Denoising via MCMC-Based Lossy Compression

Shirin Jalali and Tsachy Weissman, *Senior Member, IEEE*

***Abstract*—It has been established in the literature, in various theoretical and asymptotic senses, that universal lossy compression followed by some simple postprocessing results in universal denoising, for the setting of a stationary ergodic source corrupted by additive white noise. However, this interesting theoretical result has not yet been tested in practice in denoising simulated or real data. In this paper, we employ a recently developed MCMC-based universal lossy compressor to build a universal compression-based denoising algorithm. We show that applying this iterative lossy compression algorithm with appropriately chosen distortion measure and distortion level, followed by a simple derandomization 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*—Compression-based denoising, denoising, Markov chain Monte Carlo, simulated annealing, universal lossy compression.**

I. Introduction

CONSIDER a discrete finite-alphabet random process $\mathbf{X} = \{X_i\}_{i=1}^{\infty}$ corrupted by an additive white (i.e., i.i.d.) noise process $\mathbf{Z} = \{Z_i\}_{i=1}^{\infty}$. The receiver observes the noisy process $\mathbf{Y} = \{Y_i\}_{i=1}^{\infty}$, where

$$Y_i = X_i + Z_i$$

and desires to recover the noise-free signal $\mathbf{X}$. For simplicity and concreteness, we assume that starting here and throughout the paper, the source, noise, and reconstruction alphabets are the M-ary alphabet, i.e., $\mathcal{X} = \hat{\mathcal{X}} = \mathcal{Z} = \{0, 1, \ldots, M-1\}$, and the noise is additive modulo-M. This model covers a natural and wide class of channels in the discrete finite-alphabet setting. (See [1] and references therein.) For example, a symmetric error channel is covered by the case where each Z_i has equal probability of assuming each of the $M-1$ nonzero elements. This channel, for $M = 4$, arises naturally when modeling errors in genomic data [2]. However, the idea and the approach can be extended to more general settings as well.

Let the vector $\boldsymbol{\pi} = (\pi(z))_{z\in\mathcal{Z}}$ denote the probability mass function (pmf) of the noise process $\mathbf{Z}$. That is, for $z \in \mathcal{Z}$ and $i \in \mathbb{N}$

$$\mathrm{P}(Z_i = z) = \pi(z).$$

Without loss of generality, assume that $\pi(z) > 0$, for each $z \in \mathcal{Z}$.

Let $\mathcal{Y}$ and $\hat{\mathcal{X}}$ denote the alphabets of the received signal and the reconstruction signal, respectively. Then, an n-block denoiser can be described by its block length n and denoising mapping

$$\theta_n : \mathcal{Y}^n \to \hat{\mathcal{X}}^n$$

such that $\hat{X}^n = \theta_n(Y^n)$. The average distortion between source and reconstruction blocks x^n and $\hat{x}^n$ is defined as

$$d_n(x^n, \hat{x}^n) \triangleq \frac{1}{n}\sum_{i=1}^{n} d(x_i, \hat{x}_i) \tag{1}$$

where $d : \mathcal{X} \times \hat{\mathcal{X}} \to \mathbb{R}^+$ is a per-letter distortion measure. The performance of an n-block denoiser θ_n is measured in terms of its expected average loss defined as

$$L_{\theta_n}(\mathbf{X}, \boldsymbol{\pi}) \triangleq \mathrm{E}\left[d_n\left(X^n, \theta_n(Y^n)\right)\right]. \tag{2}$$

The expectation in (2) is both with respect to the randomness in signal and the randomness in the noise. Therefore, the performance of a given denoiser depends on both the source distribution and the noise distribution.

In the case where the denoiser has knowledge of source distribution and noise pmf $(\pi(z))_{z\in\mathcal{Z}}$, the optimal denoiser which minimizes expected average loss $L_{\theta_n}(\mathbf{X}, \pi)$ over all possible n-block denoisers is a Bayesian denoiser whose i^{th} reconstruction is given by

$$\hat{X}_i^{\text{Bayes}} = \hat{X}_i^{\text{Bayes}}(Y^n) \triangleq \arg\min_{\hat{x}\in\hat{\mathcal{X}}} \mathrm{E}\left[d(X_i, \hat{x})|Y^n\right] \tag{3}$$

where $Y^n = X^n + Z^n$. For stationary ergodic source $\mathbf{X}$ corrupted by an additive white noise process with pmf $\boldsymbol{\pi}$, let $L^{\text{opt},B}(\mathbf{X}, \boldsymbol{\pi})$ denote the asymptotic performance of the Bayesian denoiser defined by the set of denoisers $\{\hat{X}_i^{\text{Bayes}}\}_{i=1}^{n}$. In other words

$$L^{\text{opt},B}(\mathbf{X}, \boldsymbol{\pi}) = \lim_{n\to\infty} \mathrm{E}\left[\frac{1}{n}\sum_{i=1}^{n} d\left(X_i, \hat{X}_i^{\text{Bayes}}\right)\right] \tag{4}$$

where the limit exists by subadditivity. In the case where $\mathbf{X}$ is Markov, the solution of (3) can be obtained efficiently by dynamic programming via the backward-forward recursions [3], [4].

In many practical situations, the assumption that the source distribution is available to the denoiser is unrealistic. Therefore, it is desirable to construct denoising algorithms that are oblivious of the source distribution, and yet achieve reasonable performance. In fact, it can be shown that the knowledge of the source distribution is not essential, and as long as the denoiser has access to the noise distribution $\boldsymbol{\pi}$, it can achieve $L^{\text{opt},B}(\mathbf{X}, \boldsymbol{\pi})$. In other words, if only the noise distribution $\boldsymbol{\pi}$

Manuscript received July 21, 2011; revised December 17, 2011; accepted February 23, 2012. Date of publication March 12, 2012; date of current version May 11, 2012. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Roberto Lopez-Valcarce.

S. Jalali is with the Center for Mathematics of Information, California Institute of Technology, Pasadena, CA 91125 USA.

T. Weissman is with the Department of Electrical Engineering, Stanford University, Stanford, CA 94305 USA.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSP.2012.2190597

is known by the denoiser, there exists a family of n-block denoisers that asymptotically achieves the optimal performance $L^{\text{opt},B}(\mathbf{X},\boldsymbol{\pi})$, for any stationary ergodic $\mathbf{X}$ [5]. In fact, not only is this performance achievable for any stationary ergodic process $\mathbf{X}$, but it can be achieved with linear-complexity via the discrete universal denoising (DUDE) algorithm proposed in [5].

