Rate-Distortion via Markov Chain Monte Carlo

Shirin Jalali* and Tsachy Weissman*†, *Department of Electrical Engineering, Stanford University, Stanford, CA 94305, {shjalali, tsachy}@stanford.edu † Department of Electrical Engineering, Technion, Haifa 32000, Israel

Abstract

We propose an approach to lossy source coding, utilizing ideas from Gibbs sampling, simulated annealing, and Markov Chain Monte Carlo (MCMC). The idea is to sample a reconstruction sequence from a Boltzmann distribution associated with an energy function that incorporates the distortion between the source and reconstruction, the compressibility of the reconstruction, and the point sought on the rate-distortion curve. To sample from this distribution, we use a 'heat bath algorithm': Starting from an initial candidate reconstruction (say the original source sequence), at every iteration, an index i is chosen and the i-th sequence component is replaced by drawing from the conditional probability distribution for that component given all the rest. At the end of this process, the encoder conveys the reconstruction to the decoder using universal lossless compression.

The complexity of each iteration is independent of the sequence length and only linearly dependent on a certain context parameter (which grows sub-logarithmically with the sequence length). We show that the proposed algorithms achieve optimum rate-distortion performance in the limits of large number of iterations, and sequence length, when employed on any stationary ergodic source. These theoretical findings are confirmed by initial experimentation showing near Shannon limit performance in various cases.

Employing our lossy compressors on noisy data, with appropriately chosen distortion measure and level, followed by a simple de-randomization operation, results in a family of denoisers that compares favorably (both theoretically and in practice) with other MCMC-based schemes, and with the Discrete Universal Denoiser (DUDE).

Index Terms

Rate-distortion coding, Universal lossy compression, Markov chain Monte carlo, Gibbs sampler, Simulated annealing

I. INTRODUCTION

Consider the basic setup of lossy coding of a stationary ergodic source $\mathbf{X} = \{X_i : i \geq 1\}$. Each source output block of length n, X^n, is mapped to an index $f(X^n) \in \{1, 2, \dots, 2^{nR}\}$. The index is sent to the decoder which decodes it to a reconstruction block $\hat{X}^n = g(f(X^n))$. The performance of a coding scheme $\mathcal{C} = (f, g, n, R)$ is measured by its average expected distortion between source and reconstruction blocks, i.e.

$$D = Ed_n(X^n, \hat{X}^n) \triangleq \frac{1}{n}\sum_{i=1}^{n} Ed(X_i, \hat{X}_i), \tag{1}$$

where $d : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}^+$ is a single-letter distortion measure. Here $\mathcal{X}$ and $\hat{\mathcal{X}}$ denote the source and reconstruction alphabets respectively, which we assume are finite. For any $D \geq 0$, the minimum achievable rate (cf. [2] for exact definition of achievability) is characterized as [1], [3], [4]

$$R(\mathbf{X},D)=\lim_{n\to\infty}\min_{p(\hat{X}^n|X^n):Ed_n(X^n,\hat{X}^n)\leq D}\frac{1}{n}I(X^n;\hat{X}^n). \tag{2}$$

For the case of lossless compression, i.e. $D=0$ (assuming a non-degenerate distortion measure), we know that the minimum required rate is the entropy rate of the source, i.e. $R(\mathbf{X},0)=H(\mathbf{X})\triangleq\lim\limits_{k\to\infty}H(X_0|X_{-k}^{-1})$. Moreover, there are known implementable *universal* schemes, such as Lempel-Ziv coding [9] and arithmetic coding [11], that are able to describe any stationary ergodic source at rates as close as desired to the entropy rate of the source without any error. In contrast to the situation of lossless compression, for $D>0$, neither the explicit solution of (2) is known for a general source (even not for a first-order Markov source [36]), nor are there known practical schemes that universally achieve the rate-distortion curve. In recent years, there has been progress towards designing universal lossy compressor especially in trying to tune some of the existing universal lossless coders to work in the lossy case as well [12], [13], [14]. In [12], a lossy version of Lempel-Ziv algorithm at fixed distortion is rendered, and is shown to be optimal for memoryless sources. On the other hand, for the non-universal setting, specifically the case of lossy compression of an i.i.d. source with a known distribution, there is an ongoing progress towards designing codes that get very close to the optimal performance [22], [23], [24], [25].

In this paper, we present a new approach to implementable lossy source coding, which borrows two well-known tools from statistical physics and computer science, namely Markov Chain Monte Carlo (MCMC) methods, and simulated annealing [18],[19]. MCMC methods refer to a class of algorithms that are designed to generate samples of a given distribution through generating a Markov chain having the desired distribution as its stationary distribution. MCMC methods include a large number of algorithms; For our application, we use Gibbs sampler [17] also known as the *heat bath* algorithm, which is well-suited to the case where the desired distribution is hard to compute, but the conditional distributions of each variable given the rest are easy to work out.

The second required tool is simulated annealing which is a well-known optimization method. Its goal is to find the minimum of a function $f_{\text{min}}\triangleq\min f(s)$ along with the minimizing state s_{min} over a set of possibly huge number of states S. In order to do simulated annealing, a sequence of probability distributions $p_1,p_2,\ldots$ corresponding to the temperatures $T_1>T_2>\ldots$, where $T_i\to 0$ as $i\to\infty$, and a sequence of positive integers N_1, N_2, $\ldots$, are considered. For the first N_1 steps, the algorithm runs one of the relevant MCMC methods in an attempt to sample from distribution p_1. Then, for the next N_2 steps, the algorithm, using the output of the previous part as the initial point, aims to sample from p_2, and so on. The probability distributions are designed such that: 1) their output, with high probability, is the minimizing state s_{min}, or one of the states close to it, 2) the probability of getting the minimizing state increases as the temperature drops. The probability distribution that satisfies these characteristics, and is almost always used, is the Boltzman distribution $p_\beta(s)\propto e^{-\beta f(s)}$, where $\beta\propto\frac{1}{T}$. It can be proved that using Boltzman distribution, if the temperature drops slowly enough, the probability of ultimately getting the minimizing state as the output of the algorithm approaches one [17]. Simulated annealing has been suggested before in the context of lossy compression, either as a way for approximating the rate distortion function (i.e., the optimization problem involving minimization of the mutual information) or as a method for designing the codebook in vector quantization [20],[21], as an alternative to the conventional generalized Lloyd algorithm (GLA) [16]. In contrast, in this paper we use the simulated annealing approach to obtain a particular reconstruction sequence, rather than a whole codebook.

Let us briefly describe how the new algorithm codes a source sequence x^n. First, to each reconstruction block y^n, it assigns an *energy*, $\mathcal{E}(y^n)$, which is a linear combination of its conditional empirical entropy, to be defined formally in the next section, and its distance from the source sequence x^n. Then, it assumes a Boltzman probability distribution over the reconstruction blocks as $p(y^n) \propto e^{-\beta\mathcal{E}(y^n)}$, for some $\beta > 0$, and tries to generate $\hat{x}^n$ from this distribution using Gibbs sampling [17]. As we will show, for β large enough, with high probability the reconstruction block of our algorithm would satisfy $\mathcal{E}(\hat{x}^n) \approx \min \mathcal{E}(y^n)$. The encoder will output $\mathsf{LZ}(\hat{x}^n)$, which is the Lempel-Ziv [9] description of $\hat{x}^n$. The decoder, upon receiving $\mathsf{LZ}(\hat{x}^n)$, reconstructs $\hat{x}^n$ perfectly.

In this paper, instead of working at a fixed rate or at a fixed distortion, we are fixing the slope. A fixed slope rate-distortion scheme, for a fixed slope $s < 0$, looks for the coding scheme that minimizes $R - s \cdot D$, where as usual R and D denote the rate and the average expected distortion respectively. In comparison to a given coding scheme of rate R and expected distortion D, for any $0 < \delta < R - R(\mathbf{X}, D)$, there exists a code which works at rate $R(\mathbf{X}, D) + \delta$ and has the same average expected distortion, and consequently a lower cost. Therefore, it follows that any point that is optimal in the fixed-slope setup corresponds to a point on the rate-distortion curve.

The organization of the paper is as follows. In Section II, we set up the notation, and in Section III describe the count matrix and empirical conditional entropy of a sequence. Section IV describes an exhaustive search scheme for fixed-slope lossy compression which universally achieves the rate-distortion curve for any stationary ergodic source. Section V describes our new universal MCMC-based lossy coder, and Section VI presents another version of the algorithm for finding sliding-block codes which again universally attain the rate-distortion bound. Section VII gives some simulations results. Section VIII describes the application of the algortihm introduced in Section V to universal compression-based denoising. Finally, Section IX concludes the paper with a discussion of some future directions.

II. Notation

Let $\mathcal{X}$ and $\mathcal{Y}$ denote the source and reconstructed signals alphabets respectively. For simplicity, we restrict attention to the case where $\mathcal{X} = \mathcal{Y} = \{\alpha_1, \ldots, \alpha_N\}$, though our derivations and results carry over directly to general finite alphabets. Bold low case symbols, e.g. $\mathbf{x}, \mathbf{y}, \mathbf{z}$, denote individual sequences.

Let $d : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}^+$ be the loss function (fidelity criterion) which measures the loss incurred in decoding a symbol α_i to another symbol α_j. Moreover, let $d_m = \max\limits_{i,j} d(\alpha_i, \alpha_j)$, and note that $d_m < \infty$, since the alphabets are finite. The normalized cumulative loss between a source sequence x^n and reconstructed sequence $\hat{x}^n$ is denoted by $d_n(x^n, \hat{x}^n) = \frac{1}{n} \sum\limits_{i=1}^{n} d(x_i, \hat{x}_i)$.

III. Counts and Empirical Conditional Entropy

Let $H_k(y^n)$ denote the conditional empirical entropy of order k induced by y^n, i.e.

$$H_k(y^n) = H(Y_{k+1}|Y^k), \tag{3}$$

where Y^{k+1} on the right hand side of (3) is distributed according to

$$P(Y^{k+1} = u^{k+1}) = \frac{1}{n}\left|\left\{1 \le i \le n : y_{i-k}^i = u^{k+1}\right\}\right|, \tag{4}$$

where in (4), and throughout we assume a cyclic convention whereby $y_i \triangleq y_{n+i}$ for $i \le 0$. We introduce the count notation $\mathbf{m}_k(y^n, u^k)$, which is a column vector counting the number of appearances of the different symbols in y^n with the left context u^k. More explicitly, $\mathbf{m}_k(y^n, u^k)$ is a column vector whose a-th component, $a \in \mathcal{Y}$, is given by

$$\mathbf{m}_k(y^n, u^k)[a] = \left|\left\{1 \le i \le n : y_{i-k}^i = u^k a\right\}\right|, \tag{5}$$

where $u^k a$ denotes the $(k+1)$-tuple obtained by concatenating u^k with the symbol a. We let $\mathbf{m}_k(y^n, \cdot)$ denote the $|\mathcal{Y}| \times |\mathcal{Y}|^k$ matrix whose columns are given by $\mathbf{m}_k(y^n, u^k)$, for the $|\mathcal{Y}|^k$ values of u^k lexicographically ordered. Note that, with the count notation, the conditional empirical entropy in (3) can be expressed as

$$H_k(y^n) = \frac{1}{n}\sum_{u^k} \mathcal{H}\left(\mathbf{m}_k(y^n, u^k)\right) \mathbf{1}^T \mathbf{m}_k(y^n, u^k), \tag{6}$$

where $\mathbf{1}$ denotes the all-ones column vector of length $|\mathcal{Y}|$, and for a vector $v = (v_1, \ldots, v_\ell)^T$ with non-negative components, we let $\mathcal{H}(v)$ denote the entropy of the random variable whose probability mass function (pmf) is proportional to v. Formally,

$$\mathcal{H}(v) = \begin{cases} \sum_{i=1}^{\ell} \frac{v_i}{\|v\|_1} \log \frac{\|v\|_1}{v_i} & \text{if } v \neq (0, \ldots, 0)^T \\ 0 & \text{if } v = (0, \ldots, 0)^T. \end{cases} \tag{7}$$

The important point is that H_k is sum of terms over u^k involving only $m_k(y^n, u^k)$.

