{3}$$

where

$$F(x) = 1 - 2|x|. \tag{4}$$

Specifically, the encoder maps the source letter x_0 into the sequence $x[n] = F^{(n)}(x_0)$ where $F^{(n)}(\cdot)$ is used to denote the n-fold iteration of the tent map. As discussed in [1], for almost all x_0, $x[n]$ will be an ergodic sequence of random variables uniformly distributed on the interval $[-1, 1]$.

We exploit three different but interrelated interpretations of this tent map code.

1. The tent map code is the state trajectory of a chaotic system (3) with the source letter x_0 embedded in the initial state.
2. The tent map code is the superposition of less significant bits in the quantization of x_0 on top of a BPSK encoding of more significant bits.
3. The tent map code corresponds to nonlinearly modulating a set orthogonal sequences with the source letter x_0. In this case the orthogonal sequences are $\phi_i[n] = \delta[n-i]$, $0 \le i \le N-1$. The nonlinear modulating functions are $F^{(i)}(x_0)$, so that

$$x[n] = \sum_{i=0}^{N-1} F^{(i)}(x_0)\phi_i[n]. \tag{5}$$

As we will see, each of these interpretations provide useful and complementary insights into the performance of the code.

Papadopoulos and Wornell have developed the maximum likelihood (ML) estimator for such sequences in AWGN [1]. This estimator can be implemented as the cascade of a nonlinear recursive filter and corresponding smoother. The decoder for our tent map code is simply this ML estimator followed by a sampler which samples the estimator output at $n = 0$, giving the ML estimate of x_0.

3. TENT MAP VS. DIGITAL CODING

In this section we compare the expected distortion resulting from the tent map code with the smallest achievable distortion using a channel code with a finite alphabet size of M on the quantized source. Our main result is that for any finite M there always exists some SNR and N such that the tent map code yields superior performance.

To begin, as developed in [3] the mean-square error of the ML estimator in estimating x_0 decays exponentially with N to a threshold. Specifically, for small N the decay takes the form

$$D_{\text{tent}} \le \left(\frac{1}{2}\right)^{2(N-1)} \frac{1/3}{\text{SNR}}, \tag{6}$$

while the threshold level is inversely proportional to $\text{SNR}^{3/2}$, i.e., for large N,

$$D_{\text{tent}} \approx \frac{0.18}{\text{SNR}^{3/2}}. \tag{7}$$

In practice, D_{tent} is effectively the maximum of (6) and (7).

Next, a few information-theoretic results allow us to obtain a lower bound on the distortion for any code involving an M-ary quantization [3]. Given separation of the (lossy) source encoder and channel encoder, the following inequality holds:

$$R(D) \le NC, \tag{8}$$

where $R(D)$ is the rate-distortion function and C is the channel capacity. When the source is uniform on the interval $[-1, 1]$, it can be shown that [3]

$$R(D) = \frac{1}{2}\log_2 \frac{2}{\pi e D}. \tag{9}$$

Combining (9) with (8) yields, after some minor algebra:

$$D \ge \frac{2}{\pi e}(2^2)^{-NC}. \tag{10}$$

Thus, we have a lower bound on the distortion which depends only on the channel capacity. When the channel input is constrained to be one of M letters,

$$C_M \le \log_2 M, \tag{11}$$

with equality if and only if the input can be determined from the output with no uncertainty. Thus, combining (11) with (10) leads to the inequality

$$D_M \ge \frac{2}{\pi e}(M^2)^{-N}. \tag{12}$$

Comparison of (6) and (7) with (12) yields the sufficient, but not necessary, conditions under which the tent map code is guaranteed to result in smaller expected distortion than any M-ary code, viz.,

$$\text{SNR} \ge \left(7.55 + 10N\log_{10}\frac{M^2}{4}\right) \text{dB} \tag{13}$$

and

$$\text{SNR} \ge \left(-0.762 + 10N\frac{2}{3}\log_{10} M^2\right) \text{dB} \tag{14}$$

Eqs. (13) and (14) define an operating region in the power-bandwidth plane: in regions corresponding to high power (SNR) and low bandwidth (N), tent map coding results in smaller distortion than any M-ary code. The boundaries for the regions corresponding to M = 2,4, and 8 are plotted in Fig. 2. Additional results, including a method for combining M-ary coding with tent map coding to make M-ary codes more robust to fluctuations in SNR, are developed in [3].

