Algorithms for Discrete Denoising Under Channel Uncertainty

George M. Gemelos*	Styrmir Sigurjonsson	Tsachy Weissman
Packard 251	Packard 251	Packard 256
350 Serra Mall	350 Serra Mall	350 Serra Mall
Stanford, CA 94305	Stanford, CA 94305	Stanford, CA 94305
(650) 723-4544	(650) 723-4544	(650) 736-1448
ggemelos@stanford.edu	styrmir@stanford.edu	tsachy@stanford.edu

EDICS: 2-ESTM

Abstract

The goal of a denoising algorithm is to reconstruct a signal from its noise-corrupted observations. Perfect reconstruction is seldom possible and performance is measured under a given fidelity criterion. In a recent work, the authors addressed the problem of denoising unknown discrete signals corrupted by a discrete memoryless channel when the channel, rather than being completely known, is only known to lie in some uncertainty set of possible channels. A (sequence of) denoiser(s) was derived for this case and shown to be asymptotically optimal with respect to a worst-case criterion argued most relevant to this setting. In the present work we address the implementation and complexity of this denoiser, establishing its practicality. We also present empirical results suggesting the potential of these schemes to do well in practice. A key component of our schemes is an estimator of the subset of channels in the uncertainty set that are feasible in the sense of giving rise to the noisy signal statistics for some noiseless signal distribution. We establish the efficiency of this estimator (theoretically, algorithmically, and experimentally). We also present a modification of the recently developed discrete universal denoiser (DUDE) that assumes a channel based on the said estimator, and show that, in practice, the resulting scheme often performs comparably to our asymptotically minimax schemes. For concreteness, we focus on the binary alphabet case, but also discuss the extensions of the algorithms to general finite alphabets.

Index Terms

Channel Uncertainty, convex optimization, denoising algorithms, discrete universal denoising, image denoising, minimax schemes.

I. Introduction

In [9] it was shown that optimum denoising of a finite-alphabet process corrupted by a discrete memoryless channel (DMC) whose associated matrix is invertible can be achieved, asymptotically, without knowledge of the source statistics provided the channel is known. In [4] the problem where, in addition to the lack of knowledge of the source statistics, there is also uncertainty in the channel characteristics was addressed. Motivation for the setting of channel uncertainty is also discussed in [4]. The main focus of this paper is on algorithms for

The work of the first two authors was supported by MURI Grant DAAD-19-99-1-0215 and NSF Grant CCR-0311633. The work of the third author was supported in part by NSF Grant CCR-0312839.

implementing the denoising schemes suggested in [4]. A key component in these algorithms is a problem of independent interest: estimating the set of channels in the channel uncertainty set that are “feasible” in the sense of giving rise to the noisy signal distribution when corrupting some noiseless source. We shall focus for concreteness on the case of binary alphabets, since this is enough to capture the essence of the problem while minimizing cumbersome notation. We shall demonstrate, however, how the algorithms extend to the case of non-binary alphabets. We shall also make some observations regarding fundamental differences between the denoisers suggested in [4] and the discrete universal denoiser (DUDE) of [9].

In the setting of denoising for a known channel [9] there is a one to one correspondence between the channel output distribution and its input distribution. Our present setting is fundamentally different in that, given a noisy signal distribution, there may be many (input process, channel) pairs that can give rise to it. As an example consider the binary symmetric channel (BSC). If the output process is observed to be a Bernoulli(α) process, the (input process, channel) combinations producing that output distribution can be, e.g., both a Bernoulli(0) process going through a BSC with crossover probability α and a Bernoulli(α) process going through a noise-free BSC (with crossover probability 0), along with uncountably many other (input,channel) combinations. This is a fundamental difference which renders the task of attaining the performance of the optimum non-universal Bayesian scheme impossible, even for a genie-aided scheme with complete knowledge of the noisy signal statistics. This point is elaborated on in [4].

It has thus been argued in [4] that, under these circumstances, given any noisy source, a natural criterion under which the performance of a denoising scheme should be judged is its worst case performance under all (source, channel) pairs consistent with the given noisy source distribution. This clearly bounds the attainable worst case performance since, in our universal setting, even the noisy source distribution is not a priori known.

A family of denoisers that are universally optimal in the sense of asymptotically minimizing the said worst case performance are presented in [4]. The main contribution of the present work is the development of low complexity algorithms for implementing these denoising rules.

When considering the binary alphabet case we shall assume the setting of a binary signal with an unknown distribution, corrupted by a BSC with a crossover probability δ known to satisfy $\delta < 1/2$, but otherwise unknown. As will be detailed, our work in the binary alphabet setting conveys the essence of the implementation of the scheme for an arbitrary finite alphabet and an invertible channel matrix with parameters only known to lie in a given uncertainty set.

After setting up some notation in Section II, we turn in Section III to the binary minimax setting, where the goal is to minimize the maximum expected fraction of errors made by the denoiser over all source-channel pairs consistent with the noisy source distribution. In this section we will present the denoiser suggested in [4], as well as some of its performance guarantees. We will then present our algorithm for efficiently implementing the suggested denoisier, as well as a method for incorporating side information about the channel uncertainty set into the denoiser.

In Section IV we discuss methods for estimating the set of "feasible" channels, i.e., those channels in the uncertainty set that can give rise to the active noisy source distribution when corrupting some noiseless source. In particular, we present a low complexity algorithm to efficiently estimate this set, and compare its performance to a scheme suggested in [9]. This comparison is continued in Section V, where we report the results of several simulations comparing the performance of our denoiser with various versions of the DUDE of [9]. These simulations include both one- and two-dimensionally indexed data. In Section VI we demonstrate ways to extend the algorithms of Sections III and IV, which were developed for the binary case, to the more general case of non-binary alphabets and naturally structured channel uncertainty sets. In Section VII we show that the $\mathrm{min\text{-}max}$ in our problem setting is not equivalent to the $\mathrm{max\text{-}min}$, implying that our denoisers are in general not optimal for the worst feasible channel. We conclude in Section VIII with a summary of our results.

II. Definitions and Notation

We assume that the components of the noise-free signal, the observation and the reconstruction signal take their values in the same finite alphabet $\mathcal{A}$. Let $\mathcal{W}(\mathcal{A} \to \mathcal{A})$ denote the set of all channels with input and output alphabet $\mathcal{A}$. We identify an element $\Pi \in \mathcal{W}(\mathcal{A} \to \mathcal{A})$ with a $|\mathcal{A}| \times |\mathcal{A}|$ stochastic matrix, $\Pi(a, b)$ denoting the probability of a symbol b at the channel output when the channel input is a. Let $\mathcal{M}(\mathcal{A})$ denote the simplex of probability distributions on $\mathcal{A}$ and $\mathcal{M}(\mathcal{A}^{2k+1})$ denote the simplex of distributions on $(2k+1)$-tuples with $\mathcal{A}$-valued components. Denote by $\Lambda : \mathcal{A} \times \mathcal{A} \to \mathbb{R}$ a given loss function where $\Lambda(x, \hat{x})$ is the loss incurred when estimating the symbol x with the symbol $\hat{x}$. A randomized n-block denoiser $\hat{X}^n$ is a mapping $\hat{X}^n : \mathcal{A}^n \to \mathcal{M}(\mathcal{A})^n$, i.e. upon seeing the noisy signal $z^n \in \mathcal{A}^n$, the i-th reconstruction symbol is $a \in \mathcal{A}$ with probability $\hat{X}^n_{[i]}(z^n)[a]$, where $\hat{X}^n_{[i]}(z^n) \in \mathcal{M}(\mathcal{A})$ denotes the i-th component of $\hat{X}^n(z^n)$ and $\hat{X}^n_{[i]}(z^n)[a]$ denotes the probability that it assigns to $a \in \mathcal{A}$.

Uppercase letters will denote random quantities while lower case letters denote deterministic values. Bold notation will indicate doubly infinite sequences e.g. $\mathbf{X} = (\ldots, X_{-1}, X_0, X_1, \ldots)$. Also, X_i^j denotes the sequence $(X_i, \ldots, X_j)$, omitting the subscript when $i = 1$.

For $x^n \in \mathcal{A}^n$ and $z^n \in \mathcal{A}^n$ we denote

$$L_{\hat{X}^n}(x^n, z^n) = \frac{1}{n} \sum_{i=1}^{n} \sum_{a \in \mathcal{A}} \Lambda(x_i, a) \hat{X}^n_{[i]}(z^n)[a]. \tag{1}$$

In words $L_{\hat{X}^n}(x^n, z^n)$ is the expected normalized cumulative loss of the denoiser $\hat{X}^n$ on the individual sequence pair (x^n, z^n), the expectation with respect to the randomization of the denoiser.

With the exception of Section VI, we consider processes with binary alphabets i.e. $\mathcal{A} = \{0, 1\}$, corrupted by a BSC and take $\Lambda(\cdot, \cdot)$ to be the Hamming loss function. In this case a n-block denoiser $\hat{X}^n$ is a mapping $\hat{X}^n : \{0,1\}^n \to [0,1]^n$, $\hat{X}^n_{[i]}(z^n) \in [0, 1]$ denoting the probability that the i-th reconstruction symbol is a 1 when the denoiser observes z^n. For $x^n \in \{0,1\}^n$ and $z^n \in \{0,1\}^n$ (1) becomes

$$L_{\hat{X}^n}(x^n, z^n) = \frac{1}{n} \sum_{i=1}^{n} \left| x_i - \hat{X}^n_{[i]}(z^n) \right|. \tag{2}$$

We similarily denote for $f \in M_k := \{f : \mathcal{A}^{2k+1} \to \mathcal{M}(\mathcal{A})\}$

$$L_f(x^n, z^n) = \frac{1}{n} \sum_{i=1}^{n} \left| x_i - f(z_{i-k}^{i+k}) \right|, \tag{3}$$

namely, the expected normalized loss when employing the sliding-window denoiser defined by $\hat{X}^n_{[i]}(z^n) = f(z_{i-k}^{i+k})$.