IV. An exhaustive search scheme for fixed-slope compression

Consider the following scheme for lossy source coding at fixed slope $s \le 0$. For each source sequence x^n let the reconstruction block $\hat{x}^n$ be

$$\hat{x}^n = \arg\min_{y^n} \left[H_k(y^n) - s \cdot d(x^n, y^n)\right]. \tag{8}$$

The encoder, after computing $\hat{x}^n$, losslessly conveys it to the decoder using LZ compression. Let k grow slowly enough with n so that

$$\limsup_{n \to \infty} \max_{y^n} \left[\frac{1}{n}\ell_{\mathsf{LZ}}(y^n) - H_k(y^n)\right] \le 0, \tag{9}$$

where $\ell_{\mathsf{LZ}}(y^n)$ denotes the length of the LZ representation of y^n. Note that Ziv's inequality guarantees that if $k = k_n = o(\log n)$ then (9) holds. We can prove the following theorem whose proof is given in Appendix A.

Theorem 1: Let $\mathbf{X}$ be a stationary and ergodic source, let $R(\mathbf{X}, D)$ denote its rate distortion function, and let $\hat{X}^n$ denote the reconstruction using the above scheme on X^n. Then

$$\mathbb{E}\left[\frac{1}{n}\ell_{\mathsf{LZ}}(\hat{X}^n) - s \cdot d(X^n, \hat{X}^n)\right] \overset{n\to\infty}{\longrightarrow} \min_{D\geq 0}\left[R(\mathbf{X}, D) - s \cdot D\right]. \tag{10}$$

V. Universal lossy coding via MCMC

In this section, we will show how simulated annealing Gibbs sampling enables us to get close to the performance of the impractical exhaustive search coding algorithm described in the previous section. Throughout this section we fix the slope $s \leq 0$.

Associate with each reconstruction sequence y^n the *energy*

$$\begin{aligned}\mathcal{E}(y^n) &\triangleq n\left[H_k(y^n) - s \cdot d_n(x^n, y^n)\right]\\ &= \sum_{u^k} \mathcal{H}\left(\mathbf{m}_k(y^n, u^k)\right) \mathbf{1}^T \mathbf{m}_k(y^n, u^k) - s \cdot \sum_{i=1}^{n} d(x_i, y_i).\end{aligned}$$

The *Boltzmann distribution* can now be defined as the pmf on $\mathcal{Y}^n$ given by

$$p_\beta(y^n) = \frac{1}{Z_\beta}\exp\{-\beta\mathcal{E}(y^n)\}, \tag{11}$$

where Z_β is the normalization constant (partition function). Note that, though this dependence is suppressed in the notation for simplicity, $\mathcal{E}(y^n)$, and therefore also p_β and Z_β depend on x^n and s, which are fixed until further notice. When β is large and $Y^n \sim p_\beta$, then with high probability

$$H_k(Y^n) - s \cdot d_n(x^n, Y^n) \approx \min_{y^n}\left[H_k(y^n) - s \cdot d_n(x^n, y^n)\right]. \tag{12}$$

Thus, using a sample from the Boltzmann distribution p_β, for large β, as the reconstruction sequence, would yield performance close to that of an exhaustive search scheme that would use the achiever of the minimum in (12). Unfortunately, it is hard to sample from the Boltzmann distribution directly. We can, however, get approximate samples via MCMC, as we describe next.

As mentioned earlier, Gibbs sampler [17] is useful in cases where one is interested in sampling from a probability distribution which is hard to compute, but the conditional distribution of each variable given the rest of variables is accessible. In our case, the conditional probability under p_β of Y_i given the other variables $Y^{n\setminus i} \triangleq \{Y_n : n \neq i\}$ can be expressed as

$$\begin{aligned}p_\beta(Y_i = a | Y^{n\setminus i} = y^{n\setminus i}) &= \frac{p_\beta(Y_i = a, Y^{n\setminus i} = y^{n\setminus i})}{\sum_b p_\beta(Y_i = b, Y^{n\setminus i} = y^{n\setminus i})}, &(13)\\ &= \frac{\exp\{-\beta\mathcal{E}(y^{i-1}ay^n_{i+1})\}}{\sum_b \exp\{-\beta\mathcal{E}(y^{i-1}by^n_{i+1})\}}, &(14)\\ &= \frac{\exp\{-\beta n\left[H_k(y^{i-1}ay^n_{i+1}) - s \cdot d_n(x^n, y^{i-1}ay^n_{i+1})\right]\}}{\sum_b \exp\{-\beta n\left[H_k(y^{i-1}by^n_{i+1}) - s \cdot d_n(x^n, y^{i-1}by^n_{i+1})\right]\}}, &(15)\\ &= \frac{1}{\sum_b \exp\{-\beta\left[n\Delta H_k(y^{i-1}by^n_{i+1}, a) - s \cdot \Delta d(b, a, x_i)\right]\}}, &(16)\end{aligned}$$

where $\Delta H_k(y^{i-1}by_{i+1}^n, a)$ and $\Delta d(y^{i-1}by_{i+1}^n, a, x_i)$ are defined as

$$\Delta H_k(y^{i-1}by_{i+1}^n, a) \triangleq H_k(y^{i-1}by_{i+1}^n) - H_k(y^{i-1}ay_{i+1}^n), \tag{17}$$

and

$$\Delta d(b, a, x_i) \triangleq d(b, x_i) - d(a, x_i), \tag{18}$$

respectively. Evidently, $p_\beta(Y_i = y_i | Y^{n \setminus i} = y^{n \setminus i})$ depends on y^n only through $\{H_k(y^{i-1}by_{i+1}^n) - H_k((y^{i-1}ay_{i+1}^n))\}_{a,b\in\mathcal{Y}}$ and $\{d(x_i, a)\}_{a\in\mathcal{Y}}$. In turn, $\{H_k(y^{i-1}by_{i+1}^n) - H_k(y^{i-1}ay_{i+1}^n)\}_{a,b\in\mathcal{Y}}$ depends on y^n only through $\{\mathbf{m}_k(y^{i-1}yy_{i+1}^n, \cdot)\}_{y\in\mathcal{Y}}$.

Note that, given $\mathbf{m}_k(y^n, \cdot)$, the number of operations required to obtain $\mathbf{m}_k(y^{i-1}yy_{i+1}^n, \cdot)$, for any $y \in \mathcal{Y}$ is linear in k, since the number of contexts whose counts are affected by a change of one component in y^n is no larger than $2k+2$. I.e., letting $\mathcal{S}_i(y^n, y)$ denote the set of contexts whose counts are affected when the ith component of y^n is flipped from y_i to y, we have $|\mathcal{S}_i(y^n, y)| \leq 2k+2$. Further, since

$$\begin{aligned} n[H_k(y^{i-1}by_{i+1}^n) - H_k(y^{i-1}ay_{i+1}^n)] = \sum_{u^k \in \mathcal{S}_i(y^{i-1}by_{i+1},a)} &\left[\mathbf{1}^T\mathbf{m}_k(y^{i-1}by_{i+1}^n, u^k)\mathcal{H}\left(\mathbf{m}_k(y^{i-1}by_{i+1}^n, u^k)\right)\right. \\ &\left. -\mathbf{1}^T\mathbf{m}_k(y^{i-1}ay_{i+1}^n, u^k)\mathcal{H}\left(\mathbf{m}_k(y^{i-1}ay_{i+1}^n, u^k)\right)\right], \end{aligned} \tag{19}$$

it follows that, given $\mathbf{m}_k(y^{i-1}by_{i+1}^n, \cdot)$ and $H_k(y^{i-1}by_{i+1}^n)$, the number of operations required to compute $\mathbf{m}_k(y^{i-1}ay_{i+1}^n, \cdot)$ and $H_k(y^{i-1}ay_{i+1}^n)$ is linear in k (and independent of n).

Now consider the following algorithm (Algorithm 1 below) based on the Gibbs sampling method for sampling from p_β, and let $\hat{X}_{s,r}^n(X^n)$ denote its (random) outcome when taking $k = k_n$ and $\beta = \{\beta_t\}_t$ to be deterministic sequences satisfying $k_n = o(\log n)$ and $\beta_t = \frac{1}{T_0^{(n)}}\log(\lfloor \frac{t}{n} \rfloor + 1)$, for some $T_0^{(n)} > n\Delta$, where

$$\Delta = \max_i \max_{\left\{\substack{u^{i-1} \in \mathcal{Y}^{i-1}, \\ u_{i+1}^n \in \mathcal{Y}^{n-i}, \\ a, b \in \mathcal{Y}}\right.} |\mathcal{E}(u^{i-1}au_{i+1}^n) - \mathcal{E}(u^{i-1}bu_{i+1}^n)|, \tag{20}$$

applied to the source sequence X^n as input.[1] By the previous discussion, the computational complexity of the algorithm at each iteration is independent of n and linear in k.

Theorem 2: Let $\mathbf{X}$ be a stationary and ergodic source. Then

$$\lim_{n\to\infty}\lim_{r\to\infty} \mathbb{E}\left[\frac{1}{n}\ell_{\mathsf{LZ}}\left(\hat{X}_{s,r}^n(X^n)\right) - s \cdot d_n(X^n, \hat{X}^n)\right] = \min_{D\geq 0}\left[R(\mathbf{X}, D) - s \cdot D\right]. \tag{21}$$

Proof: The proof is presented in Appendix B. ■

VI. Sliding-window rate-distortion coding via MCMC

The classical approach to lossy source coding is block coding initiated by Shannon [1]. In this method, each possible source block of length n is mapped into a reconstruction block of the same length. One of the disadvantages

[1] Here and throughout it is implicit that the randomness used in the algorithms is independent of the source, and the randomization variables used at each drawing are independent of each other.

Algorithm 1 Generating the reconstruction sequence

```
Input: $x^n$, $k$, $s$, $\{\beta_t\}_t$, $r$
Output: a reconstruction sequence $\hat{x}^n$
$y^n \leftarrow x^n$
for $t = 1$ to $r$ do
    Draw an integer $i \in \{1, \dots, n\}$ uniformly at random
    For each $y \in \mathcal{Y}$ compute $p_{\beta_t}(Y_i = y | Y^{n\backslash i} = y^{n\backslash i})$ given in (16)
    Update $y^n$ by replacing its $i^{\text{th}}$ component $y_i$ by $y$ drawn from the pmf $p_{\beta_t}(Y_i = \cdot | Y^{n\backslash i} = y^{n\backslash i})$
    Update $\mathbf{m}_k(y^n, \cdot)$ and $H_k(y^n)$
end for
$\hat{x}^n \leftarrow y^n$
```

of this method is that applying a block code to a stationary process converts it into a non-stationary reconstruction process. Another approach to the rate-distortion coding problem is sliding-block (SB) coding introduced by R.M. Gray in 1975 [7]. In this method, a fixed SB map of a certain order $2k_f + 1$ slides over the source sequence and generates the reconstruction sequence which has lower entropy rate compared to the original process. The advantage of this method with respect to the block coding technique is that while the achievable rate-distortion regions of the two methods provably coincide, the stationarity of the source is preserved by a SB code [7]. Although SB codes seem to be a good alternative to block codes, there has been very little progress in constructing good such codes since their introduction in 1975, and there is no known practical method for finding practical SB codes up to date. In this section we show how our MCMC-based approach can be applied to finding good SB codes of a certain order $2k_f + 1$.

A SB code of window length $2k_f + 1$, is a function $f : \mathcal{X}^{2k_f+1} \rightarrow \mathcal{Y}$ which is applied to the source process $\{X_n\}$ to construct the reconstruction block as follows

$$\hat{X}_i = f(X_{i-k_f}^{i+k_f}). \tag{22}$$

The total number of $2k_f + 1$-tuples taking values in $\mathcal{X}$ is

$$K_f = |\mathcal{X}|^{2k_f+1}.$$

Therefore, for specifying a SB code of window length $2k_f + 1$, there are K_f values to be determined, and f can be represented as a vector $f^{K_f} = [f_{K_f-1}, \dots, f_1, f_0]$ where $f_i \in \mathcal{Y}$ is the output of function f to the input vector **b** equal to the expansion of i in $2k_f + 1$ symbols modulo $|\mathcal{X}|$, i.e. $i = \sum\limits_{j=0}^{2k_f} b_j |\mathcal{X}|^j$.