3.1. Additional Perspectives

Useful interpretations of our chaotic codes in the context of digital codes results from exploiting a natural correspondence between the dynamics of tent map systems and those of infinite length binary shift registers. Specifically, given

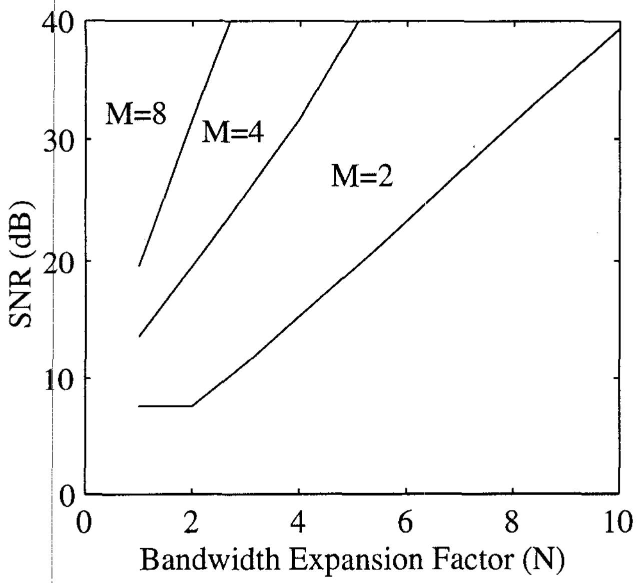

Figure 2. *The regions in the power-bandwidth plane in which the tent map code results in lower expected distortion than M-ary codes lie above the curves, i.e., high SNR and low bandwidth.*

any binary sequence $b[0], b[1], \cdots$ of ± 1, there exists an initial state $x[0]$ such that

$$\operatorname{sgn}\{x[n]\} = b[n]. \tag{15}$$

Given the state $x[n]$, one can obtain every subsequent binary element via

$$b[n+k] = \operatorname{sgn}\left\{F^{(k)}(x[n])\right\}, \qquad \forall k \geq 0. \tag{16}$$

Thus, any digital encoder whose output is a function of the contents of some binary shift register may be represented by a tent map system with the appropriate choice of observation function $g(x[n])$. Specifically, if the encoder is some function $\psi(b[n], b[n+1], \cdots, b[n+k])$, then

$$g(x[n]) = \psi\left(\operatorname{sgn}\{x[n]\}, \operatorname{sgn}\{F(x[n])\}, \cdots, \operatorname{sgn}\left\{F^{(k)}(x[n])\right\}\right).$$

Similarly, one can represent the dynamics of a register that shifts R bits at a time with the state evolution function $F^{(R)}(\cdot)$. Then, one can emulate, for example, a rate-k/N, 2^K-state convolutional encoder with the one-dimensional dynamic system,

$$\begin{aligned} x[n+1] &= F^{(k)}(x[n]) \\ y[n] &= g(x[n]), \end{aligned} \tag{17}$$

where $g(\cdot)$ is a piecewise constant function that maps 2^{K+k} intervals into one of 2^N possible channel inputs.

The tent map code we consider in this paper uses the identity function as the observation function. In this case the equivalent digital encoder has an infinite length shift register with encoding function

$$\psi(b[n], b[n+1], \cdots) = \frac{1}{2}\sum_{k=0}^{\infty}\left(-\frac{1}{2}\right)^k b[n]b[n+1]\cdots b[n+k]. \tag{18}$$

Furthermore, the binary sequence $b[0], b[1], \cdots$ is directly connected to the source letter x_0. Specifically, the sequence is the Gray code quantization of x_0, $b[0]$ being the most significant bit. Thus, we can interpret $x[n]$ as the superposition of less significant bits on top of 2-PAM (BPSK) transmission of the nth bit $b[n]$.