The DUDE first estimates the probability distribution of the source using its observed noisy signal, and then performs a Bayesian-type denoising operation using the estimated statistics. A different approach to denoising is based on lossy compression of the noisy signal. This method provides an alternative and more implicit method for learning the required source statistics. After obtaining such statistics, the Bayesian estimation can be performed as done by the DUDE. The intuition behind this approach is as follows. In universal lossy compression, to encode a sequence y^n, the encoder looks for a sequence $\hat{y}^n$ "close" to y^n which is more compressible. At a high level, lossy compression of y^n at distortion level D can be done by searching among all sequences $\hat{y}^n$ that are within radius D of y^n, i.e., $d_n(y^n,\hat{y}^n)\leq D$, and choosing the one that has the lowest "complexity" or "description length." On the other hand, adding noise to a signal always increases its entropy, i.e., since $I(X^n+Z^n;Z^n)\geq 0$, it follows that:

$$\begin{aligned} &H(X^n+Z^n)-H(X^n+Z^n|Z^n)\\ &\qquad\qquad =H(X^n+Z^n)-H(X^n)\geq 0 \qquad (5)\end{aligned}$$

and therefore $H(Y^n)\geq H(X^n)$. Hence, in lossy compression of noisy sequence $Y^n=X^n+Z^n$, if the distortion level is set appropriately, a reasonable candidate for the reconstruction sequence can be the original sequence X^n.

Minimum Kolmogorov complexity estimator (MKCE) proposed in [6] is constructed based on the same intuition. Let X^n be a binary sequence passed through an additive binary channel with $\pi(1)=1-\pi(0)=\delta$. Let Y^n denote the received binary signal. The MKCE denoiser looks for the minimizer of the following optimization problem

$$\begin{aligned} \min \quad & K(\hat{x}^n)\\ \text{subject to} \quad & x^n\in\{0,1\}^n,\\ & d_n(Y^n,x^n)\leq\delta. \qquad (6)\end{aligned}$$

In (6), $K(\hat{x}^n)$ represents the Kolmogorov complexity of $\hat{x}^n$ [7]. Basically, $K(\hat{x}^n)$ measures the complexity or compressibility of $\hat{x}^n$. It is shown in [6] that the performance of MKCE is strictly worse than an optimal denoiser, but by a factor no larger than 2. Later this result was refined in [8] and was shown that replacing (6) with a universal lossy compressor, and then performing some postprocessing operation results in a universal denoising algorithm with asymptotically optimal performance. As explained before, the role of universal lossy compressor is helping the denoiser to estimate the distribution of the source. Using the estimated source statistics, the postprocessing operation consists of Bayesian denoising.

Compression-based denoising algorithms have been proposed before by a number of researchers. It has been studied both from a theoretical standpoint [6], [8] and from a practical point of view. (See [9]–[13] and references therein.) While the theoretical results, specially the work of [8], suggest that compression-based denoising is able to achieve the optimal performance, there is yet a gap between the theory and practice in this area. The implementable algorithms, while achieving promising results, are suboptimal, and the theoretical results have not yet led to practical compression-based denoising algorithms. In this paper, we show how combining the lossy compression algorithm proposed in [14] and the denoising approach of [8] leads to an implementable universal denoiser. Our simulation results show that the performance of the resulting scheme is comparable with the performance of the DUDE, when applied to one-dimensional or two-dimensional binary data.

The lossy compression algorithm proposed in [14] is based on Gibbs sampling, and simulated annealing [16]–[18]. Consider the probability distribution p over all sequences in $\mathcal{X}^n$, such that for $x^n\in\mathcal{X}^n$, $p(x^n)\propto f(x^n)$, where $f(x^n)\geq 0$, for all x^n. In many applications, it is desirable to sample from such a distribution, which is only specified through another function f. Clearly, $p(x^n)=\frac{f(x^n)}{Z}$, where $Z=\sum_{x^n\in\mathcal{X}^n}f(x^n)$. However, since the size of the sample space grows exponentially with n, computing Z in general requires exponential computational complexity. Therefore, to sample from p, one usually looks for sampling methods that do not directly require the computation of Z. Markov chain Monte Carlo (MCMC) methods are a class of algorithms that address this issue. They consist of a class of sampling algorithms that sample from distribution p by generating a Markov chain whose stationary distribution is p. Hence, running such a Markov chain, after it reaches the steady state, its state is a sample drawn from the distribution p. The Gibbs sampler, also known as the *heat bath* algorithm, is an instance of the MCMC methods. It is applied in the cases where the computational complexity of finding the conditional distributions of each variable given the rest, i.e., $p(x_i|x^{n\backslash i})$, is manageable.

Simulated annealing is a well-known method in discrete optimization problems. Its goal is to find the minimizer of by a given real-valued function h over the finite set $\mathcal{X}^n$, i.e., $x_o^n=\arg\min_{x^n\in\mathcal{X}^n}h(x^n)$. In order to perform simulated annealing, a sequence of probability distributions $\{p_i(x^n)\}_{i=1}^{\infty}$ corresponding to temperatures $\{T_i\}_{i=1}^{\infty}$ is considered. The temperatures are chosen such that $T_i\rightarrow 0$ as $i\rightarrow\infty$. At each time i, the algorithm runs one of the relevant MCMC methods in an attempt to sample from distribution $p_i(x^n)\propto e^{-\beta_i h(x^n)}$, where $\beta_i=1/T_i$. Note that as $T_i\rightarrow 0$ (or $\beta_i\rightarrow\infty$), p_i converges to a probability distribution which is uniform over the set of minimizers of h, and zero otherwise. Hence, clearly a sample from this distribution gives a minimizer of h. On the other hand, starting from an extremely low temperature results in a Markov chain that requires exponential time to reach the steady state. Therefore, in simulated annealing, the algorithm starts from a moderate temperature, and decreases the temperature gradually. It can be proved that, if the temperature drops slowly enough, the probability of getting the minimizing state as the output of the algorithm approaches one [16].

The organization of this paper is as follows. Section II introduces the notation and definitions used in this paper. Section III reviews the universal lossy compression algorithm proposed in [14]. In Section IV, we show how the universal lossy compression algorithm of [14] can be employed to construct a universal denoising algorithm. Section V presents some experimental results. Section VI concludes the paper.

II. Notation and Definitions

Calligraphic letters such as $\mathcal{X}$ and $\mathcal{Y}$ denote sets. The size of a set $\mathcal{X}$ is denoted by $|\mathcal{X}|$. Bold letters such as $\mathbf{X} = (X_1, X_2, \dots, X_n)$ and $\mathbf{x} = (x_1, x_2, \dots, x_n)$ represent n-tuples, where n is implied by the context. An n-tuple $\mathbf{X}$ or $\mathbf{x}$ of length n is also represented as X^n or x^n when we want to be explicit about n. For $1 \leq i \leq j \leq n$, $x_i^j = (x_i, x_{i+1}, \dots, x_j)$. For two vectors x^i and y^j, $x^i y^j$ denotes a vector of length $i+j$ formed by concatenating the two vector as $(x_1, \dots, x_i, y_1, \dots, y_j)$. Capital letters represent random variables and capital bold letters represent random vectors. For a random variable X, let $\mathcal{X}$ denote its alphabet set. For vectors $\mathbf{u}$ and $\mathbf{v}$ both of length n, let $\|\mathbf{u}-\mathbf{v}\|_1$ denote the ℓ_1 distance between $\mathbf{u}$ and $\mathbf{v}$ defined as $\|\mathbf{u}-\mathbf{v}\|_1 \triangleq \sum_{i=1}^n |u_i - v_i|$.