For coding a source output sequence x^n by a SB code of order $2k_f + 1$, among $|\mathcal{Y}|^{|\mathcal{X}|^{2k_f+1}}$ possible choices, similar to the exhaustive search algorithm described in Section IV, here we look for the one that minimizes the energy function assigned to each possible SB code as

$$\mathcal{E}(f^{K_f}) \triangleq n \left[H_k(y^n) - s \cdot d_n(x^n, y^n) \right], \tag{23}$$

where $y^n = y^n[x^n, f^{K_f}]$ is defined by $y_i = f(x_{i-k_f}^{i+k_f})$. Like before, we consider a cyclic rotation as $x_i = x_{i+n}$, for any $i \in \mathbb{N}$. Again, we resort to simulated annealing Gibbs sampling method in order to find the minimizer of (23). Unlike in (11), instead of the space of possible reconstruction blocks, here we define Boltzmann distribution over the space of possible SB codes. Each SB code is represented by a unique vector f^{K_f}, and $p_\beta(f^{K_f}) \propto \exp\left(-\beta \mathcal{E}(y^n)\right)$, where $y^n = y^n[x^n, f^{K_f}]$. The conditional probabilities required at each step of the Gibbs sampler can be written as

$$p_\beta(f_i = \theta | f^{K_f \setminus i}) = \frac{p_\beta(f^{i-1}\theta f_{i+1}^{K_f})}{\sum\limits_{\vartheta} p_\beta(f^{i-1}\vartheta f_{i+1}^{K_f})}, \tag{24}$$

$$= \frac{1}{\sum\limits_{\vartheta} \exp\left(-\beta(\mathcal{E}(f^{i-1}\vartheta f_{i+1}^{K_f}) - \mathcal{E}(f^{i-1}\theta f_{i+1}^{K_f}))\right)}. \tag{25}$$

Therefore, for computing the conditional probabilities we need to find out by how much changing one entry of f^{K_f} affects the energy function. Compared to the previous section, finding this difference in this case is more convoluted and should be handled with more deliberation. To achieve this goal, we first categorize different positions in x^n into $|\mathcal{X}|^{2k_f+1}$ different types and construct the s^n vector such that the label of x_i, α_i, is defined to be

$$\alpha_i \triangleq \sum_{j=-k_f}^{k_f} x_{n+j} |\mathcal{X}|^{k_f+j}. \tag{26}$$

In other words, the label of each position is defined to be the symmetric context of length $2k_f + 1$ embracing it, i.e. $x_{i-k_f}^{i+k_f}$. Using this definition, applying a SB code f^{K_f} to a sequence x^n can alternatively be expressed as constructing a sequence y^n where

$$y_i = f_{\alpha_i}. \tag{27}$$

From this representation, changing f_i from θ to ϑ while leaving the other elements of f^{K_f} unchanged only affects the positions of the y^n sequence that correspond to the label i in the s^n sequence, and we can write the difference between energy functions appearing in (25) as

$$\mathcal{E}(f^{i-1}\vartheta f_{i+1}^{K_f}) - \mathcal{E}(f^{i-1}\theta f_{i+1}^{K_f}) = n\left[H_k(\mathbf{m}_k(y^n,\cdot)) - H_k(\mathbf{m}_k(\hat{y}^n,\cdot))\right] - s \sum_{j:\alpha_j = i} (d(x_j,\vartheta) - d(x_j,\theta)), \tag{28}$$

where y^n and $\hat{y}^n$ represent the results of applying $f^{i-1}\vartheta f_{i+1}^{K_f}$ and $f^{i-1}\theta f_{i+1}^{K_f}$ to x^n respectively, and as noted before the two vectors differ only at the positions $\{j : \alpha_j = i\}$. Flipping each position in y^n sequence in turn affects at most $2(k+1)$ columns of the count matrix $\mathbf{m}_k(y^n,\cdot)$. Here at each pass of the Gibbs sampler a number of positions in the y^n sequence are flipped simultaneously. Alg. 2 describes how we can keep track of all these changes and update the count matrix. After that in analogy to Alg. 1, Alg. 3 runs the Gibbs sampling method to find the best SB code of order $2k_f + 1$, and at each iteration it employs Alg. 2.

Let $f_{\beta,s,r}^{K_f^{(n)}}$ denote the output of Alg. 3 to input vector x^n at slope s after r iterations, and annealing process β. $K_f^{(n)} = 2^{2k_f^{(n)}+1}$ denotes the length of the vector f representing the SB code. The following theorem whose proof is given in the Appendix 3 states that Alg. 3 is asymptotically optimal for any stationary ergodic source. I.e. coding a source sequence by applying the SB code $f_{\beta,s,r}^{K_f^{(n)}}$ to the source sequence, and then describing the output to the

Algorithm 2 Updating the count matrix of $y^n = f(x^n)$, when f_i changes from ϑ to θ

```
Input: $x^n$, $k_f$, $k$, $\mathbf{m}_k(y^n,\cdot)$, $i$, $\vartheta$, $\theta$
Output: $\mathbf{m}_k(\hat{y}^n,\cdot)$
$a^n \leftarrow \mathbf{0}$
$\hat{y}^n \leftarrow y^n$
for $j = 1$ to $n$ do
    if $\alpha_j = i$ then
        $\hat{y}_j \leftarrow \theta$
    end if
end for
$\mathbf{m}_k(\hat{y}^n,\cdot) \leftarrow \mathbf{m}_k(y^n,\cdot))$
for $j = k_f + 1$ to $n - k_f$ do
    if $j = f_i$ then
        $a_j^{j+k} \leftarrow \mathbf{1}$
    end if
end for
for $j = k + 1$ to $n - k$ do
    if $a_j = 1$ then
        $\mathbf{m}_k(\hat{y}^n, y_{j-k}^{j-1})[y_j] \leftarrow \mathbf{m}_k(\hat{y}^n, y_{j-k}^{j-1})[y_j] - 1$
        $\mathbf{m}_k(\hat{y}^n, \hat{y}_{j-k}^{j-1})[\hat{y}_j] \leftarrow \mathbf{m}_k(\hat{y}^n, \hat{y}_{j-k}^{j-1})[\hat{y}_j] + 1$
    end if
end for
```

decoder using Lempel-Ziv algorithm, asymptotically, as the number of iterations and window length k_f grow to infinity, achieves the rate-distortion curve.

Theorem 3: Given a sequence $\{k_f^{(n)}\}$ such that $k_f^{(n)} \to \infty$, schedule $\beta_t^{(n)} = \frac{1}{T_0^{(n)}} \log(\lfloor \frac{t}{K_f^{(n)}} \rfloor + 1)$ for some $T_0^{(n)} > K_f \Delta$, where

$$\Delta = \max_i \max_{\left\{\begin{array}{c} f^{i-1} \in \mathcal{Y}^{i-1}, \\ f_{i+1}^{n} \in \mathcal{Y}^{K_f - i}, \\ \vartheta, \theta \in \mathcal{Y} \end{array}\right.} |\mathcal{E}(f^{i-1}\vartheta f_{i+1}^{K_f}) - \mathcal{E}(f^{i-1} b f_{i+1}^{K_f})|, \tag{29}$$

and $k = o(\log(n))$. Then, for any stationary and ergodic source $\mathbf{X}$, we have

$$\lim_{n\to\infty} \lim_{r\to\infty} \mathbb{E}\left[\frac{1}{n}\ell_{\mathsf{LZ}}\left(\hat{X}^n\right) - s \cdot d_n(X^n, \hat{X}^n)\right] = \min_{D \geq 0}\left[R(\mathbf{X}, D) - s \cdot D\right], \tag{30}$$

where $\hat{X}^n$ is the result of applying SB code $f_{\beta,s,r}^{K_f}$ to X^n.

Proof: The proof is presented in Appendix C. ■

Algorithm 3 Universal SB lossy coder based on simulated annealing Gibbs sampler

```
Input: $x^n$, $k_f$, $k$, $s$, $\beta$, $r$
Output: $f^{K_f}$
for $t = 1$ to $r$ do
    Draw an integer $i \in \{1, \ldots, K_f\}$ uniformly at random
    For each $a \in \mathcal{Y}$ compute $p_{\beta_t}(f_i = \theta | f^{K_f \setminus i})$ using Algorithm 2, equations (25), and (28)
    Update $f^{K_f}$ by replacing its $i$-th component $f_i$ by $\theta$ drawn from the pmf computed in the previous step
end for
```

Note that in Alg. 3, for a fixed k_f, the SB code is a vector of length $|\mathcal{X}|^{2k_f+1}$. Hence, the size of the search space is $|\mathcal{Y}|^{K_f}$ which is independent of n. Moreover, the transition probabilities of the SA as defined by (25) depend on the differences of the form presented in (28), which, for a stationary ergodic source and fixed k_f, if n is large enough, linearly scales with n. I.e. for a given f^{i-1}, $f_{i+1}^{K_f}$, ϑ and θ,

$$\lim_{n\to\infty} \frac{1}{n}[\mathcal{E}(f^{i-1}\vartheta f_{i+1}^{K_f}) - \mathcal{E}(f^{i-1}\theta f_{i+1}^{K_f})] = q \qquad \text{a.s.}, \tag{31}$$

where $q \in [0,1]$ some fixed value depending only on the source distribution. This is an immediate consequence of the ergodicity of the source plus the fact that SB coding of a stationary ergodic process results in another process which is jointly stationary with the initial process and is also ergodic. On the other hand, similar reasoning proves that Δ defined in (29) scales linearly by n. Therefore, overall, combining these two observations, for large values of n and fixed k_f, the transition probabilities of the nonhomogeneous MC defined by the SA algorithm incorporated in Alg. 3 are independent of n. This does not mean that the convergence rate of the algorithm is independent of n, because for achieving the rate-distortion function one needs to increase k_f and n simultaneously to infinity.

VII. Simulation results

We dedicate this section to the presentation of some initial experimental results obtained by applying the schemes presented in the previous sections on simulated and real data. The Sub-section VII-A demonstrates the performance of Alg. 1 on simulated 1D and real 2D data. Some results on the application Alg. 3 on simulated 1D data is shown in Sub-section VII-B.

A. Block coding

In this sub-section, some of the simulation results obtained from applying Alg. 1 of Section V to real and simulated data are presented. The algorithm is easy to apply, as is, to both 1D and 2D data . As a first example, consider a Bernoulli(p) i.i.d source with $p = 0.1$. Fig. 1 compares the performance of Alg. 1 against the optimal rate-distortion tradeoff given by $R(D) = h(p) - h(D)$, where $h(\alpha) = -\alpha\log(\alpha) - (1-\alpha)\log(1-\alpha)$, for a source sequence of length $n = 5e4$. Here, in order to get different points, s has been linearly increased from $s = -5$ to $s = -3$. To illustrate the encoding process, Fig. 2 depicts the evolutions of $H_k(\hat{x}^n)$, $d_n(x^n, \hat{x}^n)$, and

$\mathcal{E}(\hat{x}^n) = H_k(\hat{x}^n) - s \cdot d_n(x^n, \hat{x}^n)$ during the encoding process of a source block of length $n = 5e4$ at $s = -2.5$, $k = 10$, and $\beta(t) = 1 + t$.

As another example, Fig. 6 compares the performance of Alg. 1 when applied to a binary symmetric Markov source (BSMS) with transition probability $q = 0.2$ against the Shannon lower bound (SLB) which sates that for a BSMS

$$R(D) \geq R_{\text{SLB}}(D) \triangleq h(p) - h(D). \tag{32}$$

There is no known explicit characterization of the rate-distortion tradeoff for a BSMS except for a low distortion region. It has been proven that for $D < D_c$, where

$$D_c = \frac{1}{2}\left(1 - \sqrt{1-(p/q)^2}\right), \tag{33}$$

the SLB holds with equality, and for $D > D_c$, we have strict inequality, i.e. $R(D) > R_{\text{SLB}}$ [5]. In our case $D_c = 0.0159$. For distortions beyond D_c, a lower bound and an upper bound on the rate-distortion function, derived based on the results presented in [36], are also shown for comparison. The parameters here are: $n = 5e4$, $k = 8$ and $\beta_t = n|s/3|t^{1.5}$.

