COMMUNICATIONS IN INFORMATION AND SYSTEMS
Vol. 5, No. 2, pp. 257-288, 2005

005

DISCRETE DENOISING FOR CHANNELS WITH MEMORY*

RUI ZHANG† AND TSACHY WEISSMAN†

Abstract. We consider the problem of estimating a discrete signal $X^n = (X_1, \ldots, X_n)$ based on its noise-corrupted observation signal $Z^n = (Z_1, \ldots, Z_n)$. The noise-free, noisy, and reconstruction signals are all assumed to have components taking values in the same finite M-ary alphabet $\{0, \ldots, M-1\}$. For concreteness we focus on the additive noise channel $Z_i = X_i + N_i$, where addition is modulo-M, and $\{N_i\}$ is the noise process. The cumulative loss is measured by a given loss function. The distribution of the noise is assumed known, and may have memory restricted only to stationarity and a mild mixing condition. We develop a sequence of denoisers (indexed by the block length n) which we show to be asymptotically universal in both a semi-stochastic setting (where the noiseless signal is an individual sequence) and in a fully stochastic setting (where the noiseless signal is emitted from a stationary source). It is detailed how the problem formulation, denoising schemes, and performance guarantees carry over to non-additive channels, as well as to higher-dimensional data arrays. The proposed schemes are shown to be computationally implementable. We also discuss a variation on these schemes that is likely to do well on data of moderate size. We conclude with a report of experimental results for the binary burst noise channel, where the noise is a finite-state hidden Markov process (FS-HMP), and a finite-state hidden Markov random field (FS-HMRF), in the respective cases of one- and two-dimensional data. These support the theoretical predictions and show that, in practice, there is much to be gained by taking the channel memory into account.

1. Introduction. The problem of denoising an unknown discrete-time discrete-valued signal corrupted by a known discrete memoryless channel (DMC) was recently studied in [26], which presented a practical denoising algorithm (DUDE), and established its asymptotic universal optimality. Subsequent work considered, among other things, the sequential version of the problem [22], the case of non-discrete noisy signal components [4], the case of channel uncertainty [7, 8], and applications of the DUDE in communications [19]. We refer to these papers, and to the references therein, for the increasing variety of applications where the discrete denoising problem is encountered.

In this work we revisit the setting of [26] for the case where the noise, rather than being memoryless, is a more generally distributed process. For concreteness, we focus on the case of additive noise, though indicate how our findings carry over to the more general case. We first consider a one-dimensional index set in Section 2, where we begin with a concrete description of our setting and assumptions in Subsection 2-A. We then derive a denoiser, arguing intuitively why it should be effective for our setting, in Subsection 2-B. In Subsection 2-C we present a result establishing the asymptotic universal optimality of the scheme suggested in Section 2-B. In Subsec-

*The material in this paper was partially presented at the 42nd Annual Allerton Conference on Communication, Control, and Computing, Monticello, IL, Sept. 29th – Oct. 1st, 2004, [27]. The research was partially supported by NSF Grant CCF-0512140.

†Department of electrical engineering, Stanford University, Stanford, CA 94305, USA. E-mails: {ee.ruizhang, tsachy}@stanford.edu

tion 2-D we point out that our schemes and their performance guarantees extend to accommodate quite general (non-additive) channels. Section 3 presents the extension of the algorithm and its performance guarantees to higher dimensional data arrays. We then discuss computational aspects of the denoiser in Section 4. For simplifying the computation of the denoising rule we establish a result of independent interest regarding the form of the diagonalizing transform of a "lexicographically circulant" matrix. In Section 4 we also present a modified version of the denoiser, which is more efficient than the original one both computationally and, in various scenarios, statistically. In Section 5 we present and discuss experimental results for both one- and two-dimensional data arrays. These experiments show discrete signals and images corrupted by, respectively, a finite-state hidden Markov process (the burst noise channel), and a finite-state hidden Markov random field (FS-HMRF). We conclude in Section 6 with a summary of our findings and some remarks.

2. One-Dimensional Data.

2-A. Problem Setting. We consider the problem of estimating a discrete signal $X^n = (X_1, \ldots, X_n)$ based on its noise-corrupted observation signal $Z^n = (Z_1, \ldots, Z_n)$. For concreteness, we start by assuming that the noise-free, noisy, and reconstruction signals all have components taking values in the same finite M-ary alphabet $\mathcal{A} = \{0, \ldots, M-1\}$, and that the noise is additive

$$Z_i = X_i \oplus N_i, \tag{1}$$

$\oplus$ denoting modulo-M addition and $\{N_i\}$ being the noise process, with $\mathcal{A}$-valued components as well. Our universality setting is w.r.t. the noiseless source, which is entirely unknown. As in [26], we assume knowledge of the channel characteristics (i.e., of the distribution of the noise process).

A n-block denoiser $\hat{X}^n$ is, formally, a mapping taking $\mathcal{A}^n$ into itself. We assume a given single-letter loss function Λ and denote, for $x^n, z^n \in \mathcal{A}^n$

$$L_{\hat{X}^n}(x^n, z^n) = \frac{1}{n}\sum_{i=1}^{n} \Lambda\left(x_i, \hat{X}_i(z^n)\right), \tag{2}$$

with $\hat{X}_i(z^n)$ denoting the i-th component of the n-tuple $\hat{X}^n(z^n)$. In words, $L_{\hat{X}^n}(x^n, z^n)$ is the normalized cumulative loss of the denoiser $\hat{X}^n$ when observing z^n while the underlying signal is x^n.