4. TENT MAP CODING VS. LINEAR MODULATION

We now turn our attention to the third interpretation of the tent map code—that it corresponds to nonlinear modulation of a set of orthogonal sequences with the source letter x_0. Then, it is natural to ask how this modulation compares to linear modulation. Linear modulation codes are codes in which the source letter simply multiplies some signature sequence of length N. A common example of such codes is the repetition code, in which the signature sequence is a sequence of N ones. In contrast to linear modulation codes, the tent map code can be interpreted as a type of nonlinear modulation, referred to as "twisted modulation" by Wozencraft and Jacobs [4]. Connections between the tent map code and general twisted modulation systems are developed in [3].

The expected distortion for all linear modulation codes over the AWGN channel is

$$D_{\text{lin}} = \frac{1/3}{N}\frac{1}{\text{SNR}} \tag{19}$$

Again, comparison of (6) and (7) with (19) allows one to find the region in the power-bandwidth plane where the tent map code results in lower distortion than linear modulation codes. In particular, since (6) is always less than (19) for $N > 1$, the nontrivial condition is that (7) be less than (19). This condition is equivalent to

$$\text{SNR} \geq 0.29N^2 \tag{20}$$

Fig. 3 shows the region in the power-bandwidth plane where the tent map code results in lower distortion than linear modulation codes. A source sequence of 1000 uniformly distributed random variables on the interval $[-1, 1]$ was encoded using both the tent map and repetition codes with $2 \leq N \leq 15$ and $0 \leq \text{SNR} \leq 25$ dB, and the resulting distortion in the decoded sequence was measured. The dashed curve $(- - -)$ represents the theoretically predicted boundary given by (20). As in the digital comparison, we see that the tent map code is better than linear modulation codes at high power or low bandwidth.

5. TENT MAP CODING FOR THE UNKNOWN ISI CHANNEL

We now consider chaotic codes in the context of the more general class of noisy channels characterized by dispersion in the form of convolutional distortion. As we develop in this section, tent map coding can be highly effective on these channels when the associated intersymbol interference (ISI) is removed by an appropriately designed equalizer that is matched to these codes.

Fig. 4 illustrates the channel and receiver structure we consider. The transmitted signal $x[n]$ is convolved with the channel impulse response $h[n]$, which we assume to be unknown, and the received signal $r[n]$ is a noisy version of

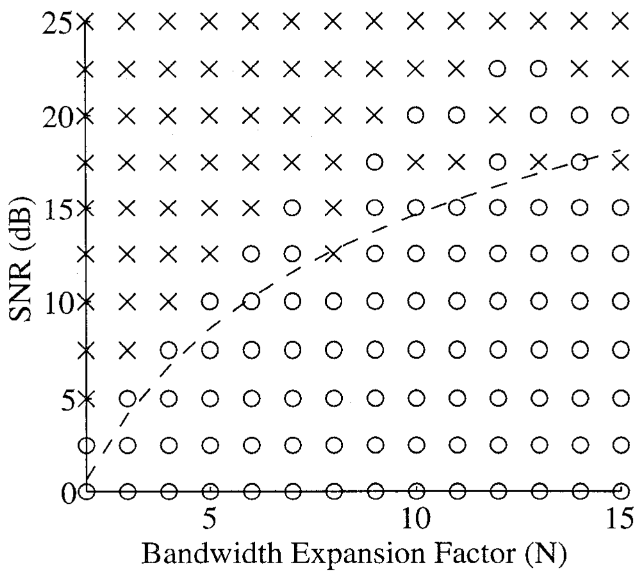

Figure 3. *The experimentally determined region in the power-bandwidth plane where the tent map code resulted in lower distortion than the repetition code is marked with ×'s. The region where the repetition code resulted in lower distortion is marked with ○'s. The dashed line is the theoretically predicted boundary.*