Finally, consider applying the algorithm to a $n \times n$ binary image, where $n = 252$. Fig. 7.1 shows the original image, while Fig. 7.2 shows the coded version after $r = 50n^2$ iterations. The parameters are: $s = -0.1$, and $\beta(t) = 0.1 \log(t)$. The empirical conditional entropy of the image has decreased from $H_k = 0.1025$ to $H_k = 0.0600$ in the reconstruction image, while an average distortion of $D = 0.0337$ per pixel is introduced. Fig. 1 shows the pixels that form the 2-D ‘history’ (causal neighborhood) of each pixel in our experiments, i.e. the pixels whose different configurations form the columns of the count vector $\mathbf{m}_k(y^{m\times n}, \cdot)$.

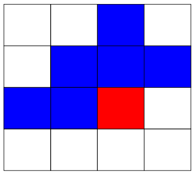


Fig. 1.

In Fig. 1, the red square depicts the location of the current pixel, and the blue squares denote its 6th order context.

Comparing the required space for storing the original image as a PNG file with the amount required for the coded image reveals that in fact the algorithm not only has reduced the conditional empirical entropy of the image by 41.5%, but also has cut the size of the file by around 39%.

B. Sliding-block coding

Consider applying Alg. 3 of Section VI to the output of a BSMS with $q = 0.2$. Fig. 8 shows the algorithm output along with Shannon lower bound and lower/upper bounds on $R(D)$ from [36]. Here the parameters are: $n = 5e4$, $k = 8$, SB window length of $k_f = 11$ and $\beta_t = K_f|s|\log(t+1)$.

In all of the presented simulation results, it is the empirical conditional entropy of the final reconstruction block that we are comparing to the rate-distortion curve. It should be noted that, though this difference vanishes as the block size grows, for finite values of n there would be an extra (model) cost for losslessly describing the reconstruction block to the decoder.

VIII. Application: Optimal denoising via MCMC-based lossy coding

Consider the problem of denoising a stationary ergodic source $\mathbf{X}$ with unknown distribution corrupted by additive white noise $\mathbf{V}$. Compression-based denoising algorithms have been proposed before by a number of researchers, cf. [28], [29], [30] and references therein. The idea of using a universal lossy compressor for denoising was proposed in [29], and then refined in [30] to result in a universal denoising algorithm. In this section, we show how our new MCMC-based lossy encoder enables the denoising algorithm proposed in [30] to lead to an implementable universal denoiser.

In [30], it is shown how a universally optimal lossy coder tuned to the right distortion measure and distortion level combined with some simple “post-processing” results in a universally optimal denoiser. In what follows we first briefly go over this compression-based denoiser described in [30], and then show how our lossy coder can be embedded in for performing the lossy compression part.

Throughout this section we assume that the source, noise, and reconstruction alphabets are $\mathcal{M}$-ary alphabet $\mathcal{A} = \{0, 1, \dots, M-1\}$, and the noise is additive modulo-M and $P_V(a) > 0$ for any $a \in \mathcal{A}$, i.e. $Z_i = X_i + V_i$.

As mentioned earlier, in the denoising scheme outlined in [30], first the denoiser lossily compresses’ the noisy signal appropriately, and partly removes the additive noise. Consider a sequence of *good* lossy coders characterized by encoder/decoder pairs $(\mathrm{E}_n, \mathrm{D}_n)$ of block length n working at distortion level $H(V)$ under the difference distortion measure defined as

$$\rho(x, y) = \log \frac{1}{P_V(x-y)}. \tag{34}$$

By *good*, it is meant that for any stationary ergodic source $\mathbf{X}$, as n grows, the rate distortion performance of the sequence of codes converges to a point on the rate-distortion curve. The next step is a simple “post-processing” as follows. For a fixed m, define the following count vector over the noisy signal Z^n and its quantized version $Y^n = \mathrm{D}_n(\mathrm{E}_n(Z^n))$,

$$\begin{aligned}&\hat{Q}^{2m+1}[Z^n, Y^n](z^{2m+1}, y) \triangleq \\ &\qquad \frac{1}{n}|\{1 \le i \le n : (Z_{i-k}^{i+k}, Y_i) = (z^{2m+1}, y)\}|.\end{aligned} \tag{35}$$

After constructing these count vectors, the denoiser output is generated through the “post-processing” or “derandomization” process as follows

$$\hat{X}_i = \arg\min_{\hat{x}\in\mathcal{A}} \sum_{y\in\mathcal{A}} \hat{Q}^{2m+1}[Z^n, Y^n](z^{2m+1}, y) d(\hat{x}, y), \tag{36}$$

where $d(\cdot, \cdot)$ is the original loss function under which the performance of the denoiser is to be measured. The described denoiser is shown to be universally optimal [30], and the basic theoretical justification of this is that the rate-distortion function of the noisy signal $\mathbf{Z}$ under the difference distortion measure satisfies the Shannon lower bound with equality, and it is proved in [30] that for such sources [2] for a fixed k, the k-th order empirical joint distribution between the source and reconstructed blocks defined as

$$\hat{Q}^k[X^n,Y^n](x^k,y^k) \triangleq \frac{1}{n}|\{1 \leq i \leq n : (X_i^{i+k-1}, Y_i^{i+k-1}) = (x^k,y^k)\}|, \tag{37}$$

resulting from a sequence of *good* codes converge to P_{X^k,Y^k} in distribution, i.e. $\hat{Q}^k[X^n,Y^n] \overset{d}{\Rightarrow} P_{X^k,Y^k}$, where P_{X^k,Y^k} is the unique joint distribution that achieves the k-th order rate-distortion function of the source. In the case of quantizing the noisy signal under the distortion measure defined in (34), at level $H(V)$, P_{X^k,Y^k} is the k-th order joint distribution between the source and noisy signal. Hence, the count vector $\hat{Q}^{2m+1}[Z^n,Y^n](z^{2m+1},y)$ defined in (35) asymptotically converges to $P_{X_i|Z^n}$ which is what the optimal denoiser would base its decision on. After estimating $P_{X_i|Z^n}$, the post-processing step is just making the optimal Bayesian decision at each position.

The main ingredient of the described denoiser is a universal lossy compressor. Note that the MCMC-based lossy compressor described in Section V is applicable to any distortion measure. The main problem is choosing the parameter s corresponding to the distortion level of interest. To find the right slope, we run the quantization MCMC-based part of the algorithm independently from two different initial points s_1 and s_2. After convergence of the two runs we compute the average distortion between the noisy signal and its quantized versions. Then assuming a linear approximation, we find the value of s that would have resulted in the desired distortion, and then run the algorithm again from this starting point, and again computed the average distortion, and then find a better estimate of s from the observations so far. After a few repetitions of this process, we have a reasonable estimate of the desired s. Note that for finding s it is not necessary to work with the whole noisy signal, and one can consider only a long enough section of data first, and find s from it, and then run the MCMC-based denoising algorithm on the whole noisy signal with the estimated parameter s. The outlined method for finding s is similar to what is done in [26] for finding appropriate Lagrange multiplier.

A. Experiments

In this section we compare the performance of the proposed denoising algorithm against discrete universal denoiser, DUDE [31], introduced in [27]. DUDE is a practical universal algorithm that asymptotically achieves the performance attainable by the best n-block denoiser for any stationary ergodic source. The setting of operation of DUDE is more general than what is described in the previous section, and in fact in DUDE the additive white noise can be replaced by any known discrete memoryless channel.

As a first example consider a BSMS with transition probability p. Fig. 9 compares the performance of DUDE with the described algorithm. The slope s is chosen such that the expected distortion between the noisy image and

[2] In fact it is shown in [30] that this is true for a large class of sources including i.i.d sources and those satisfying the Shannon lower bound with equality.

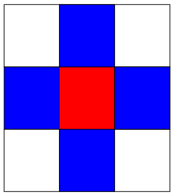


Fig. 2.

its quantized version using Alg. 1 is close to the channel probability of error which is $\delta = 0.1$ in our case. Here we picked $s = -0.9$ for all values of p and did not tune it specifically each time. Though, it can be observed that, even without optimizing the MCMC parameters, the two algorithms performances are very close, and for small values of p the new algorithm outperforms DUDE.

As another example consider denoising a binary image. The channel a is DMC with error probability of 0.04. Fig. 10.2 shows the noisy image. Fig. 10.3 shows the reconstructed image generated by DUDE and 10.4 depicts the reconstructed image using the described algorithm. In this experiment the DUDE context structure is set as Fig. 2.

The 2-D MCMC coder employs the same context as the one used in the example of Section VII-A shown in Fig. 1, and the derandomization block is chosen as Fig. 3.

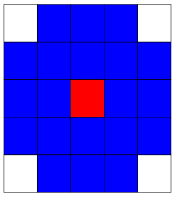


Fig. 3.

Discussion: The new proposed approach which is based on MCMC coding plus de-randomization is an alternative not only to the DUDE, but also to MCMC-based denoising schemes that have been based on and inspired by the Geman brothers' work [17]. While algorithmically, this approach has much of the flavor of previous MCMC-based denoising approaches, ours has the merit of leading to a universal scheme, whereas the previous MCMC-based schemes guarantee, at best, convergence to something which is good according to the posterior distribution of the original given the noisy data, but as would be induced by the rather arbitrary prior model placed on the data. It is clear that here no assumption about the distribution/model of the original data is made.

IX. CONCLUSIONS AND FUTURE WORK

In this paper, a new implementable universal lossy source coding algorithm based on simulated annealing Gibbs sampling is proposed, and it is shown that it is capable of getting arbitrarily closely to the rate-distortion curve of any stationary ergodic source. For coding a source sequence x^n, the algorithm starts from some initial reconstruction block, and updates one of its coordinates at each iteration. The algorithm can be viewed as a process of systematically introducing ‘noise’ into the original source block, but in a biased direction that results in a decrease of its description complexity. We further developed the application of this new method to universal denoising.

In practice, the proposed algorithms 1 and 3, in their present form, are only applicable to the cases where the size of the reconstruction alphabet, $|\mathcal{Y}|$, is small. The reason is twofold: first, for larger alphabet sizes the contexts will be too sparse to give a true estimate of the empirical entropy of the reconstruction block, even for small values of k. Second, the size of the count matrix $\mathbf{m}_k$ grows exponentially with $|\mathcal{Y}|$ which makes storing it for large values of $|\mathcal{Y}|$ impractical. Despite this fact, there are practical applications where this constraint is satisfied. An example is lossy compression of binary images, like the one presented in Section VII. Another application for lossy compression of binary data is shown in [37] where one needs to compress a stream of 0 and 1 bits with some distortion.

The convergence rate of the new algorithms and the effect of different parameters on it is a topic for further study. As an example, one might wonder how the convergence rate of the algorithm is affected by choosing an initial point other than the source output block itself. Although our theoretical results on universal asymptotic optimality remain intact for any initial starting point, in practice the choice of the starting point might significantly impact the number of iterations required.

Finally, note that in the non-universal setup, where the optimal achievable rate-distortion tradeoff is known in advance, this extra information can be used as a stopping criterion for the algorithm. For example, we can set it to stop after reaching optimum performance to within some fixed distance.