III. The MiniMax Criterion

Our setting assumes uncertainty in the channel characteristics. We quantify this uncertainty by assuming that we are *given* a set of channels which includes the true channel that corrupted the clean source of interest. In the binary setting we assume the uncertainty set to consist of BSCs. The set is parameterized by the crossover probability and, for the sake of simplicity, we assume it is of the form $[0,\mathcal{U}]$, $\mathcal{U} < 1/2$.

For a stationary ergodic source $P_{\mathbf{Z}}$ let

$$\mathcal{C}(P_{\mathbf{Z}}) = \{0 \leq \delta \leq 1/2 : \exists P_{\mathbf{X}} \text{ s.t. } P_{\mathbf{X}} * \delta = P_{\mathbf{Z}}\}, \tag{4}$$

where $P_{\mathbf{X}} * \delta$ denotes the output distribution of a BSC(δ) whose input process has distribution $P_{\mathbf{X}}$. Thus when the output process has distribution $P_{\mathbf{Z}}$ the possible (source,channel) pairs can be described by the following set, $\{(P_{\mathbf{X}}, \delta) : P_{\mathbf{X}} * \delta = P_{\mathbf{Z}}, \delta \in \mathcal{C}(P_{\mathbf{Z}})\}$. Further we let

$$\Gamma(P_{\mathbf{Z}}) = \max_{\mathcal{C}(P_{\mathbf{Z}})} \delta. \tag{5}$$

It is easy to see that $\mathcal{C}(P_{\mathbf{Z}}) = [0, \Gamma(P_{\mathbf{Z}})]$ (since if $\delta \in \mathcal{C}(P_{\mathbf{Z}})$ then $\delta' \in \mathcal{C}(P_{\mathbf{Z}})$ for any $\delta' < \delta$). Combining this with the given uncertainty set we define,

$$\Delta(P_{\mathbf{Z}}) = [0,\mathcal{U}] \cap [0, \Gamma(P_{\mathbf{Z}})], \tag{6}$$

which is the set of all channels in the uncertainty set that can give rise to $P_{\mathbf{Z}}$ when corrupting some noiseless source. Note that Δ only depends on $P_{\mathbf{Z}}$ through Γ and this dependence will often be omitted.

For an n-block denoiser $\hat{X}^n$ let

$$\mathcal{L}^{(n)}_{\hat{X}^n}(P_{\mathbf{Z}}, \Delta, \mathbf{Z}) = \sup_{\{(P_{\mathbf{X}},\delta): P_{\mathbf{X}} * \delta = P_{\mathbf{Z}}, \delta \in \Delta(P_{\mathbf{Z}}) \cap \mathbb{Q}\}} E_{P_{\mathbf{X}},\delta}\left[L_{\hat{X}^n}(X^n, Z^n) | \mathbf{Z}\right], \tag{7}$$

where $E_{P_{\mathbf{X}},\delta}[\cdot|\mathbf{Z}]$ denotes (a version of) the conditional expectation (conditioning on $\mathbf{Z}$) under the joint distribution on the noiseless and noisy sources induced when the noiseless source $\sim P_{\mathbf{X}}$ is corrupted by a BSC(δ).

$\mathcal{L}^{(n)}_{\hat{X}^n}(P_{\mathbf{Z}}, \Delta, \mathbf{Z})$ is thus the worst-case performance of the denoiser $\hat{X}^n$ over all channels that can give rise to $P_{\mathbf{Z}}$, as assessed by a "genie" that both knows $P_{\mathbf{Z}}$ and sees the whole noisy realization $\mathbf{Z}$. In the definition of $\mathcal{L}^{(n)}_{\hat{X}^n}(P_{\mathbf{Z}}, \Delta, \mathbf{Z})$ we take the sup over the set of all channels in Δ intersected with the rational numbers, $\mathbb{Q}$. This is to guarantee that the sup is over a countable set and thus that $\mathcal{L}^{(n)}_{\hat{X}^n}(P_{\mathbf{Z}}, \Delta, \mathbf{Z})$ is a well-defined random variable.

Define now

$$\mu^{(n)}_k(P_{\mathbf{Z}}, \Delta, \mathbf{Z}) = \min_{f \in M_k} \mathcal{L}^{(n)}_f(P_{\mathbf{Z}}, \Delta, \mathbf{Z}), \tag{8}$$

where $\mathcal{L}^{(n)}_f(P_{\mathbf{Z}}, \Delta, \mathbf{Z})$ stands for $\mathcal{L}^{(n)}_{\hat{X}^n}(P_{\mathbf{Z}}, \Delta, \mathbf{Z})$ when $\hat{X}^n$ is the sliding window denoiser that uses f for its denoising rule. Let further

$$\mu_k(P_{\mathbf{Z}}, \Delta, \mathbf{Z}) = \limsup_{n \to \infty} \mu^{(n)}_k(P_{\mathbf{Z}}, \Delta, \mathbf{Z}), \tag{9}$$

and finally let the "sliding-window minimum loss" be defined by

$$\mu(P_{\mathbf{Z}}, \Delta, \mathbf{Z}) = \lim_{k \to \infty} \mu_k(P_{\mathbf{Z}}, \Delta, \mathbf{Z}), \tag{10}$$

where the limit exists since $\mu_k(P_{\mathbf{Z}}, \mathbf{Z})$ is clearly non-increasing with k.

A. Construction of a Universal Scheme

For $\alpha \in [0, 1]$, $\delta < 1/2$, $d_0 \in [0, 1]$ and $d_1 \in [0, 1]$ let

$$F(\alpha, \delta, d_0, d_1) = \frac{1}{1 - 2\delta} \left[(1 - \delta - \alpha)(1 - \delta)d_0 + (1 - \delta - \alpha)\delta d_1 + (\alpha - \delta)\delta(1 - d_0) + (\alpha - \delta)(1 - \delta)(1 - d_1)\right]. \tag{11}$$

In the single observation problem, for $\alpha \in [\delta, 1 - \delta]$, $F(\alpha, \delta, d_0, d_1)$ is the expected loss of a denoising scheme which says 1 with probability d_0 upon observing a zero and says 1 with probability d_1 upon observing a one, when observing a Bernoulli($(\alpha - \delta)/(1 - 2\delta)$) corrupted by a BSC($\delta$) (so that the channel output is a Bernoulli(α)). For a probability distribution on binary $(2k + 1)$-tuples $P_{Z^k_{-k}}$ and $f : \{0, 1\}^{2k+1} \to [0, 1]$ we define the functional G_k by

$$G_k\left(P_{Z^k_{-k}}, \delta, f\right) = \sum_{c_k} F\left(P_{Z^k_{-k}}(Z_0 = 1|c_k), \delta, f(c_k, 0), f(c_k, 1)\right) P_{Z^k_{-k}}(c_k), \tag{12}$$

where the summation is over all 2^{2k} kth-order double-sided contexts c_k, with slight abuse of notation we let (c_k, a) denote the $2k+1$-tuple formed by the context c_k with the symbol a in the middle, $P_{Z_{-k}^k}(Z_0 = 1|c_k)$ denotes $\Pr(Z_0 = 1|(Z_{-k}^{-1}, Z_1^k) = c_k)$ (under the source $P_{Z_{-k}^k}$), and $P_{Z_{-k}^k}(c_k)$ is shorthand for $\Pr((Z_{-k}^{-1}, Z_1^k) = c_k)$ (under the source $P_{Z_{-k}^k}$). We further define J_k by

$$J_k\left(P_{Z_{-k}^k}, \Delta, f\right) = \max_{\delta \in \Delta} G_k\left(P_{Z_{-k}^k}, \delta, f\right), \tag{13}$$

and

$$f_{\mathbb{MM}_k}\left(P_{Z_{-k}^k}, \Delta\right) = \arg \min_{f:\{0,1\}^{2k+1} \to [0,1]^{2k+1}} J_k\left(P_{Z_{-k}^k}, \Delta, f\right), \tag{14}$$

selecting an arbitrary achiever when it is not unique. In our setting $P_{\mathbf{Z}}$ is not known and thus neither is $\Gamma(P_{\mathbf{Z}})$ (recall (5) for its definition), so we need to estimate it. If we knew the distribution of the $2l+1$-tuple $P_{Z_{-l}^l}$ (induced by $P_{\mathbf{Z}}$) we could evaluate the following upper bound on $\Gamma(P_{\mathbf{Z}})$:

$$\Gamma_l\left(P_{Z_{-l}^l}\right) = \max\left\{0 \le \delta \le 1/2 : \exists P_{X_{-l}^l} \text{ s.t. } P_{X_{-l}^l} * \delta = P_{Z_{-l}^l}\right\}. \tag{15}$$

Instead, we shall use the empirical distribution of a $2l+1$-tuple induced by the noisy observation sequence. An efficient algorithm for calculating $\Gamma_k(\cdot)$ is presented in section IV. We further define now

$$\Delta_l(P_{Z_{-l}^l}) = [0, \mathcal{U}] \cap [0, \Gamma_l(P_{Z_{-l}^l})]. \tag{16}$$

Now let $\hat{Q}^{2k+1}[z^n]$ denote the $(2k+1)$-th order empirical distribution induced by z^n and let $\hat{X}^{n,k}$ denote the n-block denoiser defined by

$$\hat{X}^{n,k,l}(z^n)[i] = f_{\mathbb{MM}_k}\left(\hat{Q}^{2k+1}[z^n], \hat{\Delta}_l(z^n)\right)[z_{i-k}^{i+k}] \quad k+1 \le i \le n-k, \tag{17}$$

where, with slight abuse of notation we write $\hat{\Delta}_l(z^n)$ as shorthand for $\Delta_l\left(\hat{Q}^{2l+1}[z^n]\right)$. $\hat{\Delta}_l(z^n)$ is our estimate of the feasible set, i.e., the set of channel crossover probabilities belonging to the uncertainty set that can give rise to the noisy source distribution. The denoiser $\hat{X}^{n,k,l}(z^n)[i]$ can be arbitrarily defined for i outside the range $k+1 \le i \le n-k$.