For $y^n \in \mathcal{Y}^n$, let the $|\mathcal{Y}| \times |\mathcal{Y}|^k$ matrix $\mathbf{m}(y^n)$ denote the $(k+1)^{\text{th}}$ order empirical distribution of y^n. Each column of matrix $\mathbf{m}(y^n)$ is indexed by a k-tuple $b^k \in \mathcal{Y}^k$, and each row is indexed by an element $\beta \in \mathcal{Y}$. The element in row β and column b^k of $\mathbf{m}(y^n)$ is defined as

$$m_{\beta,b^k}(y^n) \triangleq \frac{1}{n-k} \left|\left\{k+1 \leq i \leq n : y_{i-k}^{i-1} = b^k, y_i = \beta\right\}\right|$$

i.e., the fraction of occurrences of the $(k+1)$-tuple $b^k\beta$ along the sequence. Let $H_k(y^n)$ denote the conditional empirical entropy of order k induced by y^n, i.e.

$$H_k(y^n) \triangleq \sum_{b^k \in \mathcal{Y}^k} \left\|\mathbf{m}_{\cdot,b^k}(y^n)\right\|_1 \mathcal{H}\left(\mathbf{m}_{\cdot,b^k}(y^n)\right)$$

where $\mathbf{m}_{\cdot,b^k}(y^n)$ denotes the column in $\mathbf{m}(y^n)$ corresponding to b^k, and for a vector $\mathbf{v} = (v_1, \dots, v_\ell)$ with nonnegative components, $\mathcal{H}(\mathbf{v})$ denotes the entropy of the random variable whose pmf is proportional to $\mathbf{v}$. Formally

$$\mathcal{H}(\mathbf{v}) \triangleq \begin{cases} \sum\limits_{i=1}^{\ell} \frac{v_i}{\|\mathbf{v}\|_1} \log \frac{\|\mathbf{v}\|_1}{v_i}, & \text{if } \mathbf{v} \neq (0,\dots,0) \\ 0, & \text{if } \mathbf{v} = (0,\dots,0) \end{cases}$$

where $0\log(0) \triangleq 0$ by convention. Alternatively, $H_k(y^n) \triangleq H(U_{k+1}|U^k)$, where U^{k+1} is distributed according to the $(k+1)^{\text{th}}$ order empirical distribution induced by y^n, i.e., $\mathrm{P}(U^{k+1} = b^k\beta) = m_{\beta,b^k}(y^n)$.

Consider lossy compression of a stationary ergodic source $\mathbf{X} = \{X_i\}_{i=1}^{\infty}$. The encoder maps each source output block of length n, X^n, to a binary sequence $f_n(X^n)$, i.e.

$$f_n : \mathcal{X}^n \to \{0,1\}^*$$

where $\{0,1\}^*$ denotes the set of all finite-length binary sequences. The index $f_n(X^n)$ is then losslessly transmitted to the decoder[1]. The decoder maps $f_n(X^n)$ into a reconstruction block $\hat{X}^n = g_n(f_n(X^n))$, where

$$g_n : \{0,1\}^* \to \hat{\mathcal{X}}^n.$$

The performance of a lossy coding algorithm $\mathcal{C}_n = (f_n, g_n)$ with block length n is measured by its induced rate R_n and distortion D_n. Let $l(f_n(X^n))$ denote the length of the binary sequence assigned to sequence X^n. The rate R_n of code $\mathcal{C}_n$ is defined as the expected average number of bits per source symbol, i.e.

$$R_n \triangleq \mathrm{E}\left[\frac{l\left(f_n(X^n)\right)}{n}\right].$$

The distortion D_n induced by code $\mathcal{C}_n$ is defined as the average expected distortion between source and reconstruction blocks, i.e.

$$D_n \triangleq \mathrm{E}\left[d_n(X^n, \hat{X}^n)\right]$$

where $d_n(x^n, \hat{x}^n)$ is defined according to (1), and, as before, $d : \mathcal{X} \times \hat{\mathcal{X}} \to \mathbb{R}^+$ defines a single-letter distortion measure.

For any $D \geq 0$, and stationary ergodic process $\mathbf{X}$, the minimum achievable rate (cf. [7] for exact definition of achievability) is characterized as [19]–[21]

$$R(D, \mathbf{X}) = \lim_{n\to\infty} \min_{p(\hat{x}^n|x^n) : \mathrm{E}\left[d_n(X^n,\hat{X}^n)\right] \leq D} \frac{1}{n} I(X^n; \hat{X}^n).$$

III. Universal Lossy Compression via MCMC

Lossy compression algorithms can be divided into three groups as: i) fixed-rate, ii) fixed-distortion, and iii) fixed-slope. A family of universal lossy compression algorithms $\mathcal{C}_n = (f_n, g_n)$, $n \geq 1$, is called fixed-rate, if, for every stationary ergodic source $\mathbf{X}$, $R_n \leq R$, for all n, and $\limsup_n D_n \leq D(R, \mathbf{X})$. Similarly, a family of codes is called fixed distortion, if, for every stationary ergodic source, $\mathbf{X}$, $D_n \leq D$, for all n, and $\limsup_n R_n \leq R(D, \mathbf{X})$. Finally, a family of codes is called fixed slope, if, for every stationary ergodic source $\mathbf{X}$, $\limsup_n [R_n + \alpha D_n] = \min_{D \geq 0}[R(D, \mathbf{X}) + \alpha D]$. For a given source $\mathbf{X}$, a fixed-slope universal lossy compression algorithm at slope α asymptotically achieves point (D_α, R_α), which is the point on the rate-distortion curve $R(D, \mathbf{X})$, where the slope $\frac{\partial R(D,\mathbf{X})}{\partial D}$ is equal to $-\alpha$.

For a fixed slope $\alpha > 0$, consider the quantization mapping $\hat{x}^n : \mathcal{X}^n \to \hat{\mathcal{X}}^n$ defined as

$$\hat{x}^n = \arg\min_{y^n} \left[H_k(y^n) + \alpha d_n(x^n, y^n)\right]. \tag{7}$$

Finding the solution of (7), and losslessly conveying it to the decoder using a universal lossless compression algorithm such as the Lempel-Ziv (LZ) algorithm constitutes a lossy compression algorithm. It can be proved that the described scheme attains the optimum rate-distortion performance at the slope α, universally for any stationary ergodic process [14], [22] in a strong almost sure sense. In other words, for any stationary ergodic source $\mathbf{X}$,

$$\frac{1}{n}\ell_{\mathsf{LZ}}(\hat{X}^n) + \alpha d_n(X^n, \hat{X}^n) \overset{n\to\infty}{\longrightarrow} \min_{D \geq 0}\left[R(D, \mathbf{X}) + \alpha D\right], \tag{8}$$

almost surely. In (8), $\ell_{\mathsf{LZ}}(\hat{X}^n)$ denotes the description length of $\hat{X}^n$ by the LZ algorithm [23], [24].

[1] Whether or not a unique decodability or even prefix condition is imposed on the lossless description of the index does not affect the achievable performance in the limit of large blocklengths, which is our focus in what follows.