APPENDIX A: PROOF OF THEOREM 1

First we show that for any $n \in \mathbb{N}$,

$$\mathbb{E}\left[\frac{1}{n}\ell_{\mathsf{LZ}}(\hat{X}^n) - s\cdot d(X^n,\hat{X}^n)\right] \geq \min_{D\geq 0}\left[R(\mathbf{X},D) - s\cdot D\right]. \tag{A-1}$$

For any fixed n, the expected average loss of our scheme would be $D_0^{(n)} \triangleq Ed(X^n,\hat{X}^n)$. For this expected average distortion, the rate of our code can not be less than $R(\mathbf{X},D_0^{(n)})$ which is the minimum required rate for achieving distortion $D_0^{(n)}$. Therefore,

$$\begin{aligned}\mathbb{E}\left[\frac{1}{n}\ell_{\mathsf{LZ}}(\hat{X}^n)\right] - sD_0^{(n)} &\geq R(\mathbf{X},D_0^{(n)}) - s\cdot D_0^{(n)}\\ &\geq \min_{D\geq 0}\left[R(\mathbf{X},D) - s\cdot D\right].\end{aligned} \tag{A-2}$$

Now letting n go to infinity we get

$$\begin{aligned}&\liminf_{n\to\infty}\mathbb{E}\left[\frac{1}{n}\ell_{\mathsf{LZ}}(\hat{X}^n) - s\cdot d(X^n,\hat{X}^n)\right]\\ &\qquad\geq \min_{D\geq 0}\left[R(\mathbf{X},D) - s\cdot D\right].\end{aligned} \tag{A-3}$$

On the other hand, in order to obtain the upper bound, we first split the cost function into two terms as follows

$$\mathbb{E}\left[\frac{1}{n}\ell_{\mathsf{LZ}}(\hat{X}^n) - s\cdot d(X^n,\hat{X}^n)\right], \tag{A-4}$$

$$= \mathbb{E}\left[\frac{1}{n}\ell_{\mathsf{LZ}}(\hat{X}^n) - H_{k_n}(\hat{X}^n) + H_{k_n}(\hat{X}^n) - s\cdot d(X^n,\hat{X}^n)\right] \tag{A-5}$$

$$= \mathbb{E}\left[\frac{1}{n}\ell_{\mathsf{LZ}}(\hat{X}^n) - H_{k_n}(\hat{X}^n)\right] + \mathbb{E}\left[H_{k_n}(\hat{X}^n) - s\cdot d(X^n,\hat{X}^n)\right]. \tag{A-6}$$

From [10], for $k_n = o(\log n)$ and any given $\epsilon > 0$, there exists $N_\epsilon \in \mathbb{N}$ such that for any individual infinite-length sequence $\hat{\mathbf{x}} = (\hat{x}_1, \hat{x}_2, \ldots)$ and any $n \geq N_\epsilon$

$$\left[\frac{1}{n}\ell_{\mathsf{LZ}}(\hat{x}^n) - H_{k_n}(\hat{x}^n)\right] \leq \epsilon. \tag{A-7}$$

Therefore, for $n \geq N_\epsilon$

$$\mathbb{E}\left[\frac{1}{n}\ell_{\mathsf{LZ}}(\hat{X}^n) - H_{k_n}(\hat{X}^n)\right] \leq \epsilon. \tag{A-8}$$

Consider an arbitrary point $(R(\mathbf{X},D),D)$ on the rate-distortion curve of source $\mathbf{X}$. Then we know that for any $\delta > 0$ there exists a process $\tilde{\mathbf{X}}$ such that $(\mathbf{X},\tilde{\mathbf{X}})$ are jointly stationary and ergodic, and moreover

1) $H(\tilde{\mathbf{X}}) \leq R(\mathbf{X},D)$,
2) $\mathbb{E}d(X_0,\tilde{X}_0) \leq D+\delta$,

where $H(\tilde{\mathbf{X}}) = H(\tilde{X}_0|\tilde{X}_{-\infty}^{-1})$ is the entropy rate of process $\tilde{\mathbf{X}}$ [32]. Now since for each source block X^n, the reconstruction block $\hat{X}^n$ is chosen to minimize $H_k(\hat{X}^n) - s\cdot d(X^n,\hat{X}^n)$, we have

$$H_{k_n}(\hat{X}^n) - s\cdot d(X^n,\hat{X}^n) \leq H_{k_n}(\tilde{X}^n) - s\cdot d(X^n,\tilde{X}^n). \tag{A-9}$$

For a fixed k, from the definition of the k-th order entropy, we have

$$H_k(\tilde{X}^n) = \frac{1}{n}\sum_{u^k\in\hat{\mathcal{X}}^n} \mathcal{H}\left(\mathbf{m}_k(\tilde{X}^n,u^k)\right)\mathbf{1}^T\mathbf{m}_k(\tilde{X}^n,u^k), \tag{A-10}$$

where

$$\frac{1}{n}\mathbf{m}_k(\tilde{X}^n,u^k)[y] = \frac{1}{n}\sum_{i=1}^{n}\mathbb{1}_{\tilde{X}_{i-k}^{i}=u^k y} \tag{A-11}$$

$$\overset{n\to\infty}{\longrightarrow} Pr\left(\tilde{X}_{-k}^{0} = u^k y\right), \quad \text{w.p.1.} \tag{A-12}$$

Therefore, combining (A-10) and (A-12), as n goes to infinity, $H_k(\tilde{X}^n)$ converges to $H(\tilde{\mathbf{X}}_0|\tilde{\mathbf{X}}_{-k}^{-1})$ with probability one. It follows from the monotonicity of $H_k(\hat{x}^n)$ in k, (A-9), and the convergence we just established that for any $\hat{x}^n$ and any k,

$$H_{k_n}(\hat{X}^n) - s\cdot d(X^n,\hat{X}^n) \leq H(\tilde{X}_0|\tilde{X}_{-k}^{-1}) + \epsilon - s\cdot d(X^n,\tilde{X}^n), \quad \text{eventually a.s.} \tag{A-13}$$

On the other hand

$$d(\tilde{X}^n,X^n) = \frac{1}{n}\sum_{i=1}^{n} d(X_i,\tilde{X}_i) \overset{n\to\infty}{\longrightarrow} Ed(\tilde{X}_0,X_0) \leq D+\delta. \tag{A-14}$$

Combining (A-7) and (A-13) gives

$$\limsup_{n\to\infty} \mathbb{E}\left[\frac{1}{n}\ell_{\mathsf{LZ}}(\hat{X}^n) - s\cdot d(X^n,\hat{X}^n)\right] \leq H(\tilde{\mathbf{X}}_0|\tilde{\mathbf{X}}_{-k}^{-1}) + 2\epsilon - s(D+\delta). \tag{A-15}$$

The arbitrariness of k, ϵ and δ implies

$$\begin{aligned}\limsup_{n\to\infty} \mathbb{E}&\left[\frac{1}{n}\ell_{\mathsf{LZ}}(\hat{X}^n) - s\cdot d(X^n,\hat{X}^n)\right]\\ &\leq R(\mathbf{X},D) - s\cdot D,\end{aligned} \tag{A-16}$$

for any $D \geq 0$. Since D is also arbitrary, it follows that

$$\begin{aligned}\limsup_{n\to\infty} \mathbb{E}&\left[\frac{1}{n}\ell_{\mathsf{LZ}}(\hat{X}^n) - s\cdot d(X^n,\hat{X}^n)\right]\\ &\leq \min_{D\geq 0}[R(\mathbf{X},D) - s\cdot D],\end{aligned} \tag{A-17}$$

Finally, combining (A-3), and (A-17) we get the desired result:

$$\begin{aligned}\lim_{n\to\infty} \mathbb{E}&\left[\frac{1}{n}\ell_{\mathsf{LZ}}(\hat{X}^n) - s\cdot d(X^n,\hat{X}^n)\right]\\ &= \min_{D\geq 0}[R(\mathbf{X},D) - s\cdot D].\end{aligned} \tag{A-18}$$

APPENDIX B: PROOF OF THEOREM 2

Our proof follows the results presented in [35]. Throughout this section, $E = \mathcal{X}^N$ denotes the state space of our Markov chain (MC), **P** defines a stochastic transition matrix from E to itself, and π defines a distribution on E satisfying $\pi\mathbf{P} = \pi$. Moreover,

$$\mathbf{P} = \begin{bmatrix} p_1 \\ p_2 \\ \vdots \\ p_N \end{bmatrix}, \tag{B-1}$$

where $N := |\mathcal{X}|^n$ is the size of the state space. From this definition, each p_i is a row vector of length N in $\mathbb{R}^{+N}$ such that $\sum_j p_{ij} = 1$.

Definition 1 (Ergodic coefficient): The Dobrushin's ergodic coefficient of **P** , $\delta(\mathbf{P})$, is defined to be

$$\delta(\mathbf{P}) = \max_{1\leq i,j\leq N} \|\mathbf{p}_i - \mathbf{p}_j\|_{\mathrm{TV}}. \tag{B-2}$$

From the definition,

$$0 \leq \delta(\mathbf{P}) \leq 1. \tag{B-3}$$

Moreover, since

$$\begin{aligned}
\|\mathbf{p}_i - \mathbf{p}_j\|_{\mathrm{TV}} &= \frac{1}{2}\sum_{k=1}^{N} |p_{ik} - p_{jk}| \\
&= \frac{1}{2}\sum_{k=1}^{N} \left[(p_{ik} - p_{jk})1_{p_{ik}\geq p_{jk}} + (p_{jk} - p_{ik})1_{p_{jk}>p_{ik}}\right] \\
&= \frac{1}{2}(1 - \sum_{k=1}^{N} p_{ik}1_{p_{ik}\leq p_{jk}}) - \frac{1}{2}\sum_{k=1}^{N} p_{jk}1_{p_{ik}\geq p_{jk}} \\
&\quad + \frac{1}{2}(1 - \sum_{k=1}^{N} p_{jk}1_{p_{jk}\leq p_{ik}}) - \frac{1}{2}\sum_{k=1}^{N} p_{ik}1_{p_{ik}\leq p_{jk}} \\
&= 1 - \sum_{k=1}^{N} \left[p_{ik}1_{p_{ik}\leq p_{jk}} + p_{jk}1_{p_{ik}\geq p_{jk}}\right] \\
&= 1 - \sum_{k=1}^{N} \min(p_{ik}, p_{jk}), \qquad \text{(B-4)}
\end{aligned}$$

the ergodic coefficient can alternatively be defined as

$$\delta(\mathbf{P}) = 1 - \min_{1\leq i,j\leq N} \sum_{k=1}^{N} \min(p_{ik}, p_{jk}). \qquad \text{(B-5)}$$

The following theorem states the connection between the ergodic coefficient of a stochastic matrix and its convergence rate to the stationary distribution.

Theorem 4 (Convergence rate in terms of Dobrushin's coefficient): Let **P** Let μ and ν be two probability distributions on E. Then

$$\|\mu\mathbf{P}^t - \nu\mathbf{P}^t\|_{\mathrm{TV}} \leq \|\mu - \nu\|_{\mathrm{TV}}\delta(\mathbf{P})^t \qquad \text{(B-6)}$$

Corollary 1: By substituting $\nu = \pi$ in (B-6), we get

$$\|\mu\mathbf{P}^t - \pi\|_{\mathrm{TV}} \leq \|\mu - \pi\|_{\mathrm{TV}}\delta(\mathbf{P})^t. \qquad \text{(B-7)}$$

Thus far, we talked about homogenous MCs with stationary transition matrix. However, in simulated annealing we deal with a nonhomogeneous MC. The transition probabilities of a nonhomogeneous MC depend on time and vary as time proceeds. Let $\mathbf{P}^{(t)}$ denote the transition Matrix of the MC at time t, and for $0 \leq n_1 < n_2 \in \mathbb{N}$, define

$$\mathbf{P}^{(n_1,n_2)} := \prod_{t=n_1}^{n_2-1} \mathbf{P}^{(t)}. \qquad \text{(B-8)}$$

By this definition, if at time n_1 the distribution of the MC on the state space E is μ_{n_1}, at time n_2, the distribution evolves to $\mu_{n_2} = \mu_{n_1}\mathbf{P}^{(n_1,n_2)}$. The following two definitions characterize the steady state behavior of a nonhomogeneous MC.