B. Performance Guarantee

We shall now cite one of the results of [4] that provides a sound performance guarantee and therefore justification for the use of the schemes considered in this work. For a sequence $\{\psi_i\}$ of non-negative reals let $\mathcal{M}_s^{\mathcal{A}}(\{\psi_i\})$ denote the set of stationary distributions whose i-th ψ-mixing coefficient is upper bounded by ψ_i (cf., e.g., [2], [4] for the definition of ψ-mixing coefficients). Most sources arising in practice, such as Markov sources with no restricted sequences and hidden Markov processes with no restricted state sequences can be shown to have exponentially diminishing ψ-mixing coefficients [2].

Theorem 1 (Corollary 6 of [4]): Let $\{\psi_i\}$ be a sequence of non-negative reals with $\psi_i \to 0$. There exists an unbounded sequence $\{w_n\}$ such that if $l_n = k_n \leq w_n$ and $\hat{X}^n_{univ} = \hat{X}^{n,k_n,l_n}$ (where $\hat{X}^{n,k,l}$ was defined in (17)), then for any $P_{\mathbf{Z}} \in \mathcal{M}_s^{\mathcal{A}}(\{\psi_i\})$, for any sequence of sets Δ_n, with $\Delta_n \subset [0, \Delta]$ and $|\Delta_n| \leq e^{\sqrt{n}}$,

$$\limsup_{n\to\infty} \left[\mathcal{L}_{\hat{X}^n_{univ}}(P_{\mathbf{Z}}, \Delta_n, \mathbf{Z}) - \mu_{k_n}^{(n)}(P_{\mathbf{Z}}, \Delta, \mathbf{Z}) \right] \leq 0 \qquad P_{\mathbf{Z}} - a.s. \tag{18}$$

In words, the sequence of denoisers $\{\hat{X}^n_{univ}\}$, as the block length n increases, perform (under our worst-case performance criterion) at least as well as the best sliding-window denoiser (of any order since k_n can be chosen to increase without bound).

C. Efficient Computation of MiniMax Denoiser

As is evident from (17), the "engine" at the heart of our denoisers is the calculation of $f_{\mathbb{MM}_k}[P_{Z_{-k}^k}, \Delta]$ (defined in (14)), which we now consider. By observing that (11) can be simplified into a function that is quadratic in δ (the maximizing argument) and whose coefficients depend on the minimizing argument, (12) can be written as,

$$G_k\left(P_{Z_{-k}^k}, \delta, f\right) = \frac{A(P_{Z_{-k}^k}, f)\delta^2 + B(P_{Z_{-k}^k}, f)\delta + C(P_{Z_{-k}^k}, f)}{1 - 2\delta}, \tag{19}$$

where,

$$A(P_{Z_{-k}^k}, f) = \sum_{c_k} \left(2f(c_k, 0) - 2f(c_k, 1)\right) P_{Z_{-k}^k}(c_k) \tag{20a}$$

$$B(P_{Z_{-k}^k}, f) = -1 + \sum_{c_k} \left(2f(c_k, 1) - 2f(c_k, 0)\right) P_{Z_{-k}^k}(c_k) \tag{20b}$$

$$C(P_{Z_{-k}^k}, f) = \sum_{c_k} \left(\eta_{c_k} + f(c_k, 0) - \eta_{c_k} f(c_k, 0) - \eta_{c_k} f(c_k, 1)\right) P_{Z_{-k}^k}(c_k), \tag{20c}$$

and $\eta_{c_k} = P_{Z_{-k}^k}(Z_0 = 1|c_k)$. From now on we will suppress A, B and C's dependence on $P_{Z_{-k}^k}$ and f. Note that $B = -A - 1$ and we can thus eliminate B from the equations. It is easily shown that (19) is either convex or concave for fixed A and C. For the cases it is convex we need to solve,

$$\frac{\partial G_k\left(P_{Z_{-k}^k}, \delta, f\right)}{\partial \delta} = 0. \tag{21}$$

The solutions to (21) are $\delta' = (A + \sqrt{-A^2 - 2A + 4AC})/(2A)$ and $\delta'' = (A - \sqrt{-A^2 - 2A + 4AC})/(2A)$. Therefore to find the maximum in (13) we have to consider no more than four values for δ, the endpoints of $[0, \Delta]$, $\delta = \delta'$ and $\delta = \delta''$. Also note that δ' and δ'' only have to be considered if they are feasible, i.e., if they lie in $[0, \Delta]$. Thus the maximum value of (13) over $0 \leq \delta \leq \Delta$ is equivalent to

$$\max\left\{C, \frac{A\Delta^2 - A\Delta - \Delta + C}{1 - 2\Delta}, \frac{A\delta'^2 - A\delta' - \delta' + C}{1 - 2\delta'}\mathbf{1}_{\delta' \in (0,\Delta)}, \frac{A\delta''^2 - A\delta'' - \delta'' + C}{1 - 2\delta''}\mathbf{1}_{\delta'' \in (0,\Delta)}\right\}, \tag{22}$$

where $\mathbf{1}$ denotes the indicator function. To minimize (22) we observe that A and C are linear functions in the minimizing argument, f, which implies that the range of A and C is convex since the range of f is convex. Also note that δ' and δ'' are continous functions of (A, C).

Consider now the following modification of (22)

$$\max\left\{C, \frac{A\Delta^2 - A\Delta - \Delta + C}{1 - 2\Delta}, \frac{A\delta'^2 - A\delta' - \delta' + C}{1 - 2\delta'}\mathbf{1}_{\delta' \in (0,\Delta)}\right\}. \tag{23}$$

Let $S(\Delta) = \{(A, C): \ \delta' \in (0, \Delta)\}$, which is a convex set. Also define $V(\Delta) = \{(A, C): \ C \leq 1/2(A(1 - \Delta) + 1)\}$ which is the region where

$$C \geq \frac{A\Delta^2 - A\Delta - \Delta + C}{1 - 2\Delta}.$$

Then define $S_1(\Delta) = S(\Delta)^c \cap V(\Delta)$ and $S_2(\Delta) = S(\Delta)^c \cap V^c(\Delta)$. Note that $\{S(\Delta), S_1(\Delta), S_2(\Delta)\}$ is a partition of the range of (A, C) for all $\Delta \in (0, 1/2)$. Since (23) is non-negative we can rewrite it as,

$$C\mathbf{1}_{S_1(\Delta)} + \frac{A\Delta^2 - A\Delta - \Delta + C}{1 - 2\Delta}\mathbf{1}_{S_2(\Delta)}$$
$$+ \max\left\{C\mathbf{1}_{S(\Delta)}, \frac{A\Delta^2 - A\Delta - \Delta + C}{1 - 2\Delta}\mathbf{1}_{S(\Delta)}, \frac{A\delta'^2 - A\delta' - \delta' + C}{1 - 2\delta'}\mathbf{1}_{S(\Delta)}\right\}.$$

Consider the terms within the $\max$. The regions where any two of them intersect is defined by the following linear equations:

$$C = \frac{A(1-\Delta)+1}{2} \tag{24a}$$

$$C = \frac{A+1}{2} \tag{24b}$$

$$C = \left(\frac{1}{2} + \Delta^2 - \Delta\right) A + \frac{1}{2}. \tag{24c}$$

Since all three terms are continuous on $S(\Delta)$, in $S(\Delta)$ the maximizing term can only be exchanged on one of the lines defined in (24). Therefore to find the maximizing term in $S(\Delta)$ we only need to test one point in each region in the partition of $S(\Delta)$ defined by (24). It is easy to evaluate the expressions for a few points and see that the maximizing term on $S(\Delta)$ is

$$\frac{A\delta'^2 - A\delta' - \delta' + C}{1-2\delta'}.$$

We can therefore rewrite (23) as

$$C\mathbf{1}_{S_1(\Delta)} + \frac{A\Delta^2 - A\Delta - \Delta + C}{1-2\Delta}\mathbf{1}_{S_2(\Delta)} + \frac{A\delta'^2 - A\delta' - \delta' + C}{1-2\delta'}\mathbf{1}_{S(\Delta)}. \tag{25}$$

This leads to the following expression for (22):

$$\max\left\{C\mathbf{1}_{S_1(\Delta)} + \frac{A\Delta^2 - A\Delta - \Delta + C}{1-2\Delta}\mathbf{1}_{S_2(\Delta)} + \frac{A\delta'^2 - A\delta' - \delta' + C}{1-2\delta'}\mathbf{1}_{S(\Delta)}, \frac{A\delta''^2 - A\delta'' - \delta'' + C}{1-2\delta''}\mathbf{1}_{\delta''\in(0,\Delta)}\right\}. \tag{26}$$

Define $S''(\Delta) = \{(A, C) : \delta'' \in (0, \Delta)\}$ and rewrite (26) as

$$\begin{aligned}&\left(C\mathbf{1}_{S_1(\Delta)} + \frac{A\Delta^2 - A\Delta - \Delta + C}{1-2\Delta}\mathbf{1}_{S_2(\Delta)} + \frac{A\delta'^2 - A\delta' - \delta' + C}{1-2\delta'}\mathbf{1}_{S(\Delta)}\right)\mathbf{1}_{S''(\Delta)^c} + \\ &\max\left\{\left(C\mathbf{1}_{S_1(\Delta)} + \frac{A\Delta^2 - A\Delta - \Delta + C}{1-2\Delta}\mathbf{1}_{S_2(\Delta)} + \frac{A\delta'^2 - A\delta' - \delta' + C}{1-2\delta'}\mathbf{1}_{S(\Delta)}\right)\mathbf{1}_{S''(\Delta)},\right. \\ &\left.\frac{A\delta''^2 - A\delta'' - \delta'' + C}{1-2\delta''}\mathbf{1}_{S''(\Delta)}\right\}.\end{aligned} \tag{27}$$