To find the minimizer of (7), one needs to search the space of all possible reconstruction sequences which is of size $|\hat{\mathcal{X}}|^n$. Hence, although the described scheme is theoretically appealing, it is impractical and its implementation requires an exhaustive search.

In [14], it is shown how simulated annealing enables us to get close to the performance of the impractical exhaustive search coding algorithm. In the rest of this section, we briefly review the lossy compression algorithm proposed in [14].

To each reconstruction sequence $y^n \in \hat{\mathcal{X}}^n$, assign energy $\mathcal{E}(y^n) \triangleq [H_k(y^n) + \alpha d_n(x^n, y^n)]$, and define the probability distribution p_β on the space of reconstruction sequences $\hat{\mathcal{X}}^n$ as

$$p_\beta(y^n) = \frac{1}{Z_\beta} e^{-\beta \mathcal{E}(y^n)}$$

where β and Z_β denote the inverse temperature parameter and the normalization constant (partition function), respectively. Sampling from this distribution for some large β results in a sequence Y^n that, with high probability, has energy $\mathcal{E}(\mathcal{Y}^\backslash)$ very close to the minimum energy, i.e.

$$H_k(Y^n) + \alpha d_n(x^n, Y^n) \approx \min_{y^n} \left[H_k(y^n) + \alpha d_n(x^n, y^n) \right].$$

However, sampling from the distribution p_β for large values of β is a challenging task. A well-known approach to circumvent this difficulty is the idea of *simulated annealing*. The main idea in simulated annealing is to explore the search space for the state of minimum energy using a time-dependent random walk. The random walk is designed such that the probability of moving from the current state to one of its *neighboring* states depends on the difference between their energies. To give the algorithm the ability to escape from local minima, the Markov chain allows leaving a state of lower energy to reach a state of higher energy. As time proceeds, the system freezes ($\beta \to \infty$), and the probability of having such energy-increasing jumps decreases to zero.

The lossy compression algorithm based on simulated annealing presented in [14] is described in Algorithm 1. In Algorithm 1, $\mathrm{P}(Y_i = \cdot | Y^{n \backslash i} = y^{n \backslash i})$ denotes the conditional probability of Y_i given $Y^{n \backslash i} \triangleq (Y_n : n \neq i)$ under p_{β_t}. For $a \in \hat{\mathcal{X}}$, $\mathrm{P}(Y_i = a | Y^{n \backslash i} = y^{n \backslash i}) = p_\beta(a | y^{n \backslash i})$ can be expressed as

$$\begin{aligned}
&\mathrm{P}(Y_i = a | Y^{n \backslash i} = y^{n \backslash i}) \\
&= \frac{p_\beta\left(y^{i-1} a y_{i+1}^n\right)}{\sum_{b \in \hat{\mathcal{X}}} p_\beta\left(y^{i-1} b y_{i+1}^n\right)} \\
&= \frac{e^{-\beta \mathcal{E}\left(y^{i-1} a y_{i+1}^n\right)}}{\sum_{b \in \hat{\mathcal{X}}} e^{-\beta \mathcal{E}\left(y^{i-1} b y_{i+1}^n\right)}} \\
&= \frac{e^{-\beta\left(H_k\left(y^{i-1} a y_{i+1}^n\right) + \alpha d_n\left(x^n, y^{i-1} a y_{i+1}^n\right)\right)}}{\sum_{b \in \hat{\mathcal{X}}} e^{-\beta\left(H_k\left(y^{i-1} b y_{i+1}^n\right) + \alpha d_n\left(x^n, y^{i-1} b y_{i+1}^n\right)\right)}} \\
&= \frac{1}{1 + \sum\limits_{b \in \hat{\mathcal{X}}, b \neq a} e^{-\beta\left(\Delta H_k\left(a, b, y^{i-1}, y_{i+1}^n\right) + \alpha \Delta d(a, b, x_i)\right)}} \qquad (9)
\end{aligned}$$

where

$$\Delta H_k\left(a, b, y^{i-1}, y_{i+1}^n\right) \triangleq H_k\left(y^{i-1} b y_{i+1}^n\right) - H_k\left(y^{i-1} a y_{i+1}^n\right)$$

and

$$\begin{aligned}
\Delta d(b, a, x_i) &\triangleq d_n\left(x^n, y^{i-1} b y_{i+1}^n\right) - d_n\left(x^n, y^{i-1} a y_{i+1}^n\right) \\
&= \frac{d(x_i, b) - d(x_i, a)}{n}.
\end{aligned}$$

Computing the conditional probability distributions described in (9) forms the main step of Algorithm 1.

Algorithm 1: Generating the reconstruction sequence

```
Input: $x^n, k, \alpha, \{\beta_t\}_t, r$
Output: a reconstruction sequence $\hat{x}^n$
$y^n \leftarrow x^n$
for $t = 1$ to $r$ or do
    Draw an integer $i \in \{1, \ldots, n\}$ uniformly at random.
    For each $b \in \hat{\mathcal{X}}$ compute $p_{\beta_t}(b | y^{n \backslash i})$ given in (9).
    Update $y^n$ by replacing its $i^{\mathrm{th}}$ component $y_i$ by $Z$, where
    $Z \sim p_{\beta_t}(\cdot | y^{n \backslash i})$.
    Update $\mathbf{m}(y^n)$ and $H_k(y^n)$.
end for
$\hat{x}^n \leftarrow y^n$
```

Note that

$$\begin{aligned}
&\Delta H_k\left(a, b, y^{i-1}, y_{i+1}^n\right) \\
&\quad = H_k\left(y^{i-1} b y_{i+1}^n\right) - H_k\left(y^{i-1} a y_{i+1}^n\right) \\
&\quad\quad \times \sum_{b^k \in \mathcal{Y}^k} \Big[\left\| \mathbf{m}_{\cdot, b^k}\left(y^{i-1} b y_{i+1}^n\right) \right\|_1 \mathcal{H}\left(\mathbf{m}_{\cdot, b^k}\left(y^{i-1} b y_{i+1}^n\right)\right) \\
&\quad\quad\quad - \left\| \mathbf{m}_{\cdot, b^k}\left(y^{i-1} a y_{i+1}^n\right) \right\|_1 \\
&\quad\quad\quad \times \mathcal{H}\left(\mathbf{m}_{\cdot, b^k}\left(y^{i-1} a y_{i+1}^n\right)\right) \Big]. \qquad (10)
\end{aligned}$$

On the other hand, changing the i^{th} element of $y^{i-1} b y_{i+1}^n$ from b to a affects at most $2k+1$ columns of the matrix $\mathbf{m}(y^{i-1} b y_{i+1}^n)$. In other words, at least $2^k - 2k - 1$ columns of $\mathbf{m}(y^{i-1} b y_{i+1}^n)$ and $\mathbf{m}(y^{i-1} a y_{i+1}^n)$ are exactly the same. Hence, from (10), for given values of n and k, the number of operations required for computing $\Delta H_k(a, b, y^{i-1}, y_{i+1}^n)$ is linear in k, and independent of n.