the channel output. The equalizer, a filter with impulse response $g[n]$, produces an estimate of $x[n]$, denoted $\tilde{x}[n]$. We constrain the equalizer to be an M-tap finite impulse response filter, so the deconvolution problem becomes one of choosing the appropriate filter taps. After equalization, subsequent processing removes residual noise from $\tilde{x}[n]$ and recovers the source data. In this paper, we use the decoder for the AWGN channel for this latter purpose.

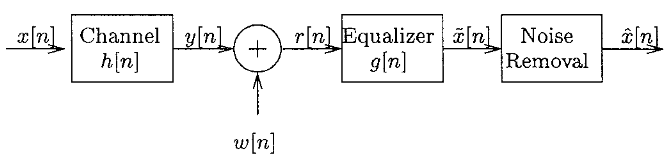

Figure 4. *Removal of ISI through Equalization*

5.1. Equalization Algorithms

In the sequel, we summarize three distinct blind equalization algorithms proposed and developed in [3] for use with tent map encoded data on the unknown ISI channel as described above.

5.1.1. Dynamics Matching Algorithm

The dynamics matching algorithm selects the equalizer taps

$$\hat{\mathbf{g}} = [\hat{g}[0]\ \hat{g}[1]\ \dots\ \hat{g}[M-1]]^T \tag{21}$$

based on the dynamics matching criteria suggested by Isabelle [5], which corresponds to choosing the $\hat{\mathbf{g}}$ that minimizes

$$J(\mathbf{g}) = \sum_{n \neq kN} \left(\tilde{x}[n] - F(\tilde{x}[n-1])\right)^2. \tag{22}$$

We use the following iterative algorithm to minimize $J(\mathbf{g})$.

1. Let $k = 0$ and guess a solution $\hat{\mathbf{g}}^{(k)}$.
2. Calculate $\tilde{x}^{(k)}[n-1] = \hat{g}^{(k)}[n-1] * r[n-1]$. Form the matrix $\mathbf{R}^{(k)}$ with elements
$$[\mathbf{R}^{(k)}]_{ij} = r[i-(j-1)] + 2\,\mathrm{sgn}\left\{\tilde{x}^{(k)}[i-1]\right\} r[i-1-(j-1)],$$
where $1 \leq i \leq NL$, $1 \leq j \leq M$, and L is the number of source letters per block.
3. Remove the rows of $\mathbf{R}^{(k)}$ for which the row index i is a multiple of N, and use $\tilde{\mathbf{R}}^{(k)}$ to denote the resulting matrix.
4. Find a new solution:
$$\hat{\mathbf{g}}^{(k+1)} = \left[\left(\tilde{\mathbf{R}}^{(k)}\right)^{\mathrm{T}} \tilde{\mathbf{R}}^{(k)}\right]^{-1} \left(\tilde{\mathbf{R}}^{(k)}\right)^{\mathrm{T}} \mathbf{b}. \tag{23}$$
where $\mathbf{b}$ is a vector of all ones.
5. Increment k and return to step 2. Continue iterating until $\hat{\mathbf{g}}^{(k+1)} = \hat{\mathbf{g}}^{(k)}$.

A computationally efficient recursive implementation of this algorithm is also developed in [3].

5.1.2. Alternating Projections Algorithm

The alternating projections algorithm selects the equalizer taps to minimize the mean-square distance between the equalizer output and the closest sequence composed of concatenated, length N tent map sequences. It performs the minimization by alternately projecting onto the set of equalizer outputs and onto the set of concatenated tent map sequences. The algorithm below converges monotonically to a local minimum.