Similarly to what we did for (22), since the two terms in the maximization are continuous on $S''(\Delta)$, we can look at the region where the two terms intersect. Note that the continuity of both terms in (A, C) follows from that of δ' and δ''. On $S''(\Delta)$ the two terms intersect only at $(24b)$ and $(24c)$. Therefore, continuity implies that the maximizing term can only be exchanged on those lines. By picking a point in each region of the partition of $S''(\Delta)$ defined by $(24b)$ and $(24c)$, we see that

$$C\mathbf{1}_{S_1(\Delta)} + \frac{A\Delta^2 - A\Delta - \Delta + C}{1-2\Delta}\mathbf{1}_{S_2(\Delta)} + \frac{A\delta'^2 - A\delta' - \delta' + C}{1-2\delta'}\mathbf{1}_{S(\Delta)}$$

is the maximizing term. Hence from (27) we see that we can rewrite (22) as

$$C\mathbf{1}_{S_1(\Delta)} + \frac{A\Delta^2 - A\Delta - \Delta + C}{1-2\Delta}\mathbf{1}_{S_2(\Delta)} + \frac{A\delta'^2 - A\delta' - \delta' + C}{1-2\delta'}\mathbf{1}_{S(\Delta)}. \tag{28}$$

Observe that since the range of (A, C) is convex and the equations in (24) are linear, $S(\Delta)$, $S_1(\Delta)$ and $S_2(\Delta)$ are convex sets. Also note that the first two terms in (28) are linear in (A, C) and hence convex.

Claim 1:

$$\frac{A\delta'^2 - A\delta' - \delta' + C}{1-2\delta'} \tag{29}$$

is convex on $S(\Delta)$ for all $\Delta \in (0, 1/2)$.

Proof: The eigenvalues of the Hessian of (29) on the set $S(\Delta)$ are

$$\left\{0, \frac{-A(A+2-4C)\sqrt{4A^2+1+4C^2-4C}}{2A(A+2-4C)^2}\right\}.$$

It is easily shown that the non-trivial eigenvalue is non-zero for all feasible (A, C) and therefore by continuity to determine whether the Hessian is positive semidefinite on $S(\Delta)$ we only have to evaluate it at a single point. Picking any $(A, B, C) \in S(\Delta)$, we find that the Hessian is positive semidefinite and, therefore, (29) is convex on $S(\Delta)$. $\square$

Hence the alternative expression (28) tells us that minimizing (22) is equivalent to finding the minimum of three separate convex functions each over a disjoint convex set. This can easily be done my minimizing each term in (22) over its appropriate partition, which can be carried out using efficient convex optimization algorithms. In particular, we have implemented the denoiser using the log-barrier method with gradient descent (see [1] for a detailed discussion of the log barrier method). After the three minimizations are carried out it remains only to select the minimum between the three values. Overall, this gives an efficient and concrete way of calculating $f_{\mathbb{MM}_k}[P_{Z_{-k}^k}, \Delta]$.

IV. Largest Crossover Probability Consistent with Observations

Recall the definitions of $\Gamma(P_{\mathbf{Z}})$ and $\Gamma_l(P_{Z_{-l}^l})$ (equations (5) and (15)). We shall also use $\Gamma_l(P_{\mathbf{Z}})$ to denote Γ_l evaluated for the $2l+1$-th order marginal of $P_{\mathbf{Z}}$. We shall say that $\delta < 1/2$ is l-feasible if there exists a distribution on a noise-free tuple $P_{X_{-l}^l}$ that gives rise to $P_{Z_{-l}^l}$ when corrupted by a BSC(δ). Thus, Γ_l is the maximum l-feasible δ. It is easy to show that for any stationary ergodic $P_{\mathbf{Z}}$, $\Gamma_l(P_{\mathbf{Z}})$ is non-increasing in l and that

$$\Gamma(P_{\mathbf{Z}}) = \lim_{l\to+\infty} \Gamma_l(P_{\mathbf{Z}}) = \inf_l \Gamma_l(P_{\mathbf{Z}}). \tag{30}$$

In our setting $P_{\mathbf{Z}}$ is unknown and, consequently, so is $\Gamma(P_{\mathbf{Z}})$. In this section we develop an efficient algorithm to estimate $\Gamma(P_{\mathbf{Z}})$ as a function of the empirical distribution $\hat{Q}^{2l+1}[z^n]$. Note that this estimate, $\Gamma_l(\hat{Q}^{2l+1}[z^n])$, or $\hat{\Gamma}_l(z^n)$ for short, is the one implicitly employed also by our denoiser in (17), since the estimate of the channel uncertainty set that it employs $\hat{\Delta}_l$ is taken as the intersection between the a priori channel uncertainty set $[0,\mathcal{U}]$ and the estimated feasible set $[0,\hat{\Gamma}_l]$.

In [4] it is shown that under mild conditions there exists an unbounded sequence l_n such that

$$\lim_{n\to\infty} \hat{\Gamma}_{l_n}(Z^n) = \Gamma(P_{\mathbf{Z}}) \qquad P_{\mathbf{Z}} - a.s. \tag{31}$$

In [9, Subsection 8-C] a method for obtaining an upper bound on $\Gamma_l(P_{\mathbf{Z}})$ is suggested. Let $\{C_j^{(l)}\}_{j=1}^{2^l}$ denote all the 2^l binary sequences. The basic idea is to look at the $\min_{i,j}\varphi_{i,j}^{(2l)}$ where,

$$\varphi_{i,j}^{(2l)} \triangleq P\left(Z_0 = 1 | Z_{-1}^{-l} = C_i^{(l)}, Z_1^l = C_j^{(l)}\right) \qquad \forall i,j, \tag{32}$$

the conditional probability on the right side being the empirical one induced by the data. It is clear that $\min_{i,j}\varphi_{i,j}^{(2l)}$ (for n large enough so that the empirical distribution is close enough to the true one) is an upper bound to Γ_l. The following is a numerical example illustrating that the method of [9] yields, in general, upper bounds that are not tight, while $\hat{\Gamma}_l$ is guaranteed by (31) to converge to the true value.

Example 1: Let $P_{\mathbf{Z}}$ be the first order symmetric Markov Process with transition matrix

$$\begin{pmatrix} 0.695 & 0.305 \\ 0.305 & 0.695 \end{pmatrix}$$

passed through a $BSC(.1)$. For this case clearly $\Gamma(P_{\mathbf{Z}}) \geq 0.1$ and, in fact, the inequality can be shown to be strict [6]. So, in particular, $\Gamma_l(P_{\mathbf{Z}}) > 0.1$ for all l. Simulations yield (with high precision and confidence) $\min_{i,j} \varphi_j^{(2)} = 0.2629$, $\min_{i,j} \varphi_j^{(4)} = 0.2440$, $\min_{i,j} \varphi_j^{(6)} = 0.2415$, and $\min_{i,j} \varphi_j^{(8)} = .2398$. On the same simulated data one finds $\hat{\Gamma}_1(z^n) = .1757$, $\hat{\Gamma}_2(z^n) = .1451$, $\hat{\Gamma}_3(z^n) = .1266$, and $\hat{\Gamma}_4(z^n) = .1107$,

A. Efficient Computation of $\hat{\Gamma}_l(z^n)$

Our estimate of Γ_l simply evaluates $\Gamma_l(\cdot)$ at the acquired empirical distribution. As we now show, this is a simple calculation. Given a stationary $P_{\mathbf{Z}}$, $\delta \in (0, 1/2)$ and l, if δ is l-feasible there exists a corresponding stationary input process $\mathbf{X}$ which when passed through a $BSC(\delta)$ yields $P_{\mathbf{Z}}^{(l)}$. Therefore, letting $\{C_j^{(l)}\}_{j=1}^{2^l}$ denote the set of binary l-tuples, we can define and rewrite

$$\begin{aligned} \beta_j^{(l)} &\triangleq P(Z_1 = 1 | Z_0^{(-l+1)} = C_j^{(l)}) \\ &= \frac{\sum_{l=1}^{2^l} (\delta * \alpha_l^{(k)}) P(x_0^{-l+1} = C_l^{(l)}) \prod_{i=-l}^{-1} m(x_i, z_i)}{P(Z_0^{-l+1} = C_j^{(l)})} \end{aligned} \tag{33}$$

where $*$ denotes binary convolution (defined as $p * q = (1-p)q + p(1-q)$), m is the channel transition probability and

$$\alpha_j^{(l)} \triangleq P(X_1 = 1 | X_{-l+1}^0 = C_j) \qquad \forall j.$$

For simplicity define,

$$\gamma_j^{(l)} \triangleq P(X_{-l+1}^0 = C_j)$$

and

$$\Theta_j^{(l)} \triangleq P(Z_{-l+1}^0 = C_j)$$

and let γ and Θ be the associated length 2^l column vectors. Let $M(\delta)$ denote the channel matrix. Then,

$$\Theta^T = \gamma^T M(\delta)^{\otimes l}$$

where $M(\delta)^{\otimes l}$ denotes the l^{th} tensor product of the matrix $M(\delta)$. Since $\delta < 1/2$, $M(\delta)^{-1}$ exists and we have

$$\gamma = (M(\delta)^{-T})^{\otimes l}\Theta.$$

Using this we can write

$$\beta_j^{(l)} = \frac{\sum_{l=1}^{2^l}(\alpha_l^{(l)} * \delta)[(M(\delta)^{-T})^{\otimes l}\Theta]_l \Gamma_{l,j}}{\Theta_j},$$