Let $\hat{X}_{\alpha, r}^n(X^n)$ denote the (random) outcome of Algorithm 1 when taking $k = k_n$ and $\boldsymbol{\beta} = \{\beta_t\}_t$ to be deterministic sequences satisfying $k_n = o(\log n)$ and

$$\beta_t = \frac{1}{T_0^{(n)}} \log\left(\left\lfloor \frac{t}{n} \right\rfloor + 1 \right)$$

for some $T_0^{(n)} > n\Delta$, where

$$\Delta = \max_i \max_{\begin{cases} u^{i-1}\in\hat{\mathcal{X}}^{i-1}, \\ u_{i+1}^n \in \hat{\mathcal{X}}^{n-i}, \\ a,b\in\hat{\mathcal{X}} \end{cases}} \left|\mathcal{E}\left(u^{i-1}au_{i+1}^n\right) - \mathcal{E}\left(u^{i-1}bu_{i+1}^n\right)\right| \tag{11}$$

applied to the source sequence X^n as input.[2] By the previous discussion, for given n and k, the computational complexity of the algorithm at each iteration is independent of n and linear in k. It can be proved that this choice of parameters yields a universal lossy compression algorithm, i.e., for any stationary ergodic process $\mathbf{X}$

$$\lim_{n\to\infty}\lim_{r\to\infty}\left[\frac{1}{n}\ell_{\mathsf{LZ}}\left(\hat{X}_{\alpha,r}^n(X^n)\right) + \alpha d_n(X^n,\hat{X}^n)\right] = \min_{D\geq 0}\left[R(D,\mathbf{X}) + \alpha D\right],$$

almost surely.

IV. Denoising via MCMC-Based Lossy Compression

In [8], it is shown how a universally optimal lossy coder tuned to the right distortion measure and distortion level combined with some simple "postprocessing" results in a universally optimal denoiser. In what follows, we first briefly review the compression-based denoiser described in [8], and then show how the lossy coder proposed in [14] can be used to perform the required universal lossy compression.

Define the *difference* distortion measure $\rho : \mathcal{X} \times \hat{\mathcal{X}} \to \mathbb{R}^+$ as

$$\rho(x,\hat{x}) \triangleq \log\frac{1}{\pi(x-\hat{x})}. \tag{12}$$

As a reminder, in (12), π denotes the pmf of the noise. Also, as before, let $\rho_n(x^n,\hat{x}^n) = n^{-1}\sum_{i=1}^n \rho(x_i,\hat{x}_i)$.

Now consider a sequence of universal lossy compression codes $\mathcal{C}_n = (f_n, g_n)$ at fixed distortion $H(Z)$ under distortion measure ρ, i.e., a sequence of codes such that

$$\mathrm{E}\left[\rho_n\left(Y^n, g_n\left(f_n(Y^n)\right)\right)\right] \leq H(Z)$$

and

$$\limsup_{n\to\infty} \mathrm{E}\left[\frac{l\left(f_n(Y^n)\right)}{n}\right] = R\left(H(Z),\mathbf{Y}\right)$$

for every stationary ergodic process $\mathbf{Y} = \{Y_i\}_{i=1}^\infty$.

As aforementioned, in the denoising scheme outlined in [8], first the denoiser compresses the noisy signal appropriately, and partially removes the additive noise through lossy compression. To achieve this goal we apply the described universal lossy compression code to the noisy signal Y^n to get $\hat{Y}^n = g_n(f_n(Y^n))$.

The next step is a "postprocessing" step, which involves computing the joint empirical distribution between the noisy signal and its compressed version, and then constructing the final reconstruction sequence based on this empirical distribution. For a given integer $m = 2m_o + 1 > 0$, the empirical joint distribution $\hat{p}_{[Y^n,\hat{Y}^n]}^{(m)}(y^m,\hat{y})$ of the noisy signal Y^n and its quantized version $\hat{Y}^n$ is defined as follows. For $y^m \in \mathcal{Y}^m$ and $\hat{y} \in \mathcal{X}$, let

$$\hat{p}_{[Y^n,\hat{Y}^n]}^{(m)}(y^m,\hat{y}) \triangleq \frac{\left|\left\{m_o+1\leq i\leq n-m_o : \left(Y_{i-m_o}^{i+m_o},\hat{Y}_i\right) = (y^m,\hat{y})\right\}\right|}{n-m+1}. \tag{13}$$

In other words, $\hat{p}_{[Y^n,\hat{Y}^n]}^{(m)}(y^m,\hat{y})$ counts the fraction of times we observe the block y^m in Y^n ($Y_{i-m_o}^{i+m_o} = y^m$), while the position in $\hat{Y}^n$ corresponding to the middle symbol y_{m_o+1} is equal to $\hat{y}$ ($\hat{Y}_i = \hat{y}$). After finding the empirical distribution, the output of the denoiser is generated through the "postprocessing" or "derandomization" process as follows:

$$\hat{X}_i = \arg\min_{\hat{x}\in\hat{\mathcal{X}}}\sum_{x\in\mathcal{X}}\hat{p}_{[Y^n,\hat{Y}^n]}^{(m)}\left(Y_{i-m_o}^{i+m_o},x\right)d(\hat{x},x) \tag{14}$$

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 [8], the argument being roughly as follows: The rate-distortion function of the noisy signal $\mathbf{Y}$ under the defined difference distortion measure satisfies the Shannon lower bound with equality. It is proved in [8] that for such sources, for a fixed $\ell > 0$, the ℓ^{th} order empirical joint distribution between the source block and its quantized version, i.e.,

$$\hat{p}_{[Y^n,\hat{Y}^n]}^{(\ell)}(y^\ell,\hat{y}^\ell) \triangleq \frac{\left|\left\{1\leq i\leq n-\ell+1 : \left(Y_i^{i+\ell-1},\hat{Y}_i^{i+\ell-1}\right) = (y^\ell,\hat{y}^\ell)\right\}\right|}{n-\ell+1}$$

converges to the unique joint distribution that achieves the minimum mutual information in the ℓ^{th} order (informational) rate-distortion function of the source. In other words, $\hat{p}_{[Y^n,\hat{Y}^n]}^{(\ell)} \overset{d}{\to} q^{(\ell)}$, where

$$q^{(\ell)} = \arg\min_{q(y^\ell,\hat{y}^\ell):\mathrm{E}_q\left[d_\ell(Y^\ell,\hat{Y}^\ell)\right]\leq D} I(Y^\ell;\hat{Y}^\ell)$$

It turns out that in quantizing the noisy signal at distortion level $H(Z)$ under the distortion measure defined in (12), $q^{(\ell)}$ is equal to the ℓ^{th} order joint distribution between the source and the noisy signal [8]. Hence, the count vector $\hat{p}_{[Y^n,\hat{Y}^n]}^{(m)}(y^m,\hat{y})$ defined in (13) asymptotically converges to $p_{X_i|Y^n}$, which is what the optimal denoiser would base its decision on. After estimating $p_{X_i|Y^n}$, the postprocessing 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 III is applicable to any distortion measure. Hence, combining the MCMC-based lossy compressor and the described postprocessing operation yields a universal denoiser for our additive white noise setting.

Let $-\alpha(\mathbf{Y},H(Z)) < 0$ denote the slope of the unique point on the rate distortion curve of the process $\mathbf{Y} = \{X_i + Z_i\}_{i=1}^\infty$

[2] 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.