1. Form the matrix $\mathbf{R}$, where
$$[\mathbf{R}]_{ij} = r[i-j]. \tag{24}$$
Note that $\mathbf{R}\mathbf{g}$ is the equalizer output, the convolution of $r[n]$ and $g[n]$.
2. Let $k = 0$ and guess a set of equalizer taps $\hat{\mathbf{g}}^{(k)}$.
3. Parse the sequence $\mathbf{R}\hat{\mathbf{g}}^{(k)}$ into sequences of length N and filter and smooth the parsed sequences with the ML estimator for tent map sequences in AWGN.
4. Concatenate the smoothed sequences and call this sequence $\hat{\mathbf{x}}^{(k+1)}$. Note that $\hat{\mathbf{x}}^{(k+1)}$ is the sequence composed of concatenated, length N tent map sequences closest in a mean-square distance sense to $\mathbf{R}\hat{\mathbf{g}}^{(k)}$.
5. Project $\hat{\mathbf{x}}^{(k+1)}$ onto the column space of $\mathbf{R}$ to find $\hat{\mathbf{g}}^{(k+1)}$:
$$\hat{\mathbf{g}}^{(k+1)} = \left(\mathbf{R}^{\mathrm{T}}\mathbf{R}\right)^{-1} \mathbf{R}^T \hat{\mathbf{x}}^{(k+1)}. \tag{25}$$
6. Increment k and return to step 3. Continue iterating until $\|\hat{\mathbf{x}}^{(k+1)} - \hat{\mathbf{x}}^{(k)}\| < \epsilon$ for some ϵ.

5.1.3. Supex-Based Algorithm

The superexponential or "supex" algorithm developed by Shalvi and Weinstein [6] uses higher-order statistics of the output to estimate the equalizer taps, and is effective when the channel input satisfies some second- and fourth-order cumulant whiteness conditions. Under such conditions, the impulse response of the cascaded channel and equalizer approaches a delay with each iteration. Specifically, the largest tap of the cascaded system grows exponentially relative to all the other taps.

It can be shown that tent map sequences downsampled by two approximately satisfy the second- and fourth-order whiteness constraints [3]. Exploiting this property, Fig. 5 illustrates a method for using the supex algorithm for deconvolution of tent map encoded data. The tent map sequence $x[n]$ is separated into even and odd phases through downsampling so that the even numbered samples pass through the upper channel and the odd numbered samples pass through the lower channel. The two phases are processed in parallel, with the channel impulse response assumed to be the same for both channels. The supex algorithm is run separately on the two received sequences $r_1[n]$ and $r_2[n]$ to find the sets of equalizer filter taps $g_1[n]$ and $g_2[n]$. After estimating the taps for the two phases separately, the system recombines the two estimates to obtain a composite estimate of the full original sequence.

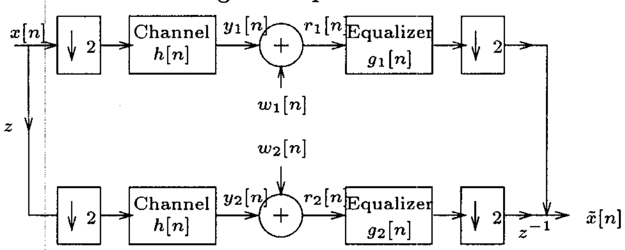

Figure 5. *Downsampling and Supex for Equalization of Tent Map Encoded Data*

5.2. Simulation Results

The three algorithms were tested on a source data sequence of 1024 independent, identically distributed (IID) random variables uniformly distributed on the interval $[-1, 1]$. Each of these variables was mapped onto a tent map sequence of length N, and L such sequences were concatenated together to form a block of length $NL = 1024$. (For the supex algorithm, L was chosen to make $NL = 2048$ so that each phase had length 1024 after downsampling.) Each algorithm was run on all of the blocks, and the average ISI across all blocks was calculated, as was the mean-square error distortion (D) in estimating the 1024 source data variables. The performance of the supex algorithm on uncoded source sequences is taken to be a somewhat arbitrary but useful baseline. The improvement in ISI and distortion over this baseline is shown in Figs. 6 and 7, respectively. The SNR and N were chosen to correspond to points on the power-bandwidth plane where tent map coding is superior to linear modulation coding over the AWGN (no ISI) channel. (cf. Fig. 3.)