where

$$\Psi = M(\delta)^{\otimes l}.$$

We can simplify the summation using vector notation. Dropping subscripts to indicate the corresponding vector gives

$$\beta_j^{(l)}\Theta_j = (\alpha^{(l)} * \delta)^T \left[(M(\delta)^{-T})^{\otimes l}\Theta\right] \odot \Psi_j,$$

where $\odot$ denotes componentwise multiplication and Ψ_j is the j^{th} column vector of Ψ. We can simplify further to obtain

$$\beta^{(l)} \odot \Theta = \Psi^T \left[\left((M(\delta)^{-T})^{\otimes l}\Theta\right) \odot (\alpha * \delta)\right]$$

which, following standard algebraic manipulations gives

$$\alpha^{(l)} * \delta = \left[(M(\delta)^{-T})^{\otimes l}(\beta^{(l)} \odot \Theta)\right] \oslash \left[(M(\delta)^{-T})^{\otimes l}\Theta\right], \tag{34}$$

where $\oslash$ denotes componentwise division. We will also use the following easily verified identity

$$\alpha^{(l)} = \frac{(\alpha^{(l)} * \delta) - \delta\bar{\mathbf{1}}}{1 - 2\delta}, \tag{35}$$

where $\bar{\mathbf{1}}$ denotes the "all ones" column vector of appropriate dimension (in this case 2^l). Combining (34) and (35) gives an explicit expression for $\alpha^{(l)}$ in terms of $\beta^{(l)}$. Γ_l is nothing but the maximum value of $\delta \in [0, 1/2]$ such that all the component of $\alpha^{(l)}$ are in $[0, 1]$. Hence a given δ is l-feasible if and only if all the components of the associated $\alpha^{(l)}$ are in $[0, 1]$. With each iteration of this feasibility test, we can shrink the uncertainty in Γ_l by a factor of $1/2$. We can therefore quickly converge to the true Γ_l as well as give precision bounds for a fixed number of iterations.

V. Simulations and Experimentation

In this section we present experimental results obtained by implementation and employment of the scheme of Subsection III.A for the case of a binary signal corrupted by a BSC with an unknown crossover probability. We shall refer to the scheme in this case as the Minimax Binary Denoiser (MBD), which we implement using the methods presented earlier. We compare the performance of the MBD to that of DUDE from [9] on simulated sequences (one dimensional), simulated fields (two dimensional), and on a text image. Throughout the simulations the channel crossover probability is, a priori, only known to lie in $[0, 1/2)$.

A. A Modified DUDE

In [9, Section 8-C] the problem of denoising an unknown source, corrupted by an unkown discrete memoryless channel is considered. The algorithm suggested is to estimate the channel parameters and then to apply the DUDE assuming the channel estimate in lieu of the unknown channel parameters. The channel estimate suggested (as mentioned in Section IV) is,

$$\hat{\delta}_l = \min_j \min \left\{ \beta_j^{(l)}, 1 - \beta_j^{(l)} \right\}. \tag{36}$$

We refer to the application of the DUDE using a channel estimate as a Modified DUDE algorithm (M-DUDE). We propose an improvement to this algorithm: Rather than using the estimate in (36), which as argued in Section IV, in general, loosely upper bounds the largest feasible channel crossover probability, we suggest using the estimate $\hat{\Gamma}_l$ in (30), which, by 31, converges to the true upper bound.

B. One Dimensional Simulations

We implemented the MBD as discussed in section III for 1D sequences going through a BSC. As a source for our simulation we chose a hidden Markov source. To generate the source, a first-order symmetric binary Markov sequence with transition probability p was simulated and then sent through a simulated BSC with parameter δ_s. The hidden Markov source was then corrupted by a simulated BSC with parameter δ, see figure 1.

We then compare the performance of the MBD to that of the M-DUDE. We apply the M-DUDE with two channel parameters: the estimate in (36) (suggested in [9]) and our estimate from section IV, $\hat{\Gamma}_l$. To get an idea

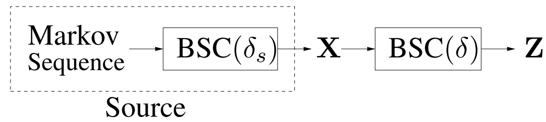

Fig. 1. The box diagram shows the process of creating the source, $\mathbf{X}$, and the output, $\mathbf{Z}$ after going through a BSC with transition probability δ.

of the optimum channel-dependent performance we also ran DUDE using the true value of the channel crossover probability δ. Tables I and II show the results of the simulations.

		Estimate of δ		Denoising BER			
δ_s	δ	$\hat{\Gamma}_4$	$\hat{\delta}_4$	M-DUDE($\hat{\Gamma}_4$)	M-DUDE($\hat{\delta}_4$)	DUDE	MBD
0.0600	0	0.0459	0.0719	0.0448	0.0531	0	0.0371
0.0492	0.0120	0.0383	0.0718	0.0334	0.0499	0.0121	0.0334
0.0378	0.0240	0.0493	0.0717	0.0417	0.0470	0.0240	0.0378
0.0259	0.0360	0.0356	0.0719	0.0358	0.0431	0.0358	0.0358
0.0133	0.0480	0.0425	0.0722	0.0369	0.0387	0.0365	0.0375
0.0067	0.0540	0.0401	0.0712	0.0369	0.0369	0.0355	0.0375

TABLE I

DENOISING USING A HIDDEN SYMMETRIC MARKOV SOURCE WHERE THE MARKOV SOURCE HAS TRANSITION PROBABILITY $p = 0.2$. HERE $\delta_s * \delta$ IS FIXED AT 0.06 (RESULTING IN THE SAME NOISY SIGNAL STATISTICS) AND $k = 2$.

		Estimate of δ		Denoise Performace			
δ_s	δ	$\hat{\Delta}_4$	$\hat{\delta}_4$	M-DUDE($\hat{\Gamma}_4$)	M-DUDE($\hat{\delta}_4$)	DUDE	MBD
0.2	0	0.197	0.221	0.122	0.184	0	0.114
0.174	0.04	0.186	0.220	0.130	0.183	0.0399	0.120
0.143	0.08	0.195	0.222	0.137	0.180	0.0804	0.131
0.105	0.12	0.194	0.223	0.140	0.174	0.120	0.136
0.0588	0.16	0.187	0.223	0.141	0.165	0.137	0.140
0	0.2	0.184	0.224	0.135	0.146	0.133	0.137

TABLE II

DENOISING USING A HIDDEN MARKOV SOURCE WHERE THE MARKOV SOURCE HAS TRANSITION PROBABILITY $p = 0.1$. HERE $\delta_s * \delta$ IS FIXED AT 0.2 AND $k = 2$.

We note that when both the source is unkwown, and there is channel uncertainty, there is a risk of injecting noise as the simulation results show. Where this is indeed the case we notice that the MBD injects substantially fewer errors than the M-DUDE, since the worst-case criterion it was designed to optimize leads to more conservative denoising. We also note that in a number of cases the performance of the MBD is actually comparable to that of the channel-dependent DUDE (shown in [9] to achieve optimum source- and channel-dependent performance).

Finally, we note that the M-DUDE($\hat{\Gamma}_4$) performs consistently better than M-DUDE($\hat{\delta}_4$), often performing comparably to the channel-dependent DUDE. This is due to the fact that $\hat{\Gamma}$ is a better (asymptotically consistent) estimate of the largest feasible δ, while $\hat{\delta}$ (even asymptotically) is in most cases a strict upper bound to it.

C. Two Dimensional Data

We begin with simulations similar to those of the previous section, except now on a two-dimensionally-indexed simulated process (image) as the source. We implemented the MBD for 2D sources as in [5], and compared it to a 2D implementation of the DUDE for a BSC as in [9]. As before, we ran the M-DUDE with the estimates suggested in [9] found in (36), and our estimate from section IV, $\hat{\Gamma}_l$. As in the previous section, for the source- and channel-dependent optimum performance benchmark we also ran DUDE using the true δ.

Here the context used for denoising consists of the 3×3 square while the 3 bits in the upper lefthand corner are used for calculating $\hat{\Gamma}_3$ and $\hat{\delta}_3$.

For our experimentation we use a hidden random field (HRF). First we generate a random field causally according to,

$$x_{i,j} = \begin{cases} N_{i,j}^{1/2} & \text{if} \quad i = 0 \text{ or } j = 0 \\ N_{i,j}^{1/2} & \text{if} \quad x_{i-1,j} \neq x_{i,j-1} \\ N_{i,j}^{\alpha} & \text{if} \quad x_{i-1,j} = x_{i,j-1} = 0 \\ N_{i,j}^{\bar{\alpha}} & \text{if} \quad x_{i-1,j} = x_{i,j-1} = 1, \end{cases} \tag{37}$$

where $x_{i,j}$ denotes the component at location (i, j) and $\{N_{i,j}^{1/2}\}$, $\{N_{i,j}^{\alpha}\}$ and $\{N_{i,j}^{\bar{\alpha}}\}$ are independent fields, consisting of independent components which are Bernoulli with parameters $1/2$, α, and $\bar{\alpha} = 1 - \alpha$, respectively. Figure 2 shows an example of a random field with $\alpha = 0.05$. We then corrupt this field by a BSC with transition probability δ_s and the output is then used as the noiseless image for the simulation, i.e., analogously as in figure 1, replacing the Markov sequence by a random field.

The hidden random field is then sent through a BSC with transition probability δ and denoising is performed. We used a test image of size 2000×2000. Tables III and IV show the bit error rate of the denoised image relative to the noiseless one.