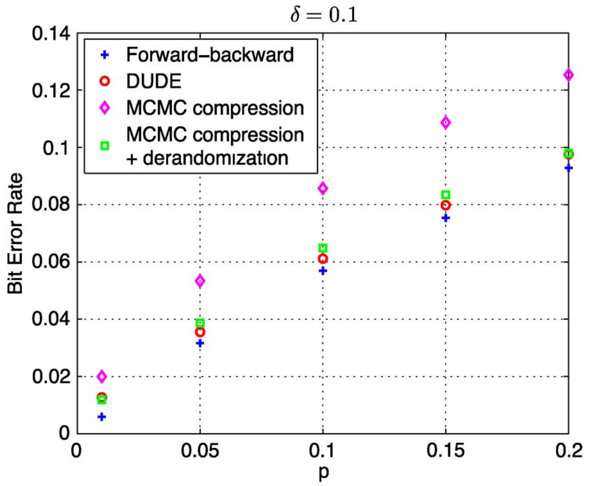


Fig. 1. Comparing the bit error rates of the denoisers derived from MCMC compression plus postprocessing (square markers), MCMC compression without postprocessing (diamond markers), the DUDE (circle markers) and the optimal nonuniversal Bayesian denoiser (+ markers). The source is a BSMS(p), and the channel is a DMC with error probability $\delta = 0.1$. The DUDE parameters are: $k_{\rm letf} = k_{\rm right} = 4$, and the MCMC compressor uses $\alpha = 0.95 : 0.3 : 2.15$, $\gamma = 0.75$, $\beta_t = (\frac{1}{\gamma})^{\lceil t/n \rceil}$, $r = 10n$, $n = 10^4$, and $k = 7$. The derandomization window length is $2 \times 4 + 1 = 9$.

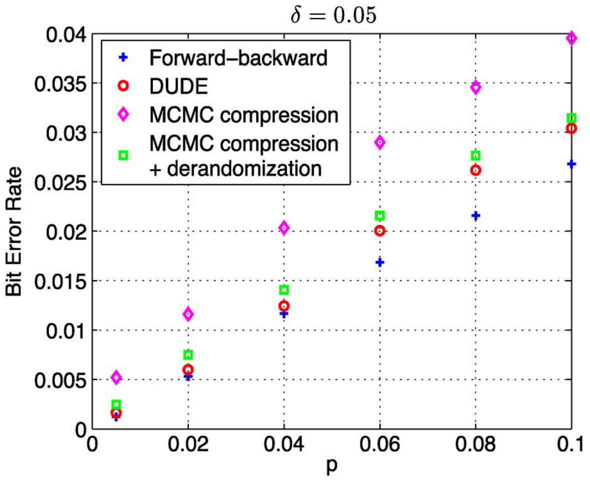


Fig. 2. Comparing the bit error rates of the denoisers derived from MCMC compression plus postprocessing (square markers), MCMC compression without postprocessing (diamond markers), the DUDE (circle markers) and the optimal nonuniversal Bayesian denoiser (+ markers). Here $\delta = 0.05$, and $\alpha = 0.9 : 0.3 : 2.4$. The rest of the parameters are identical to the setup of Fig. 1.

corresponding to the distortion level $D = H(Z)$, under the distortion measure defined in (12). Furthermore, let $k = k_n$ and $\boldsymbol{\beta} = \{\beta_t\}_t$ 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 Δ is defined in (11). Combining the results from [8] and [14] yields the following theorem, which proves the asymptotic optimality of the proposed denoising scheme.

Theorem 1: For any stationary ergodic process $\mathbf{X}$

$$\lim_{m\to\infty} \lim_{n\to\infty} \lim_{r\to\infty} \mathrm{E}\left[d_n(X^n, \hat{X}^n)\right] = L^{\mathrm{opt},B}(\mathbf{X}, \pi)$$

where $\hat{X}^n = \hat{X}^n(Y^n, \hat{Y}^n_{\alpha(\mathbf{Y},H(Z)),r}(Y^n))$ is generated by (13) and (14), and $\hat{Y}^n_{\alpha(\mathbf{Y},H(Z)),r}(Y^n)$ is the output of Algorithm 1. Moreover, for each source distribution, there exists a deterministic sequence $\{m_n\}_n$, $m_n \to \infty$, such that

$$\lim_{n\to\infty} \lim_{r\to\infty} \mathrm{E}[d_n(X^n, \hat{X}^n)] = L^{\mathrm{opt},B}(\mathbf{X}, \pi).$$

Remark 1: The main difficulty is in choosing the parameter α corresponding to the distortion level of interest, i.e., $\alpha(\mathbf{Y}, H(Z))$. To find the right slope, we run the quantization MCMC-based quantization part of the algorithm independently for two different initial slopes α_1 and α_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 α that would have resulted in the desired distortion, and then run the algorithm again from this starting point, compute the average distortion, and find a better estimate of α. After a few iterations of this process, we have a reasonable estimate of the desired α. Note that, for finding α, it is not necessary to work with the whole noisy signal, and one can consider only a long enough section of data first, estimate α from it, and then run the MCMC-based denoising algorithm on the whole noisy signal with the estimated parameter α. The outlined method for finding α is similar to what is done in [25] for finding an appropriate Lagrange multiplier.

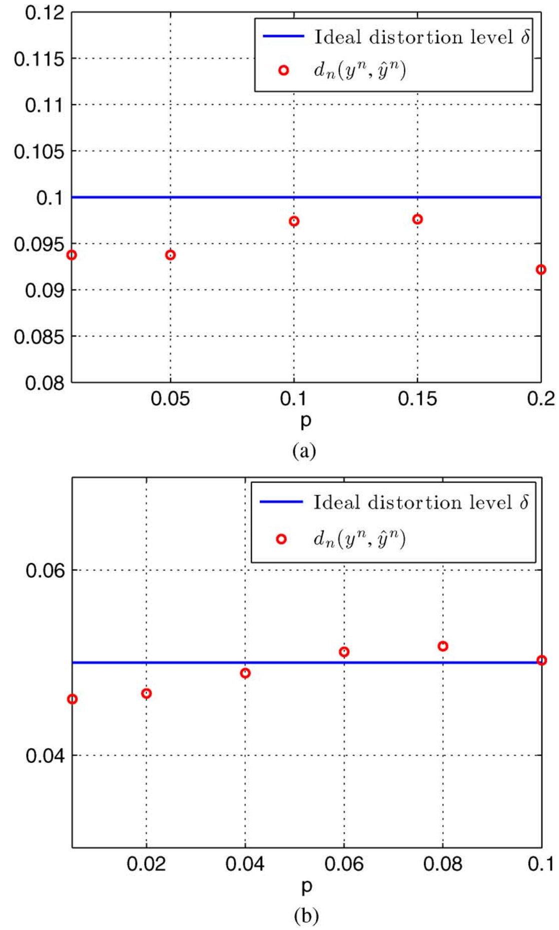
Remark 2: The deterministic sequence mentioned in Theorem 1 may depend on the source as well.