Definition 2 (Weak Ergodicity): A nonhomogeneous MC is called weakly ergodic if for any distributions μ and ν over E, and any $n_1 \in \mathbb{N}$,

$$\limsup_{n_2\to\infty} \|\mu\mathbf{P}^{(n_1,n_2)} - \nu\mathbf{P}^{(n_1,n_2)}\|_{\mathrm{TV}} = 0. \qquad \text{(B-9)}$$

Definition 3 (Strong Ergodicity): A nonhomogeneous Markov chain is called strongly ergodic if there exists a distribution over the state space E such that for any distributions μ and $n_1 \in \mathbb{N}$,

$$\limsup_{n_2\to\infty} \|\mu\mathbf{P}^{(n_1,n_2)} - \pi\|_{\mathrm{TV}} = 0. \tag{B-10}$$

Any strongly ergodic MC is also weakly ergodic, because by triangle inequality

$$\|\mu\mathbf{P}^{(n_1,n_2)} - \nu\mathbf{P}^{(n_1,n_2)}\|_{\mathrm{TV}} \leq \|\mu\mathbf{P}^{(n_1,n_2)} - \pi\|_{\mathrm{TV}} + \|\nu\mathbf{P}^{(n_1,n_2)} - \pi\|_{\mathrm{TV}}. \tag{B-11}$$

The following theorem states a necessary and sufficient condition for weak ergodicity of a nonhomogeneous MC.

Theorem 5 (Block criterion for weak ergodicity): A MC is weakly ergodic iff there exists a sequence of integers $0 \leq n_1 < n_2 < \ldots$, such that

$$\sum_{i=1}^{\infty}(1 - \delta(\mathbf{P}^{(n_i,n_{i+1})})) = \infty. \tag{B-12}$$

Theorem 6 (Sufficient condition for strong ergodicity): Let the MC be weakly ergodic. Assume that there exists a sequence of probability distributions, $\{\pi^{(i)}\}_{i=1}^{\infty}$, on E such that

$$\pi^{(i)}\mathbf{P}^{(i)} = \pi^{(i)}. \tag{B-13}$$

Then the MC is strongly ergodic, if

$$\sum_{i=1}^{\infty}\|\pi^{(i)} - \pi^{(i+1)}\|_{\mathrm{TV}} < \infty. \tag{B-14}$$

After stating all the required definitions and theorems, finally we get back to our main goal which was to prove that by the mentioned choice of the $\{\beta_t\}$ sequence, Algorithm 1 converges to the optimal solution asymptotically as block length goes to infinity. Here $\mathbf{P}^{(j)}$, the transition matrix of the MC at the j-th iteration, depends on β_j. Using theorem 5, first we prove that the MC is weakly ergodic.

Lemma 1: The ergodic coefficient of $\mathbf{P}^{(jn,(j+1)n)}$, for any $j \geq 0$ is upper-bounded as follows

$$\delta(\mathbf{P}^{(jn,(j+1)n)}) \leq 1 - n\bar{\beta}_j\Delta, \tag{B-15}$$

where

$$\Delta = \max \delta_i, \tag{B-16}$$

and

$$\delta_i = \max\{|\mathcal{E}(u^{i-1}au_{i+1}^n) - \mathcal{E}(u^{i-1}bu_{i+1}^n)|;\ u^{i-1} \in \mathcal{Y}^{i-1}, u_{i+1}^n \in \mathcal{Y}^{n-i}, a, b \in \mathcal{Y}\}.$$

Proof: Let x^n and y^n be two arbitrary sequences in $\mathcal{X}^n$. Since the Hamming distance between these two sequence is at most n, starting from any sequence x^n, after at most n steps of the Gibbs sampler, it is possible to get to any other sequence y^n. On the other hand at each step the transition probabilities of jumping from one state to a neighboring state, i.e.

$$\mathbf{P}^{(t)}(x^{i-1}bx_{i+1}^n|x^{i-1}ax_{i+1}^n) = \frac{\exp(-\beta_t\mathcal{E}(x^{i-1}ax_{i+1}^n))}{\sum\limits_{b\in\mathcal{X}}\exp(-\beta_t\mathcal{E}(x^{i-1}bx_{i+1}^n))}, \tag{B-17}$$

can be upper bounded as follows. Dividing both the numerator and denominator of (B-17) by $\exp(-\beta_t\delta_i)$, we get

$$\mathbf{P}^{(t)}(x^{i-1}bx_{i+1}^n|x^{i-1}ax_{i+1}^n) = \frac{\exp(-\beta_t(\mathcal{E}(x^{i-1}ax_{i+1}^n)-\delta_i))}{\sum\limits_{b\in\mathcal{X}}\exp(-\beta_t(\mathcal{E}(x^{i-1}bx_{i+1}^n)-\delta_i))}, \tag{B-18}$$

$$\geq \frac{1}{|\mathcal{X}|}e^{-\beta_t\Delta}. \tag{B-19}$$

Therefore,

$$\min_{x^n,y^n\in\mathcal{X}^n}\mathbf{P}^{(jn,(j+1)n)}(x^n,y^n) \geq \prod_{t=jn}^{jn+n-1}\frac{1}{|\mathcal{X}|}e^{-\beta_t\Delta} = \frac{1}{|\mathcal{X}|^n}e^{-n\bar{\beta}_j\Delta}, \tag{B-20}$$

where $\bar{\beta}_j = \frac{1}{n}\sum\limits_{t=jn}^{jn+n-1}\beta_t$.

Using the alternative definition of the ergodic coefficient given in (B-5),

$$\delta(\mathbf{P}(jn,(j+1)n)) = 1 - \min_{x^n,y^n\in\mathcal{X}^n}\sum_{z^n\in\mathcal{X}^n}\min(\mathbf{P}^{(jn,(j+1)n)}(x^n,z^n),\mathbf{P}^{(jn,(j+1)n)}(y^n,z^n))$$

$$\leq 1 - |\mathcal{X}|^n\frac{1}{|\mathcal{X}|^n}e^{-n\bar{\beta}_j\Delta} \tag{B-21}$$

$$= 1 - e^{-n\bar{\beta}_j\Delta}. \tag{B-22}$$

■

Corollary 2: Let $\beta_t = \frac{\log(\lfloor\frac{t}{n}\rfloor+1)}{T_0^{(n)}}$, where $T_0^{(n)} = cn\Delta$, for some $c > 1$, and Δ is defined in (B-16), in Algorithm 1. Then the generated MC is weakly ergodic.

Proof: For proving weak ergodicity, we use the block criterion stated in Theorem 5. Let $n_j = jn$, and note that $\bar{\beta}_j = \frac{\log(j+1)}{T_0}$ in this case. Observe that

$$\sum_{j=0}^{\infty}(1-\delta(\mathbf{P}^{(n_j,n_{j+1})})) = \sum_{j=1}^{\infty}(1-\delta(\mathbf{P}^{(jn,(j+1)n)}))$$

$$\geq \sum_{j=0}^{\infty}e^{-n\bar{\beta}_j\delta} \tag{B-23}$$

$$= \sum_{j=0}^{\infty}e^{-n\Delta\frac{\log(j+1)}{T_0}} \tag{B-24}$$

$$= \sum_{j=1}^{\infty}\frac{1}{j^{1/c}} = \infty. \tag{B-25}$$

This yields the weak ergodicity of the MC defined by the simulated annealing Gibbs sampler. ■

Now we are ready to prove the result stated in Theorem 2. Using Theorem 6, we prove that the MC is in fact strongly ergodic and the eventual steady state distribution of the MC as the number of iterations converge to infinity is a uniform distribution over the sequences that minimize the energy function.

At each time t, the distribution defined as

$$\pi^{(t)}(y^n) = \frac{1}{Z_{\beta_t}}e^{-\beta_t\mathcal{E}(y^n)} \tag{B-26}$$

satisfies $\pi^{(t)}\mathbf{P}^{(t)} = \pi^{(t)}$. Therefore, if we prove that

$$\sum_{t=1}^{\infty} \|\pi^{(t)} - \pi^{(t+1)}\|_{\text{TV}} < \infty, \tag{B-27}$$

by Theorem 6, the MC is also strongly ergodic. But it is easy to show that $\pi^{(t)}$ converges to a uniforms distribution over the set of sequences that minimize the energy function, i.e.

$$\lim_{t\to\infty} \pi^{(t)}(x^n) = \begin{cases} 0; & x^n \notin H, \\ \frac{1}{|H|}; & x^n \in H, \end{cases} \tag{B-28}$$

where $H \triangleq \{y^n : \mathcal{E}(y^n) = \min\limits_{z^n \in \mathcal{X}^n} \mathcal{E}(z^n)\}$.

Hence, if we let $\hat{X}^n_t$ denote the output of Alg. 1 after t iterations, then

$$\lim_{t\to\infty} \mathcal{E}(\hat{X}^n_t) = \min_{y^n \in \mathcal{X}^n} \mathcal{E}(y^n), \tag{B-29}$$

which combined with Theorem 1 yields the desired result.

In order to prove (B-27), we prove that $\pi^{(t)}(x^n)$ is increasing on H, and eventually decreasing on H^c, hence there exists t_0 such that for any $t_1 > t_0$,

$$\begin{aligned} \sum_{t=t_0}^{t_1} \|\pi^{(t_1)} - \pi^{(t+1)}\|_{\text{TV}} &= \sum_{t=t_0}^{t_1} \frac{1}{2} \sum_{y^n \in \mathcal{X}^n} |\pi^{(t)}(y^n) - \pi^{(t+1)}(y^n)|, & \text{(B-30)} \\ &= \frac{1}{2} \sum_{y^n \in H} \sum_{t=t_0}^{t_1} (\pi^{(t+1)}(y^n) - \pi^{(t)}(y^n)) + \frac{1}{2} \sum_{y^n \in \mathcal{X}^n \setminus H} \sum_{t=t_0}^{t_1} (\pi^{(t)}(y^n) - \pi^{(t+1)}(y^n)), & \text{(B-31)} \\ &= \frac{1}{2} \sum_{y^n \in H} (\pi^{(t_1+1)}(y^n) - \pi^{(t_0)}(y^n)) + \frac{1}{2} \sum_{y^n \in \mathcal{X}^n \setminus H} (\pi^{(t_0)}(y^n) - \pi^{(t_1+1)}(y^n)) & \text{(B-32)} \\ &< \frac{1}{2}(1)(|\mathcal{X}|^n). & \text{(B-33)} \end{aligned}$$

Since the right hand side of (B-33) of does not depend on t_1,

$$\sum_{t=0}^{\infty} \|\pi^{(t)} - \pi^{(t+1)}\|_{\text{TV}} < \infty. \tag{B-34}$$

Finally, in order to prove that $\pi^{(t)}(y^n)$ is increasing for $y^n \in H$, note that

$$\begin{aligned} \pi^{(t)}(y^n) &= \frac{e^{-\beta_t \mathcal{E}(y^n)}}{\sum\limits_{z^n \in \mathcal{X}^n} e^{-\beta_t \mathcal{E}(z^n)}} \\ &= \frac{1}{\sum\limits_{z^n \in \mathcal{X}^n} e^{-\beta_t (\mathcal{E}(z^n) - \mathcal{E}(y^n))}}. & \text{(B-35)} \end{aligned}$$

Since for $y^n \in H$ and any $z^n \in \mathcal{X}^n$, $\mathcal{E}(z^n) - \mathcal{E}(y^n) \geq 0$, if $t_1 < t_2$,

$$\sum_{z^n \in \mathcal{X}^n} e^{-\beta_{t_1} (\mathcal{E}(z^n) - \mathcal{E}(y^n))} > \sum_{z^n \in \mathcal{X}^n} e^{-\beta_{t_2} (\mathcal{E}(z^n) - \mathcal{E}(y^n))},$$

and hence $\pi^{(t_1)}(y^n) < \pi^{(t_2)}(y^n)$. On the other hand, if $y^n \notin H$, then

$$\begin{aligned}\pi^{(t)}(y^n) &= \frac{e^{-\beta_t \mathcal{E}(y^n)}}{\sum\limits_{z^n \in \mathcal{X}^n} e^{-\beta_t \mathcal{E}(z^n)}} \\ &= \frac{1}{\sum\limits_{z^n : \mathcal{E}(z^n) \geq \mathcal{E}(y^n)} e^{-\beta_t(\mathcal{E}(z^n) - \mathcal{E}(y^n))} + \sum\limits_{z^n : \mathcal{E}(z^n) < \mathcal{E}(y^n)} e^{\beta_t(\mathcal{E}(y^n) - \mathcal{E}(z^n))}}. \end{aligned} \tag{B-36}$$

For large β the denominator of (B-36) is dominated by the second term which is increasing in β_t and therefore $\pi^{(t)}(y^n)$ will be decreasing in t. This concludes the proof.