The results show a trend similar to that observed for one-dimensional signals. We notice that the MBD consistently outperforms the M-DUDE($\hat{\delta}_3$) and is comparable in performance to the DUDE with the true channel

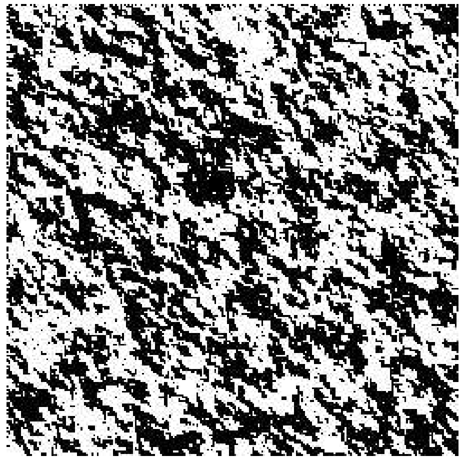

Fig. 2. An example of a random field with $\alpha = 0.05$.

		Estimate of δ		Denoise Performace			
δ_s	δ	$\hat{\Gamma}_3$	$\hat{\delta}_3$	M-DUDE($\hat{\Gamma}_3$)	M-DUDE($\hat{\delta}_3$)	DUDE	MBD
0	0.01	0.0184	0.0606	0.0115	0.0182	0.0095	0.0114
0	0.02	0.0279	0.0717	0.0169	0.0225	0.0169	0.0169
0	0.05	0.0575	0.105	0.0354	0.0457	0.0354	0.0354
0	0.10	0.106	0.160	0.0682	0.0696	0.0687	0.0691

TABLE III

DENOISING RESULTS FOR HRF WITH $\alpha = 0.05$

parameter δ, (shown to be optimal in [9]). Again, the M-DUDE($\hat{\Gamma}_3$) performs better than M-DUDE($\hat{\delta}_3$). As before, this is due to the fact that when the underlying data contains strong structure such as in the case of random fields, then $\hat{\Gamma}_l$ tends to be close to the true channel parameter δ. This fact suggests that the M-DUDE($\hat{\Gamma}_l$) would be a good algorithm for denoising natural images.

We next present denoising results for a binary text image. We scanned half a page of text at a resolution of 1000×600. In table V we show a piece (approx. $1/6$) of the original, noisy and denoised images for $\delta = 0.1$. For this particular case the DUDE using the true channel parameter δ had a normalized error rate of 0.0330, while the MBD had an error rate of 0.0479. The M-DUDE($\hat{\delta}_3$) did better than the MMD, with error rate close to that of the DUDE with the true channel parameter. This should be contrasted with Tables III and IV, where the MMD consistently did better than the M-DUDE($\hat{\delta}_3$). The high performance of the M-DUDE($\hat{\delta}_3$) is explained by the fact that we were denoising a text image. A text image is composed of mostly white background which allows

		Estimate of δ		Denoise Performace			
δ_s	δ	$\hat{\Gamma}_3$	$\hat{\delta}_3$	M-DUDE($\hat{\Gamma}_3$)	M-DUDE($\hat{\delta}_3$)	DUDE	MBD
0.0375	0.0135	0.0499	0.0640	0.0416	0.0428	0.0137	0.0268
0.0250	0.0263	0.0504	0.0639	0.0335	0.0337	0.0263	0.0628
0.0750	0.0294	0.101	0.119	0.0757	0.0836	0.0295	0.0542
0.0500	0.0556	0.1019	0.1187	0.0619	0.0677	0.0556	0.0550
0.0150	0.0052	0.0207	0.0313	0.0156	0.0163	0.0510	0.0112
0.010	0.0102	0.0202	0.0314	0.0122	0.0135	0.0102	0.0122

TABLE IV

DENOISING RESULTS FOR HRF WITH $\alpha = 0.01$

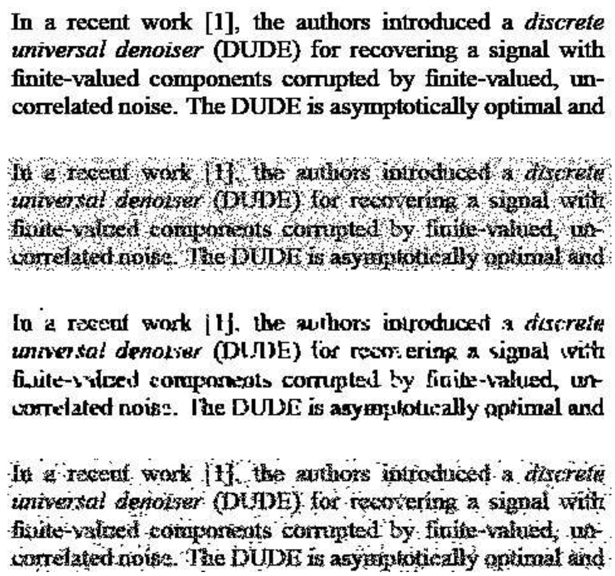

TABLE V

DENOISING OF A TEXT IMAGE. THE TOP IMAGE IS THE ORIGINAL IMAGE, NEXT IS THE NOISY VERSION WHERE $\delta = 0.1$, THEN A DENOISED VERSION USING THE DUDE WITH THE TRUE δ AND FINALLY A DENOISED VERSION USING MBD.

for any reasonable channel estimate to be highly accurate. Hence in this case the M-DUDE($\hat{\delta}_3$) is effectively implementing the DUDE, the optimal denoiser.

VI. BEYOND BINARY ALPHABETS

Up to this point we have confined ourselves to binary alphabets. Our goal in this section is to extend the algorithms developed in sections IIIC and IVA to the more general non-binary setting. In particular we will address the case of non-binary alphabets with channels parameterized by a single parameter. In other words, the uncertainty in the knowledge of the channel can be described by the uncertainty of a single parameter, analogously as in the binary case where the uncertainty was in the the crossover probability of the BSC (knowing that the crossover probability is less than $1/2$). We also require some structure in this free parameter, δ. We require that the channels be monotonic in δ, i.e., if δ_0 is admissible for a given $P_{\mathbf{Z}}$ then all $\delta \leq \delta_0$ are also admissible. Here we say that δ is admissible if there exists a source such that, when passed through the channel defined by δ, the output statistics agree with $P_{\mathbf{Z}}$.

Observe that with the above constraint, our definition of $\Gamma(P_{\mathbf{Z}})$, extended from the binary case in (5), becomes

$$\Gamma(P_{\mathbf{Z}}) = \max\left\{\delta : \exists P_{\mathbf{X}} \text{ s.t. } P_{\mathbf{X}} * Ch(\delta) = P_{\mathbf{Z}}\right\}, \tag{38}$$

where $P_{\mathbf{X}} * Ch(\delta)$ denotes the distribution of the output process of the channel with the parameter δ whose input process has distribution $P_{\mathbf{X}}$. We define for this setting a minimax performance benchmark, similarly to (10),

$$\mu(P_{\mathbf{Z}}, \mathbf{Z}) = \lim_{k\to\infty} \limsup_{n\to\infty} \min_{f\in M_k} \mathcal{L}_f^{(n)}(P_{\mathbf{Z}}, \mathbf{Z}), \tag{39}$$

where here $\mathcal{L}_f^{(n)}$ is the loss of the sliding window scheme f, extended from the binary setting.

By confining our channel uncertainty into a single parameter, we can preserve much of the structure from the binary alphabet case. With this structure intact we can adapt the algorithms from sections IIIC and IVA to the single parameterized non-binary alphabet case. This adaptation needs to be done on a case by case basis. In the following section we illustrate how this is done for a particular family of non-binary channels. We denote the alphabet by $\mathcal{A} = \{1, \ldots, M\}$, i.e. M is the alphabet size. The family of channels we will use for our example can be considered the M-ary generalization of the BSC.

A. The Symmetric Channel

Let the probability transition matrix of the noisy channel be given by

$$\begin{pmatrix} 1-\delta & \frac{\delta}{M-1} & \cdots & \frac{\delta}{M-1} \\ \frac{\delta}{M-1} & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \frac{\delta}{M-1} \\ \frac{\delta}{M-1} & \cdots & \frac{\delta}{M-1} & 1-\delta \end{pmatrix}.$$

It is readily verified that if a single variable $X \in \mathcal{A}$ is distributed as

$$P_X = \begin{pmatrix} \frac{(M-1)\alpha_1-\delta}{(M-1)-M\delta} \\ \vdots \\ \frac{(M-1)\alpha_{M-1}-\delta}{(M-1)-M\delta} \\ \frac{(M-1)-\delta-(M-1)\sum_{i=1}^{M-1}\alpha_i}{(M-1)-M\delta} \end{pmatrix}$$

is corrupted by the above channel then the channel output distribution is $P_Z = (\alpha_1, \ldots, \alpha_{M-1}, 1 - \sum_{i=1}^{M-1} \alpha_i)^T$. We represent a denoiser for this single-observation problem as $\{d_{i,j}\}_{i,j=1}^M$, with $d_{i,j}$ denoting the probability that the denoiser says j upon observing i. In particular $d_{i,j} \in [0,1]$ and $\sum_{j=1}^M d_{i,j} = 1 \ \forall i$. The expected loss of this denoiser (in the single observation problem) is

$$
\begin{aligned}
&F\left(\{\alpha_i\}_{i=1}^M, \delta, \{d_{i,j}\}_{i,j=1}^M\right) = \\
&\frac{(M-1)\alpha_1 - \delta}{(M-1) - M\delta}\left[(1-\delta)\sum_{j\neq 0} d_{0,j} + \frac{\delta}{M-1}\sum_{i\neq 0, j\neq 0} d_{i,j}\right] + \\
&\frac{(M-1)\alpha_2 - \delta}{(M-1) - M\delta}\left[(1-\delta)\sum_{j\neq 1} d_{1,j} + \frac{\delta}{M-1}\sum_{i\neq 1, j\neq 1} d_{i,j}\right] + \ldots + \\
&\frac{(M-1) - \delta - (M-1)\sum_{i=1}^{M-1}\alpha_i}{(M-1) - M\delta}\left[(1-\delta)\sum_{j\neq M} d_{M,j} + \frac{\delta}{M-1}\sum_{i\neq M, j\neq M} d_{i,j}\right] = \\
&\sum_{k=1}^{M-1}\frac{(M-1)\alpha_k - \delta}{(M-1) - M\delta}\left[(1-\delta)\sum_{j\neq k} d_{k,j} + \frac{\delta}{M-1}\sum_{i\neq k, j\neq k} d_{i,j}\right] + \\
&\frac{(M-1) - \delta - (M-1)\sum_{i=1}^{M-1}\alpha_i}{(M-1) - M\delta}\left[(1-\delta)\sum_{j\neq M} d_{M,j} + \frac{\delta}{M-1}\sum_{i\neq M, j\neq M} d_{i,j}\right].
\end{aligned} \tag{40}
$$