Discussion: Our proposed approach, MCMC coding and derandomization, is an alternative not only to the DUDE, but also to MCMC-based denoising schemes that have been based on or inspired by the Geman brothers' work [16]. While algorithmically our 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 a performance which is good according to the posterior distribution of the noise-free given the noisy data, but as would be induced by the rather arbitrary prior model placed on the data. In our case no assumptions, beyond ergodicity, about the distribution/model of the noise-free data are made, and optimum performance is guaranteed (in the appropriate limits).

V. Simulation Results

In this section, we compare the performance of the proposed denoising algorithm to that of the DUDE. As mentioned earlier, the 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 the first example, consider a binary symmetric Markov source (BSMS) with transition probability $p = 0.2$. The source sequence X^n is corrupted by a binary discrete memoryless channel (DMC) with error probability δ. Figs. 1 and 2 compare the performances of the DUDE and the new MCMC-based denoising algorithm, for the cases of $\delta = 0.1$ and $\delta = 0.05$, respectively. In each figure, we have plotted the average bit



(a)

(b)

Fig. 3. Comparing the average BER performance of the MCMC-based lossy compressor applied to the noisy signal versus p with the optimal BER which is δ. The simulations setups here are those of Fig. 1 and Fig. 2, respectively.

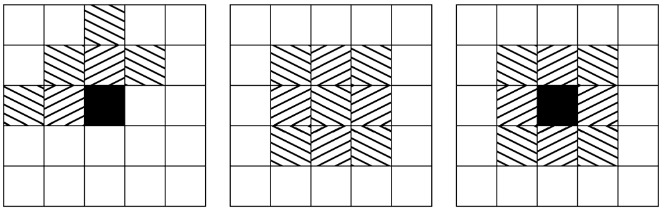
error rate (BER) of each algorithm over $N = 50$ simulations versus the transition probability p. Also, for the sake of comparison, we have added to each figure the performance of the forward-backward dynamic programming denoiser which achieves the optimum performance in recovering the described source from its noisy version, in the nonuniversal setup. In both figures the blocklength is $n = 10^4$, and the parameters of the MCMC compressors are chosen as follows: $\gamma = 0.75$, $\beta_t = (\frac{1}{\gamma})^{\lceil t/n \rceil}$, $r = 10n$, and $k = 7$.[3] It can be observed that in both figures, the performance of the proposed compression-based denoiser is very close to the performance of the DUDE.

In each case, the slope α is chosen such that the expected distortion between the noisy image and its quantized version using Algorithm 1 with distortion measure

$$\rho(x, \hat{x}) = \begin{cases} -\log \delta, & \text{if } x \neq \hat{x} \\ -\log(1-\delta), & \text{if } x = \hat{x} \end{cases}$$

is close to $H(Z)$. Note that

$$\mathrm{E}\left[\rho(X, \hat{X})\right] = -\mathrm{P}(X \neq \hat{X}) \log \delta - \mathrm{P}(X = \hat{X}) \log(1-\delta)$$



(a) (b) (c)

Fig. 4. Contexts used by the MCMC compressor DUDE and the derandomizer. (a) The sixth-order context used by the 2-D MCMC-based lossy compressor. (b) The derandomization block used in MCMC-based denoising. (c) The eighth-order context used by DUDE.

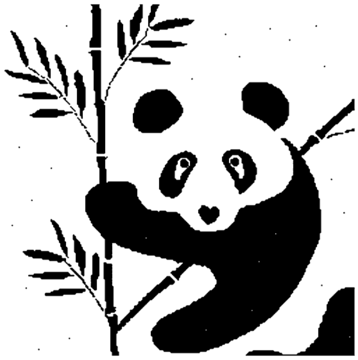


Fig. 5. Panda image.

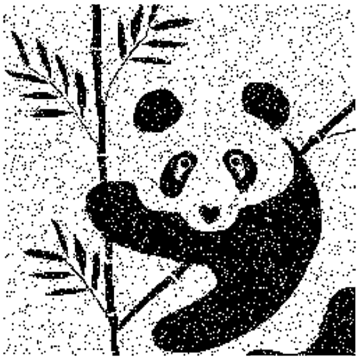


Fig. 6. Panda image corrupted by a DMC with error probability $\delta = 0.05$

which is equal to $H(Z) = -\delta \log \delta - (1-\delta) \log(1-\delta)$ when $\mathrm{P}(X \neq \hat{X}) = \delta$. Hence, we require our MCMC lossy encoder to compress the noisy signal under the Hamming distortion at $D = \delta$. Fig. 3 shows the average Hamming distortion, i.e., BER, of the MCMC-based lossy compressor versus p, for the cases of $\delta = 0.05$ and $\delta = 0.1$. In both cases, the average distortion incurred by the lossy compressor is close to its desired value which is δ.

In another example, we consider denoising the 256×256 binary image shown in Fig. 5. Fig. 6 shows its noisy version which is generated by passing the original image through a binary DMC with error probability of 0.05, i.e., $\pi(1) = 1 - \pi(0) = 0.05$. Fig. 7 shows the reconstructed image generated by the DUDE and Fig. 8(b) depicts the reconstructed image using the described algorithm. Here, the number of pixels is $n = 256^2$. In this experiment the DUDE context structure is set as Fig. 4(c). The 2-D MCMC coder employs the context shown in Fig. 4(a), and the derandomization block is chosen as Fig. 4(b). Note that

[3] A discussion on the selection of these parameters is presented in [26].

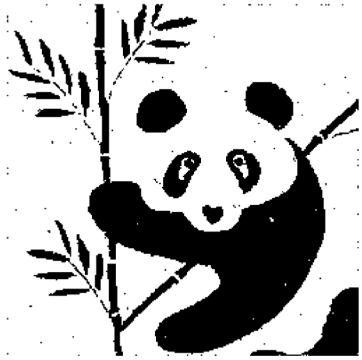
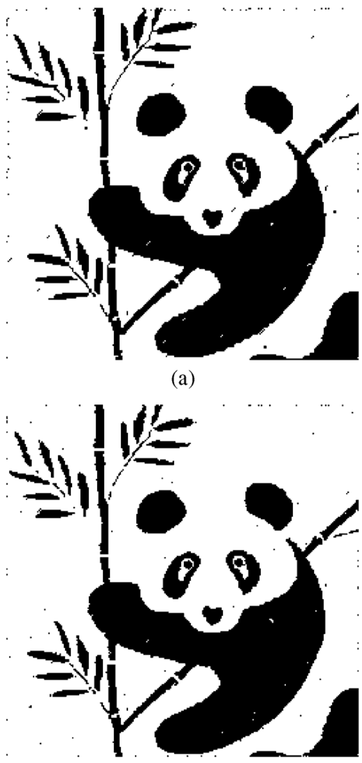


Fig. 7. The denoised image generated by the DUDE: $d_n(x^n, \hat{x}^n) = 7.97 \times 10^{-3}$.