APPENDIX : PROOF OF THEOREM 3

First we need to prove that a result similar to Theorem 1 holds for SB codes. I.e. we need to prove that for given sequences $\{k_f^{(n)}\}_n$ and $\{k_n\}_n$ such that $\lim\limits_{n\to\infty} k_f^{(n)} = \infty$ and $k_n = o(\log n)$, finding a sequence of SB codes according to

$$\hat{f}^{K_f^{(n)}} = \arg\min_{f^{K_f^{(n)}}} \mathcal{E}(f^{K_f^{(n)}}), \tag{C-1}$$

where $\mathcal{E}(f^{K_f^{(n)}})$ is defined in (23) and $K_f^{(n)} = 2^{2k_f^{(n)}+1}$, results in a sequence of asymptotically optimal codes for any stationary and ergodic source $\mathbf{X}$ at slope s. In other words,

$$\lim_{n\to\infty} \mathbb{E}[\frac{1}{n}\ell_{\mathsf{LZ}}(\hat{X}^n) - s \cdot d_n(\hat{X}^n, Y^n)] = \min_{D\geq 0}[R(\mathbf{X}, D) - s \cdot D], \quad \text{a.s.} \tag{C-2}$$

where $\hat{X}^n = \hat{X}^n[X^n, \hat{f}^{K_f^{(n)}}]$. After proving this, the rest of the proof follows from the proof of Theorem 2 by just redefining δ_i as

$$\delta_i = \max\left\{ |\mathcal{E}(f^{i-1} a f_{i+1}^{K_f^{(n)}}) - \mathcal{E}(f^{i-1} a f_{i+1}^{K_f^{(n)}})|;\ f^{i-1} \in \mathcal{Y}^{i-1}, f_{i+1}^{K_f^{(n)}} \in \mathcal{Y}^{K_f^{(n)}-i}, a, b \in \mathcal{Y} \right\}.$$

For establishing the equality stated in (C-2), like the proof of Theorem 1, we prove consistent lower and upper bounds which in the limit yield the desired result. The lower bound,

$$\liminf_{n\to\infty} \mathbb{E}\left[\frac{1}{n}\ell_{\mathsf{LZ}}(\hat{X}^n) - s \cdot d(X^n, \hat{X}^n)\right] \geq \min_{D\geq 0}\left[R(\mathbf{X}, D) - s \cdot D\right], \tag{C-3}$$

follows from an argument similar to the one given in the Appendix A. For proving the upper bound, we split the cost into two terms, as done in the equation (A-6). The convergence to zero of the first term again follows from a similar argument. The only difference is in upper bounding the second term.

Since, asymptotically, for any stationary ergodic process $\mathbf{X}$, SB codes have the same rate-distortion performance as block codes, for a point $(R(\mathbf{X}, D), D)$ on the rate-distortion curve of the source, and any $\epsilon > 0$, there exits a SB code $f^{2\kappa_f^{\epsilon}+1}$ of some order κ_f^{ϵ} such that coding the process $\mathbf{X}$ by this SB code results in a process $\tilde{\mathbf{X}}$ which satisfies

1) $H(\tilde{\mathbf{X}}) \leq R(\mathbf{X}, D)$,
2) $\mathbb{E}d(X_0, \tilde{X}_0) \leq D + \epsilon$.

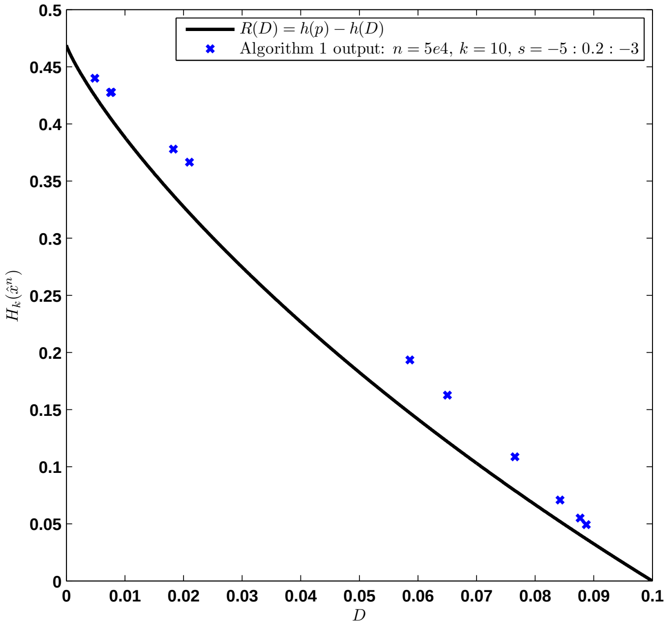


Fig. 4. Comparing the algorithm performance with the optimal rate-distortion tradeoff for a Bernoulli(p) i.i.d source, $p = 0.1$ and $\beta(t) = n\sqrt{\lceil \frac{t}{n} \rceil}$.

On the other hand, for a fixed n, $\mathcal{E}(f^{K_f})$ is monotonically decreasing in K_f. Therefore, for any process $\mathbf{X}$ and any $\delta > 0$, there exists n_δ such that for $n > n_\delta$ and $k_f^{(n)} \geq \kappa_f^\epsilon$

$$\limsup_{n\to\infty} \left[H_{k_n}(\hat{X}^n) - s \cdot d_n(X^n, \hat{X}^n) \right] \leq R(\mathbf{X}, D) - s \cdot (D + \epsilon) + \delta, \quad \text{w.p. } 1. \tag{C-4}$$

Combining (C-3) and (C-4), plus the arbitrariness of ϵ, δ and D yield the desired result.

References

[1] C. Shannon, “Coding theorems for a discrete source with a fidelity criterion,” *IRE Nat. Conv. Rec*, part 4, pp. 142-163, 1959.

[2] T. M. Cover, and J. A. Thomas, *Elements of Information Theory*, New York: Wiley, 1991.

[3] R.G. Gallager, “Information Theory and Reliable Communication,” New York, NY: John Wiley & Sons, 1968.

[4] T. Berger, *Rate-distortion theory: A mathematical basis for data compression*, Englewood Cliffs, NJ: Prentice-Hall, 1971.

[5] R. M. Gray, “Rate distortion functions for finite-state finite-alphabet Markov sources,” *IEEE Trans. on Inform. Theory*, vol. 17,no. 2, pp. 127- 134, 1971.

[6] R. M. Gray, “Information rates of autoregressive processes,” *IEEE Trans. on Inform. Theory*, vol. 16, pp. 412–421, July 1970.

[7] R. M. Gray, “Sliding-block source coding,” *IEEE Trans. on Inform. Theory*, vol. 21, pp. 357-368, July 1975.

[8] T. Berger, “Explicit bounds to $R(D)$ for a binary symmetric Markov source,” *IEEE Trans. on Inform. Theory*, 1977.

[9] J. Ziv and A. Lempel, “Compression of individual sequences via variable-rate coding,” *IEEE Trans. on Inform. Theory*, 24(5):530-536, Sep. 9178.

[10] E. Plotnik, M.J. Weinberger, J. Ziv, “Upper bounds on the probability of sequences emitted byfinite-state sources and on the redundancy of the Lempel-Ziv algorithm,” *IEEE Trans. Inform. Theory*, vol. 38, pp. 66-72, 1992.

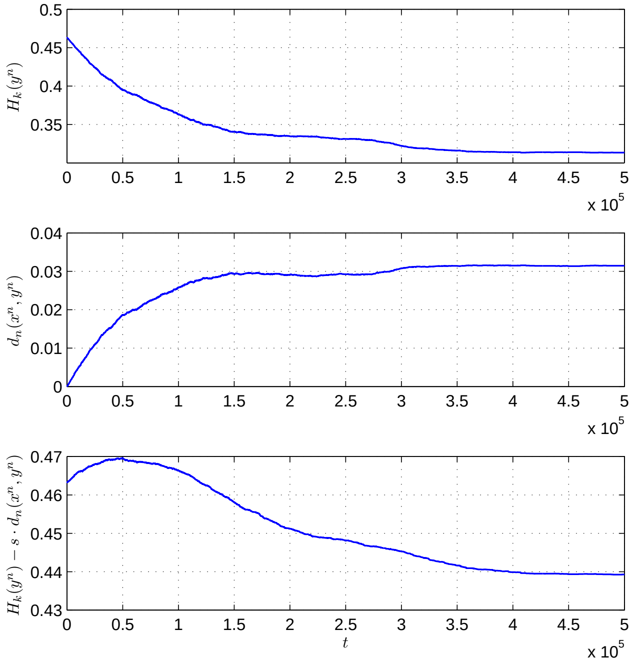


Fig. 5. Sample paths demonstrating evolutions of the empirical conditional entropy, average distortion, and energy function when Algorithm 1 is applied to the output of a Bernoulli(p) i.i.d source, with $p = 0.1$ and $n = 5e4$. The algorithm parameters are $k = 10$, $s = -4$, and $\beta(t) = 1.33n\sqrt{\lceil \frac{t}{n} \rceil}$.

[11] I. H. Witten, R. M. Neal, and J. G. Cleary, “Arithmetic coding for data compression”, *Commun. Assoc. Comp. Mach.*, vol. 30, no. 6, pp. 520-540, 1987.

[12] I. Kontoyiannis, “An implementable lossy version of the Lempel Ziv algorithm-Part I: optimality for memoryless sources,” *IEEE Trans. on Inform. Theory*, vol. 45, pp. 2293-2305, Nov. 1999.

[13] E. Yang, Z. Zhang, and T. Berger, “Fixed-slope universal lossy data compression,” , *IEEE Trans. on Inform. Theory*, vol. 43, no. 5, pp. 1465-1476, Sep. 1997.

[14] E. H. Yang and J. Kieffer, “Simple universal lossy data compression schemes derived from the Lempel-Ziv algorithm,” *IEEE Trans. on Inform. Theory*, vol. 42, no. 1, pp. 239-245, 1996.

[15] I. Kontoyiannis, “Pointwise redundancy in lossy data compression and universal lossy data compression,” *IEEE Trans. on Inform. Theory*, vol. 46, pp. 136-152, Jan. 2000.

[16] Y. Linde, A. Buzo, and R. M. Gray, “An algorithm for vector quantizer design,” *IEEE Trans. Commun.*, vol. COM-28, pp. 8495, Jan. 1980.

[17] S. Geman and D. Geman. “Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images,”. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 6, pp. 721-741, 1984.

[18] S. Kirkpatrick, C. D. Gelatt and M. P. Vecchi, “Optimization by simulated annealing”, *Science*, vol. 220, no. 4598, pp. 671-680, 1983.

[19] V. Cerny, “A thermodynamic approach to the travelling salesman problem: an efficient simulation algorithm,” *Journal of Optimization Theory and Applications*, vol. 45, pp. 41-51, 1985.

[20] K. Rose, “Deterministic annealing for clustering, compression, classification, regression, and related optimization problems,” *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2210-2239, Nov. 1998.

[21] J. Vaisey and A. Gersho, “Simulated annealing and codebook design,” *Proc. ICASSP*, pp. 1176-1179, 1988.