Consider now a probability distribution on M-ary $(2k+1)$-tuples $P_{U_{-k}^k}$. Consider a (randomized) denoiser for one symbol based on observing a noisy $(2k+1)$ tuple around it $f : \mathcal{A}^{2k+1} \to \mathcal{M}(\mathcal{A})$. Such a denoiser can be thought of as a collection of single-symbol denoisers, one for each context, where if the current context is c_k and we observe i as the middle symbol, the denoiser will change the symbol to j with probability $f(i, c_k)[j]$. We define the following functional

$$
G_k\left(P_{U_{-k}^k}, \delta, f\right) = \sum_{c_k} F\left(\left\{P_{U_{-k}^k}(U_0 = i|c_k)\right\}_{i=1}^M, \delta, f(c_k)\right) P_{U_{-k}^k}(c_k), \tag{41}
$$

where analogously as in the binary case the summation is over all M^{2k} kth-order double sided contexts, $P_{U_{-k}^k}(U_0 = i|c_k)$ denotes $\Pr(U_0 = i|(U_{-k}^{-1}, U_1^k) = c_k)$ (under the source $P_{U_{-k}^k}$), and $P_{U_{-k}^k}(c_k)$ is shorthand for $\Pr((U_{-k}^{-1}, U_1^k) = c_k)$ (under the source $P_{U_{-k}^k}$). We define

$$
f_{MM_k}\left[P_{U_{-k}^k}, \Delta\right] = \arg\min_f \max_{0\le\delta\le\Delta} G_k\left(P_{U_{-k}^k}, \delta, f\right), \tag{42}
$$

selecting an arbitrary achiever when it is not unique. Let $\hat{X}^{n,k,l}$ denote the n-block denoiser defined by,

$$\hat{X}^{n,k,l}(z^n)[i] = f_{\mathbb{MM}_k}\left(\hat{Q}^{2k+1}[z^n], \hat{\Delta}_l(z^n)\right)[z_{i-k}^{i+k}] \quad k+1 \leq i \leq n-k, \tag{43}$$

B. Algorithms for the Symmetric Channel

Now that we have examined the general symmetric channel, we want to extend the algorithms from sections IIIC and IVA to this more general case. We observe first that the single parameter structure discussed earlier exists in the case of the symmetric channel. Combined with the invertibility of the channel matrix, this single parameter structure is all we need to adapt the algorithm in section IVA to the generalized symmetric channel case. The same algebraic construction found in section IVA follows with the modification that

$$\beta_{j,i}^{(k)} \triangleq P(Z_1 = i | Z_0^{(-k+1)} = C_j^{(k)}) \tag{44}$$

$$\alpha_{j,i}^{(k)} \triangleq P(X_1 = i | X_0^{(-k+1)} = C_j^{(k)}). \tag{45}$$

Hence the channel described by a particular δ is k-feasible if and only if each row of the associated matrix $\alpha^{(k)}$ is an element of the M-dimensional simplex. Therefore we have an algorithm that not only converges to Δ_k, but also produces bounds with each iteration.

Now that we have extended the algorithm from section IVA, we turn our attention to solving (42). As in (13) we define J_k by

$$J_k\left(P_{U_{-k}^k}, \Delta, f\right) = \max_{0 \leq \delta \leq \Delta} G_k\left(P_{U_{-k}^k}, \delta, f\right). \tag{46}$$

Similarly to the analysis in Section IIIC, from (40) and (41) it follows that

$$G_k\left(P_{U_{-k}^k}, \delta, f\right) = \frac{A\delta^2 + B\delta + C}{M - 1 - M\delta}$$

where

$$A(P_{U^k_{-k}}, f) = \frac{1}{M-1-M\delta} \sum_{c_k} \sum_{k=1}^{M-1} \left(\sum_{\substack{j=1 \\ j\neq k}}^{M} f(k,c_k)[j] - \frac{1}{M-1} \sum_{\substack{i=1 \\ i\neq k}}^{M} \sum_{\substack{j=1 \\ j\neq k}}^{M} f(i,c_k)[j] \right), \tag{47}$$

$$B(P_{U^k_{-k}}, f) = \frac{1}{M-1-M\delta} \sum_{c_k} \left[\sum_{k=1}^{M-1} \left(\sum_{\substack{j=1 \\ j\neq k}}^{M} (-(M-1)\alpha_k - 1) f(k,c_k)[j] + \frac{(M-1)\alpha_k}{M-1} \sum_{\substack{i=1 \\ i\neq k}}^{M} \sum_{\substack{j=1 \\ j\neq k}}^{M} f(i,c_k)[j] \right) \right. \tag{48}$$

$$\left. + \frac{M-1-\delta-(M-1)\sum_{i=1}^{M}\alpha_i}{M-1-M\delta} \left(\frac{\delta}{M-1} \sum_{\substack{i=1 \\ j=1}}^{M} f(i,c_k)[j] - \sum_{j=1}^{M} f(M,c_k)[j] \right) \right] \tag{49}$$

$$C(P_{U^k_{-k}}, f) = \frac{M-1}{M-1-M\delta} \sum_{c_k} \sum_{k=1}^{M-1} \sum_{\substack{j=1 \\ j\neq k}}^{M} \alpha_k f(k,c_k)[j] + \frac{M-1-\delta-(M-1)\sum_{i=1}^{M}\alpha_i}{M-1-M\delta} \sum_{j=1}^{M-1} f(M,c_k)[j]. \tag{50}$$

Therefore J_k can be expressed as

$$\max\left\{ C, \frac{A\Delta^2 + B\Delta + C}{M-1-M\Delta}, \frac{A\delta'^2 + B\delta' + C}{M-1-M\delta'} \mathbf{1}_{\delta'\in(0,\Delta)}, \frac{A\delta''^2 + B\delta'' + C}{M-1-M\delta''} \mathbf{1}_{\delta''\in(0,\Delta)} \right\}, \tag{51}$$

where

$$\delta' = \frac{AM - A + \sqrt{A(AM^2 - 2AM + A + BM^2 - BM - CM^2)}}{AM} \quad \text{and}$$
$$\delta'' = \frac{AM - A - \sqrt{A(AM^2 - 2AM + A + BM^2 - BM - CM^2)}}{AM}.$$

Hence, as in Section IIIC, J_k simplifies to the $\max$ between four points which are simple functions of the coefficients A, B and C. Hence, for a given denoiser f, the quantity $J_k(f)$ is easily calculated.

The analysis for the minimization of J_k for the simple binary case in section IIIC is quite involved. Certain subtleties in the analysis suggest that the general M-array symmetric channel may not be piecewise convex or that finding the boundaries of the convex regions may be overly complicated. Hence this analysis needs to be carried out and verified for the particular alphabet size at hand.

Since we cannot simply extend the algorithm from section IIIC to the general M-array symmetric channel case without analysis for each M of interest, is there something that can be done in general? Earlier it was shown that for a given denoiser f we can easily calculate $J_k(f)$. This suggests that although it might not be possible to find the absolute minimum of J_k, we can apply methods such as simulated annealing to estimate the absolute minimum, see [7] and [8] for a detailed discussion of simulated annealing over a continuous domain. Both the simulated annealing methods discussed in [7] and [8] are concerned with unconstrained minimization. Since we are dealing with the constrained minimization of J_k, we once again will need to make use of the log barrier method as described in [1]. The benefit of using simulated annealing is that one can estimate the minimax optimal denoiser and have some control over the complexity versus accuracy of the estimation. The control over the trade off comes from controling the annealing schedule.

We have therefore managed to extend the algorithm from Subsection IVA to the general M-ary symmetric channel case. We have also shown that for a particular M, it is possible to extend the algorithm from Subsection IIIC to the general M-ary symmetric channel case, and that even if one cannot use the convex optimization methods developed in Subsection IIIC, estimates of the minimax optimal denoiser can still be obtained using simulated annealing. Hence, in practice, for any M, one can apply the minimax denoiser for the general M-ary symmetric channel.

C. Beyond Symmetric Channels

The analysis and methods used in Subsections VIA and VIB can be extended to many other families of channels. With use of simulated annealing, see [7] and [8], the above methods can be applied to any family of channels with the proper single parameter parameterization and with easily calculated expressions for J_k. This significantly extends the possible applications of the minimax denoiser to the non-binary case.

VII. MINIMAX $\neq$ MAXIMIN

A natural question arising in the context of our minimax criterion is whether it coincides with the maximin. An affirmative answer would imply that a minimax optimal scheme is an optimal scheme for the least denoisable source-channel pair consistent with the output distribution. This, in turn, would suggest that employing the DUDE of [9] tailored for the least denoisable pair (which is easy to estimate) would give rise to a universally minimax optimal and practical scheme.