(a)

(b)

Fig. 8. The MCMC-based denoiser applied to a binary image. (a) The denoised image generated by the MCMC compressor: $d_n(x^n, \hat{x}^n) = 1.01 \times 10^{-2}$. ($\alpha = 3.5, \beta_t = (\frac{1}{\gamma})^{\lceil t/n \rceil}, \gamma = 0.99$, and $r = 8n$). (b) The denoised image generated by the MCMC compressor plus derandomization: $d_n(x^n, \hat{x}^n) = 8.11 \times 10^{-3}$.

while we require the context used in computing the conditional empirical entropy of the image to have a causal structure, i.e., only contain pixels located prior to the current pixel, for the derandomization block we have no such constraint. While the former is used to measure the complexity of the signal, the latter is used for learning the joint distribution between the noisy signal and its quantized version.

The BERs of the DUDE and compression-based denoising algorithms are 8.17×10^{-3} and 7.59×10^{-3}, respectively. Though the performance of DUDE here is slightly better in terms of BER, the visual quality of the reconstruction is arguably better with the new denoiser, and the "texture" of the original image seems to be better preserved with our reconstruction. This may be a result of the fact that the compression-based approach is guaranteed of recovering not only the marginal distributions of one noise-free symbol given the noisy data, as in the DUDE, but in fact that k-dimensional distributions, for every k.

VI. Conclusion

The idea of deriving a universal denoising algorithm based on a universal lossy compression scheme with some postprocessing was proposed in [8]. However, this result has not yet been tested in practice. In this paper, we employed the MCMC-based universal lossy compression algorithm proposed in [14] to derive a universal denoising algorithm. The algorithm first applies the MCMC-based lossy compression algorithm to the noisy signal, using a distortion measure and level dictated by the distribution of the channel noise. Then it performs some simple postprocessing operations on the compressed noisy signal. Our simulation results show that the performance of the resulting denoising algorithm is promising and comparable to the performance of the DUDE. This shows that in practical situations compression-based denoising algorithms can be quite effective.

References

[1] F. Alajaji, "Feedback does not increase the capacity of discrete channels with additive noise," *IEEE Trans. Inf. Theory*, vol. 41, no. 2, pp. 546–549, Mar. 1995.

[2] S. Itzkovitz, T. Tlusty, and U. Alon, "Coding limits on the number of transcription factors," *BMC Genomics*, vol. 7, no. 1, p. 239, 2006.

[3] L. E. Baum, T. Petrie, G. Soules, and N. Weiss, "A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains," *Ann. Math. Statist.*, vol. 41, no. 1, pp. 164–171, 1970.

[4] R. Chang and J. Hancock, "On receiver structures for channels having memory," *IEEE Trans. Inf. Theory*, vol. 12, no. 4, pp. 463–468, Oct. 1966.

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

[6] D. Donoho, The Kolmogorov Sampler Jan. 2002.

[7] T. Cover and J. Thomas, *Elements of Information Theory*, 2nd ed. New York: Wiley, 2006.

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

[9] B. K. Natarajan, "Filtering random noise via data compression," in *Data Compress. Conf.*, 1993, pp. 60–69.

[10] B. Natarajan, K. Konstantinides, and C. Herley, "Occam filters for stochastic sources with application to digital images," *IEEE Trans. Signal Process.* , vol. 46, no. 5, pp. 1434–1438, May 1998.

[11] J. Rissanen, "MDL denoising," *IEEE Trans. Inf. Theory*, vol. 46, no. 7, pp. 2537–2543, Nov. 2000.

[12] S. P. Awate and R. T. Whitaker, "Unsupervised, information-theoretic, adaptive image filtering for image restoration," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 3, pp. 364–376, 2006.

[13] S. de Rooij and P. Vitanyi, "Approximating rate-distortion graphs of individual data: Experiments in lossy compression and denoising," *IEEE Trans. Comput.*, vol. 61, no. 3, Jul. 2011.

[14] S. Jalali and T. Weissman, "Rate-distortion via Markov chain Monte Carlo," in *Proc. IEEE Int. Symp. Inf. Theory*, Jul. 2008, pp. 852–856.

[15] E. Ordentlich, G. Seroussi, S. Verdú, M. Weinberger, and T. Weissman, "A discrete universal denoiser and its application to binary images," in *Proc. IEEE Int. Conf. on Image Process.*, Sep. 2003, pp. 117–120.

[16] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 6, no. 6, pp. 721–741, Nov. 1984.

[17] S. Kirkpatrick, C. D. Gelatt, Jr., and M. P. Vecchi, "Optimization by simulated annealing," *Sci.*, vol. 220, no. 4598, pp. 671–680, May 1983.

[18] V. Cerny, "Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm," *J. Optimiz. Theory Appl.*, vol. 45, no. 1, pp. 41–51, Jan. 1985.

[19] C. E. Shannon, "A mathematical theory of communication," *Bell Syst. Tech. J.*, vol. 27, pp. 379–423, 1948, 623-656.

[20] R. G. Gallager, *Information Theory and Reliable Communication*. New York: Wiley, 1968.

[21] T. Berger, *Rate-Distortion Theory: A Mathematical Basis for Data Compression*. Englewood Cliffs, NJ: Prentice-Hall, 1971.

[22] E. H. Yang, Z. Zhang, and T. Berger, "Fixed-slope universal lossy data compression," *IEEE Trans. Inf. Theory* , vol. 43, no. 5, pp. 1465–1476, Sep. 1997.

[23] J. Ziv and A. Lempel, "A universal algorithm for sequential data compression," *IEEE Trans. Inf. Theory*, vol. 23, no. 3, pp. 337–343, May 1977.

[24] J. Ziv and A. Lempel, "Compression of individual sequences via variable-rate coding," *IEEE Trans. Inf. Theory*, vol. 24, no. 5, pp. 530–536, Sep. 1978.

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

[26] S. Jalali and T. Weissman, Rate-Distortion via Markov Chain Monte Carlo arXiv:0808.4156v2.

Shirin Jalali received the B.Sc. and M.Sc. degrees in electrical engineering from Sharif University of Technology in 2002 and 2004, respectively. She received the M.Sc. degree in statistics and the Ph.D. degree in electrical engineering from Stanford University, Stanford, CA, in 2009.

Her research interests are in the areas of information theory and statistical signal processing. She is currently a postdoctoral fellow with the Center for Mathematics of Information (CMI), California Institute of Technology, Pasadena.

Tsachy Weissman (S'99–M'02–SM'07) received the B.Sc. degree (*summa cum laude*) in electrical engineering in 1997 and the Ph.D. degree in 2001, both from the Technion, Israel.

He then joined the Information Theory Group, Hewlett-Packard Laboratories, until joining Stanford University, Stanford, CA, where he has been on the faculty of the Electrical Engineering Department since 2003, and currently holds the STMicroelectronics Chair in the School of Engineering. He spent academic years 2007–2009 on leave at the Technion. His research is focused on information theory, statistical signal processing, the interplay between them, and their applications.

Dr. Weissman received the NSF CAREER award, a joint IT/COM societies Best Paper award, a Horev Fellowship for Leaders in Science and Technology, and a Henry Taub Prize for Excellence in Research. He is on the editorial board of the IEEE Transactions on Information Theory, serving as Associate Editor for Shannon Theory.