A cursory inspection of Fig. 6 might lead one to conclude that tent map coding offers only moderate improvement in residual ISI over the baseline of using the supex algorithm on an uncoded sequence. Indeed, one would expect zero ISI gain for the case $N = 2$ when supex decoding is used since the even and odd phases are in fact IID sequences. However, from Fig. 7 we see that tent map coding offers substantial improvement in minimizing distortion over the baseline case, especially if dynamics matching or alternating projections decoding is used. Thus, if we interpret residual ISI as an effective noise source, we see that just as tent map coding can reduce distortion caused by AWGN, it can

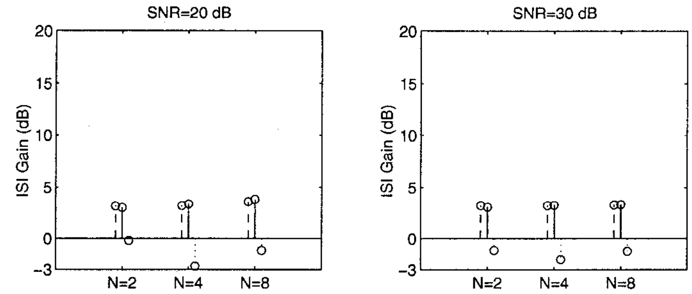

Figure 6. *Improvement over baseline in ISI using dynamics matching (dashed), alternating projections (solid), and supex (dotted) algorithms.*

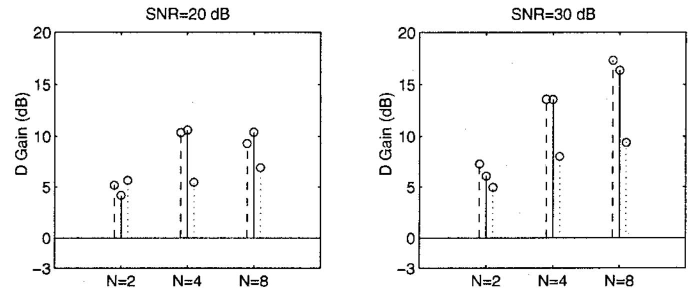

Figure 7. *Improvement over baseline in distortion D using dynamics matching (dashed), alternating projections (solid), and supex (dotted) algorithms.*

reduce "residual ISI noise" as well. Furthermore, that the gain in distortion reduction increases with SNR, even at fairly high SNR, suggests that the results demonstrating the superiority of tent map coding to M-ary coding at high power in the AWGN channel case may extend to the ISI channel case as well.

REFERENCES

[1] H. C. Papadopoulos and G. W. Wornell, "Maximum likelihood estimation of a class of chaotic signals," *IEEE Trans. Inform. Theory*, vol. 41, pp. 312–317, Jan. 1995.

[2] M. D. Trott, "Unequal error protection codes: Theory and practice," in *Proc. IEEE Inform. Th. Workshop*, p. 11, June 1996.

[3] B. Chen, "Efficient communication over additive white gaussian noise and intersymbol interference channels using chaotic sequences," RLE Tech. Rep. No. 598, M. I. T., Cambridge, MA, Apr. 1996.

[4] J. M. Wozencraft and I. M. Jacobs, *Principles of Communication Engineering.* New York, NY: John Wiley and Sons, 1965.

[5] S. H. Isabelle, *A Signal Processing Framework for the Analysis and Application of Chaos.* PhD thesis, M. I. T., Cambridge, MA, Feb. 1995. Also RLE Tech. Rep. No. 593, Feb. 1995.

[6] O. Shalvi and E. Weinstein, "Super-exponential methods for blind deconvolution," *IEEE Trans. Inform. Theory*, vol. 39, pp. 504–519, Mar. 1993.