Unfortunately, as we now show, the minimax does not coincide with the maximin in our problem. Specifically, we shall argue that for some noisy sources minimax > maximin. To show this we assume the input process, $P_{\mathbf{X}}$ is Bernoulli p, $p < 1/2$, and is corrupted by a $BSC(\delta)$ channel. We assume that the channel crossover probability, δ, belongs to an uncertainty set $\mathcal{U}$, which is given to us. In particular assume that $\mathcal{U} = \{0, 0.01, 0.02, \ldots, 0.5\}$. As a performance measure we take the unconditional measure used in Corollary 5 from [4], namely,

$$\min_{f \in M_k} \max_{\{(P_{\mathbf{X}}, \delta): \delta \in \mathcal{U}, P_{\mathbf{X}} * \delta = P_{\mathbf{Z}}\}} E_{[P_{\mathbf{X}}, \delta]}[L_f(X^n, Z^n)]. \tag{52}$$

With this setup the output process, $P_{\mathbf{Z}}$, will be Bernoulli with parameter $p * \delta$ where $\delta \in \mathcal{U}$ and we can assume it is known to us. In such a case, there are only two optimal schemes, as well as mixtures of the two. Depending on the channel δ, the optimal scheme either says what it sees ($\delta < p$) or says all zeros ($\delta > p$). The transitional point is when $\delta = p$.

We now consider the (δ, p) pair such that $\delta = p$ and $\Pr\{Z = 1\} = p * \delta$, denote the value of δ and p that satisfy this by δ^*. For this case it can be shown that the scheme attaining the minimum in (52) assigns probability $1/2$ to the "say what you see" scheme and $1/2$ to the "say all zeros" scheme. In other words it is a mixture of the two optimal schemes. In order for the minimax to be equivalent to the maximin, the optimal minimax denoiser, in the sense of (52), would have to be an optimal denoiser for the worst possible channel, i.e. to swap the minimax for maximin, the channel maximizing the loss would also need to be a BSC with transition probability δ^*. Figure 3 is a plot of δ^* and the δ maximizing the loss with respect to $\Pr\{Z = 1\}$, denoted by δ_w.

Evidently, it is not the case that $\delta^* = \delta_w$. Hence with Figure 3 in mind we can easily verify that for $P_{\mathbf{Z}}$ equal to Bernoulli(0.18) (thus $\delta^* = 0.1$),

$$\min_{f \in M_{opt}} \max_{\{(P_{\mathbf{X}}, \delta): \delta \in \mathcal{U}, P_{\mathbf{X}} * \delta = P_{\mathbf{Z}}\}} E_{[P_{\mathbf{X}}, \delta]}[L_f(X^n, Z^n)] > \max_{\{(P_{\mathbf{X}}, \delta): \delta \in \mathcal{U}, P_{\mathbf{X}} * \delta = P_{\mathbf{Z}}\}} \min_{f \in M_{opt}} E_{[P_{\mathbf{X}}, \delta]}[L_f(X^n, Z^n)] \tag{53}$$

where M_{opt} is a set of optimal denoisers for a given $(P_{\mathbf{Z}}, \delta)$ pair.

This observation leads us to the following theorem

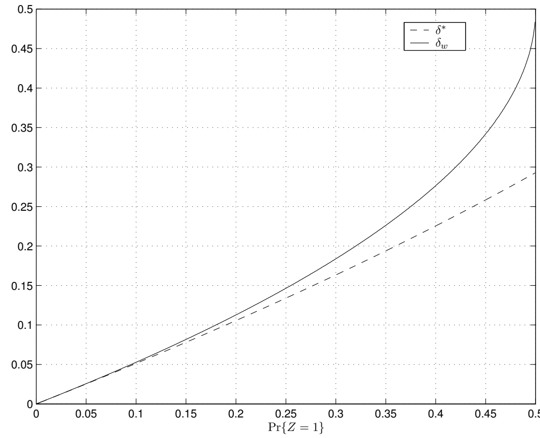

Fig. 3. Plot of δ^* and δ_w with respect to the parameter of the Bernoulli noisy source.

Theorem 1 (Minimax $\neq$ Maximin): There exist stationary ergodic sources $P_{\mathbf{Z}}$ for which

$$\min_{f\in M_k}\max_{\{(P_{\mathbf{X}},\delta):\delta\in\mathcal{U},P_{\mathbf{X}}*\delta=P_{\mathbf{Z}}\}} E_{[P_{\mathbf{X}},\delta]}[L_f(X^n,Z^n)] > \max_{\{(P_{\mathbf{X}},\delta):\delta\in\mathcal{U},P_{\mathbf{X}}*\delta=P_{\mathbf{Z}}\}}\min_{f\in M_k} E_{[P_{\mathbf{X}},\delta]}[L_f(X^n,Z^n)] \tag{54}$$

Proof: Assume, by contradiction, that (54) does not hold when $P_{\mathbf{Z}}$ is the Bernoulli(0.18) source. This implies that the minimizing denoiser is optimal for the worst possible channel. Hence to attain the minimax performace it is enough to minimize only over mixtures of *optimal* denoisers, i.e.,

$$\min_{f\in M_k}\max_{\{(P_{\mathbf{X}},\delta):\delta\in\mathcal{U},P_{\mathbf{X}}*\delta=P_{\mathbf{Z}}\}} E_{[P_{\mathbf{X}},\delta]}[L_f(X^n,Z^n)] = \min_{f\in M_{opt}}\max_{\{(P_{\mathbf{X}},\delta):\delta\in\mathcal{U},P_{\mathbf{X}}*\delta=P_{\mathbf{Z}}\}} E_{[P_{\mathbf{X}},\delta]}[L_f(X^n,Z^n)]. \tag{55}$$

Combining (55) with (53) gives

$$\min_{f\in M_{opt}}\max_{\{(P_{\mathbf{X}},\delta):\delta\in\mathcal{U},P_{\mathbf{X}}*\delta=P_{\mathbf{Z}}\}} E_{[P_{\mathbf{X}},\delta]}[L_f(X^n,Z^n)] > \max_{\{(P_{\mathbf{X}},\delta):\delta\in\mathcal{U},P_{\mathbf{X}}*\delta=P_{\mathbf{Z}}\}}\min_{f\in M_{opt}} E_{[P_{\mathbf{X}},\delta]}[L_f(X^n,Z^n)] \tag{56}$$

On the other hand clearly

$$\max_{\{(P_{\mathbf{X}},\delta):\delta\in\mathcal{U},P_{\mathbf{X}}*\delta=P_{\mathbf{Z}}\}}\min_{f\in M_{opt}} E_{[P_{\mathbf{X}},\delta]}[L_f(X^n,Z^n)] \geq \max_{\{(P_{\mathbf{X}},\delta):\delta\in\mathcal{U},P_{\mathbf{X}}*\delta=P_{\mathbf{Z}}\}}\min_{f\in M_k} E_{[P_{\mathbf{X}},\delta]}[L_f(X^n,Z^n)] \tag{57}$$

which when combined with (56) implies that (54) holds, contradicting our assumption. $\square$

Theorem 1 and Corollary 2 of [4] imply that the MBD is not an optimal denoiser for the worst (source, channel) pair.

VIII. Conclusions

In [4], denoisers that are asymptotically optimal in a worst-case sense are suggested for the setting of an unknown source corrupted by a DMC, under channel uncertainty. This paper was dedicated to the implementation of these denoisers. We have presented efficient algorithms for implementing the denoisers suggested in [4] for the binary alphabet as well as for efficiently estimating the set of feasible channels in the uncertainty set. We also extended these algorithms to a large family of channel in the non-binary case, focusing on the generalized M-array symmetric channel. It was shown that the suggested universally $\mathrm{min\text{-}max}$ denoisers do not correspond to schemes that attain the optimum distribution-dependent performance under the worst case source-channel pair. In general, the $\mathrm{min\text{-}max}$ denoisers are not optimal distribution-dependent schemes for any source-channel pair, implying that use of the DUDE of [9] (with a channel estimate) is suboptimal under the worst-case loss criterion. We have also presented a natural modification to the original DUDE, M-DUDE$(\hat{\Gamma}_l)$, which employs the DUDE using as channel crossover probability the largest one in the estimated feasible set (using our estimator for that set). Simulations shown suggest that this scheme may preform well in denoising images under channel uncertainty.

References

[1] S. Boyd and L. Vandenbeghe. *Convex Optimization,* Cambridge University Press, 2004.

[2] R. Bradley, Basic properties of strong mixing conditions, *E.Eberlein and M.S.Taqqu (eds), Dependence in Probability and Statistics*, 1986. Boston: Birkhuser.

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

[4] G. Gemelos, S. Sigurjónsson and T. Weissman, Universal Discrete Denoising Under Channel Uncertainty, submitted to IEEE Trans. on Information Theory (Preprint available at: http://www.stanford.edu/∼ggemelos/papers/mbdit.ps).

[5] G. Gemelos, S. Sigurjónsson and T. Weissman, Universal Minimax Binary Image Denoising Under Channel Uncertainty, submitted to 2004 International Conference on Image Processing (available at: http://www.stanford.edu/∼ggemelos/papers/icip2004.ps).

[6] R. M. Gray. Rate distortion functions for finite-state finite-alphabet markov sources. *IEEE Trans. Inform. Theory*, IT-17(2):127–134, March 1971.

[7] G. T. Parks, An Intelligent Stochastic Optimization Routine for Nuclear Fuel Cycle Design, *Nuclear Technology* 89, 223-246.

[8] D. Vanderbilt and S. G. Louie, A Monte Carlo Simulated Annealing Approach to Optimization over Continuous Variables, *Journal of Computer Physics*, 45, 885-890

[9] T. Weissman, E. Ordentlich, G. Seroussi, S. Verdu and M Weinberger, Universal Discrete Denoising: Known Channel, to appear in Dec 2004 in IEEE Trans. on Information Theory.