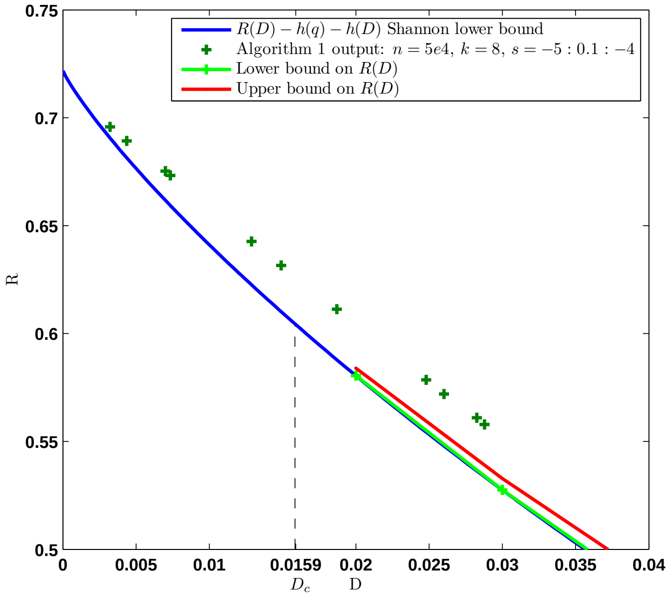


Fig. 6. Comparing the algorithm rate-distortion performance with the Shannon lower bound for a BSMS with $q = 0.2$, $\beta_t = n|s/3|t^{1.5}$

[22] E. Maneva and M. J. Wainwright, "Lossy source encoding via message passing, and decimation over generalized codewords of LDGM codes," *IEEE Int. Symp. on Inform. Theory*, Adelaide, Austrailia, Sep. 2005.

[23] A. Gupta and S. Verdú, "Nonlinear sparse-graph codes for lossy compression of discrete nonredundant sources," Information Theory Workshop, Lake Tahoe, California, USA, 2007.

[24] J. Rissanen and I. Tabus, "Rate-distortion without random codebooks," Workshop on Information Theory and Applications (ITA), UCSD, 2006.

[25] A. Gupta, S. Verdú, T. Weissman, "Linear-time near-optimal lossy compression", *IEEE Int. Symp. on Inform. Thoery*, Toronto, Canada, 2008

[26] K. Ramchandran and M. Vetterli, "Best wavelet packet bases in a rate-distortion sense", *IEEE Trans. on Image Process*, vol. 2, no. 2, pp. 160-175, April 1993.

[27] T. Weissman, E. Ordentlich, G. Seroussi, S. Verdú and M. Weinberger, "Universal discrete denoising: Known channel," *IEEE Trans. on Inform. Theory*, vol. 51, no. 1, pp. 5-28, Jan. 2005.

[28] B. Natarajan, K. Konstantinides, and C. Herley, "Occam filters for stochastic sources with application to digital images," *IEEE Trans. Signal Process.*, vol. 46, no. 11, pp. 14341438, Nov. 1998.

[29] D. Donoho, (2002, Jan.) The Kolmogorov sampler. [Online]. Available: http://www-stat.stanford.edu/ donoho/reports.html

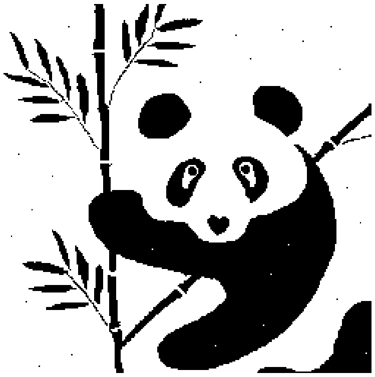
[30] T. Weissman, and E. Ordentlich, "The empirical distribution of rate-constrained source codes", *IEEE Trans. on Inform. Theory*, vol. 51 , no. 11 , pp. 3718- 3733, Nov. 2005

[31] E. Ordentlich, G. Seroussi, S. Verdú, M. Weinberger and T. Weissman, "A discrete universal denoiser and its application to binary images," Proc. *IEEE Int. Conf. on Image Processing*, p. 117-120, vol. 1, Sept. 2003.

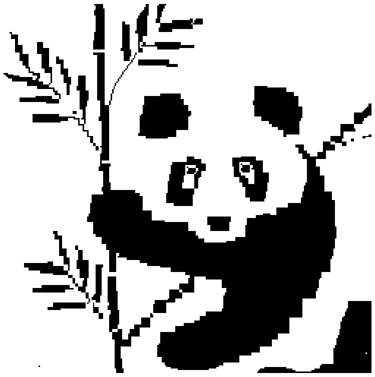
[32] R. Gray, D. Neuhoff, J. Omura, "Process definitions of distortion-rate functions and source coding theorems," *IEEE Trans. on Inf. Theory,* 21(5): 524-532, 1975.

[33] J. Ziv and A. Lempel, "Universal algorithm for sequential data compression," *IEEE Trans. on Inform. Theory*, vol. 23, pp. 337-343, 1977.

[34] H. Morita and K. Kobayashi, "On asymptotic optimality of a sliding window variation of Lempel-Ziv codes," *IEEE Trans. Inform. Theory*, vol. 39, pp. 1840-1846, 1993.



7.1: Original image with empirical conditional entropy of 0.1025



7.2: Reconstruction image with empirical conditional entropy of 0.0600 and average distortion of 0.0337 per pixel ($r = 50n^2$, $s = -0.1$, $\beta(t) = 0.1\log(t)$).

Fig. 7.

[35] P. Brémaud, *Markov chains, Gibbs fields, Monte Carlo simulation, and queues*, Springer, New York, 1998.

[36] S. Jalali and T. Weissman, "New bounds on the rate-distortion function of a binary Markov source," *Proc. IEEE Int. Symp. on Inform. Theory*, pp. 571-575, Jun. 2007.

[37] G. Motta, E. Ordentlich and M.J. Weinberger, "Defect list compression," *IEEE Int. Symp. on Inform. Theory*, Toronto, Canada, Jul. 2008.

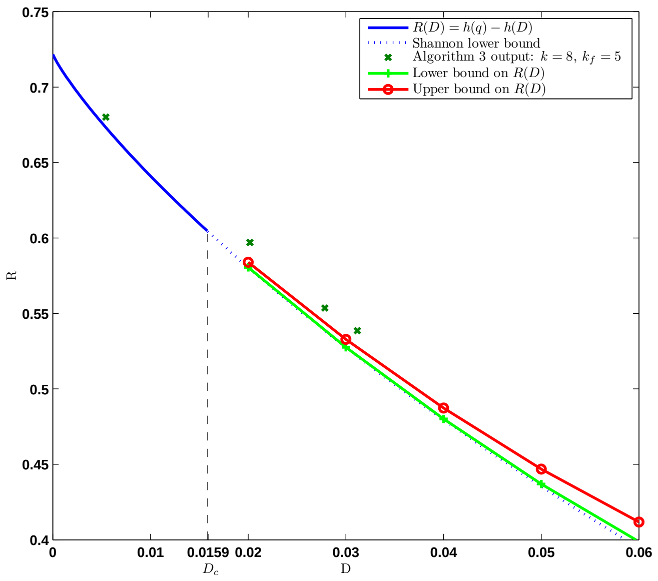


Fig. 8. Comparing the algorithm rate-distortion performance with the Shannon lower bound for a BSMS with $q = 0.2$. Algorithm parameters: $n = 5e4$, $k = 8$, $k_f = 5$ ($K_f = 2^{11}$), $\beta_t = K_f|s|\log(t+1)$, and slope values $s = -5.25$, -5, -4.75 and -4.5.

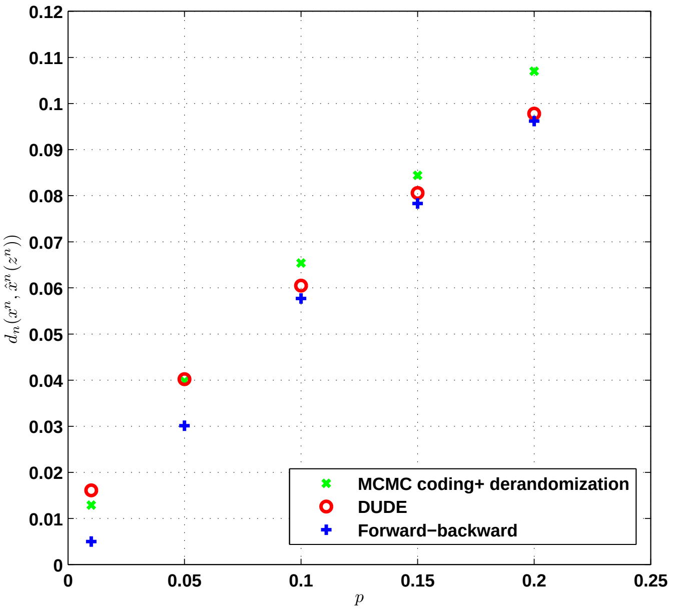
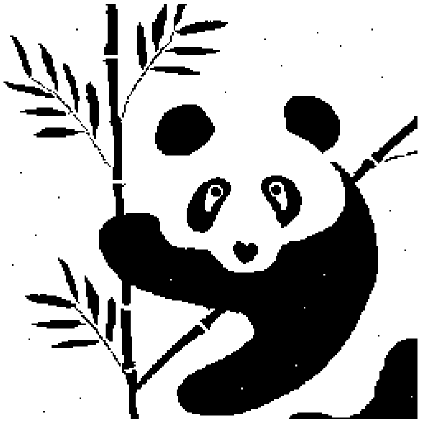
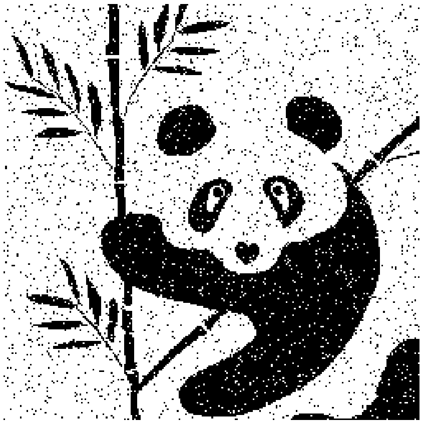


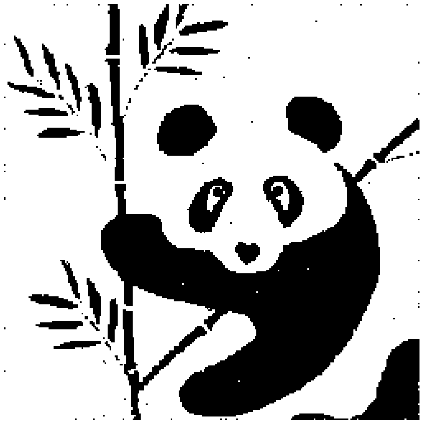
Fig. 9. Comparing the denoiser based on MCMC coding plus derandomization with DUDE and optimal non-universal Bayesian denoiser which is implemented via forward-backward dynamic programming. The source is a BSMS(p), and the channel is assumed to be a DMC with transition probability $\delta = 0.1$. The DUDE parameters are: $k_{\text{letf}} = k_{\text{right}} = 4$, and the MCMC coder uses $s = -0.9$, $\beta_t = 0.5\log t$, $r = 10n$, $n = 1e4$, $k = 7$. The derandomization window length is $2 \times 4 + 1 = 9$.



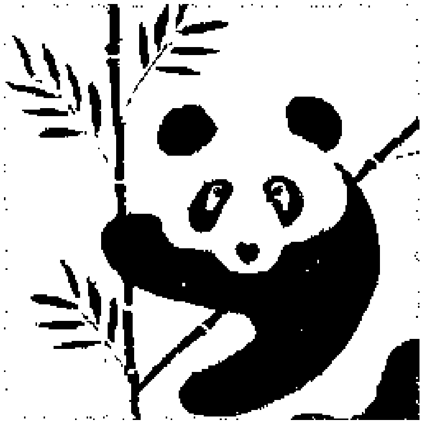
10.1: Original image.



10.2: Noisy image corrupted by a BSC(0.04).



10.3: DUDE reconstruction image with $d_n(x^n, \hat{x}^n) = 0.0081$: $k_{\text{letf}} = k_{\text{right}} = 4$.



10.4: MCMC coder + derandomization reconstruction image with $d_n(x^n, \hat{x}^n) = 0.0128$: $s = -2$, $\beta_t = 5 \log t$, $r = 10n^2$,