The α-mixing coefficients of an arbitrary process $\{U_i\}$ with finite-valued components are defined by:

$$\alpha_t^{(U)} = \sup_{\{k\le l\le m\le n:\ m-l\ge t\}} \max_{u_k^l, u_m^n} \left|P(U_k^l = u_k^l, U_m^n = u_m^n) - P(U_k^l = u_k^l)P(U_m^n = u_m^n)\right|, \tag{3}$$

where $U_k^l = (U_k, \ldots, U_l)$, $u_k^l = (u_k, \ldots, u_l)$, etc. We drop the superscript from $\alpha_t^{(U)}$ when the process U is clear from the context. The α-mixing coefficients are a measure

of the effective memory in the process. A process is said to be *α-mixing* if $\alpha_t \longrightarrow 0$ as $t \to \infty$. Of the standard types of mixing (α, β, ϕ, ψ and ρ), α-mixing is the weakest (most benign) requirement in that it is implied by any of the other types of mixing [3].

For every k, let Π^k_{-k} denote the $M^{2k+1} \times M^{2k+1}$ matrix with (x^k_{-k}, z^k_{-k})-th element

$$\Pi^k_{-k}(x^k_{-k}, z^k_{-k}) = P_{N^k_{-k}}(z^k_{-k} \ominus x^k_{-k}), \tag{4}$$

where $\ominus$ in $z^k_{-k} \ominus x^k_{-k}$ denotes componentwise modulo-M subtraction and we assume, for concreteness, lexicographic ordering between the elements of $\mathcal{A}^{2k+1}$. In other words, Π^k_{-k} is the $2k+1$-th order channel matrix whose (x^k_{-k}, z^k_{-k})-th element denotes the probability of z^k_{-k} at the channel output when the underlying noiseless $2k+1$-tuple is x^k_{-k}. Our assumption on the noise distribution is:

ASSUMPTION 1. *$\{N_i\}$ is stationary and α-mixing with $\sum_{t=1}^{\infty} \alpha_t^{(N)} < \infty$, and Π^k_{-k} is non-singular for every k.*

The condition on the summability of the α-mixing coefficients is rather benign, and is satisfied by the noise models arising in practice. In fact, the α-mixing coefficients of a Markov process of any order with no restricted sequences, as well as any hidden Markov process with no restricted state sequences, decay exponentially rapidly [3]. Also, the α-mixing coefficients of finite-length sliding-window functions of i.i.d. variables, clearly satisfy $\alpha_t = 0$ for all $t \geq t_0$ (t_0 depending on the horizon of the sliding-window function). The non-singularity stipulation is also rather benign, holding for the case of memoryless noise whenever Π^0_{-0} (the "single-letter" channel matrix) is invertible [26], as well as for all points in parameter spaces associated with the representations of Markov and hidden Markov processes, with the exception of a negligible subset of the parameter space [6]. To see why this stipulation is needed in our universal denoising context consider the following: For a $2k+1$ tuple X^k_{-k} let $P_{X^k_{-k}}$ denote the M^{2k+1}-dimensional column vector specifying the distribution of X^k_{-k}, i.e., the x^k_{-k}-th component of $P_{X^k_{-k}}$ (according to the lexicographic ordering) is $P(X^k_{-k} = x^k_{-k})$. It is then readily verified that the distribution of the $2k+1$ tuple Z^k_{-k} at the channel output when the input is X^k_{-k} satisfies

$$P^T_{Z^k_{-k}} = P^T_{X^k_{-k}} \cdot \Pi^k_{-k}, \tag{5}$$

implying equivalently, by the non-singularity of Π^k_{-k} stated in Assumption 1, that also

$$P^T_{Z^k_{-k}} \cdot \left(\Pi^k_{-k}\right)^{-1} = P^T_{X^k_{-k}}. \tag{6}$$

Evidently, when Π^k_{-k} is invertible, there is a unique correspondence between the distribution of a noisy $2k+1$-tuple at the channel output and the distribution of the noiseless $2k+1$-tuple at its input. When Π^k_{-k} is not invertible, there may be

a multitude of possible input distributions consistent with a given distribution of a noisy $2k+1$-tuple. Therefore, in this case, the input distribution cannot be inferred even with complete knowledge of the channel output distribution. It should thus be clear that the stipulation on the invertibility of Π^k_{-k}, for each k, in Assumption 1 is necessary in our setting of universality where, at best, one can hope for a good estimate of the said output statistics. Finally, it should be pointed out that processes arising in the modelling of noisy channels typically satisfy this invertibility requirement. One simple example, in the binary setting, is the case of a BSC, where only the case where the crossover probability is $1/2$ does not satisfy this requirement. Additional examples will be given in Subsection 2-C. It can in fact be shown that, when the noise is a hidden Markov process, under benign conditions on the parametrization, all parameter values, except those in a set of zero Lebesgue measure, give rise to a process satisfying Assumption 1.[1]

2-B. Derivation of the Denoiser. For an arbitrarily distributed $2k+1$-tuple X^k_{-k} at the channel input, the following relationship is readily verified to hold using Bayes' rule and (6):

$$\begin{aligned} P(X_0 = a | Z^k_{-k} = z^k_{-k}) &\propto \sum_{x^k_{-k}: x_0 = a} P^T_{X^k_{-k}}(x^k_{-k}) P_{N^k_{-k}}(z^k_{-k} \ominus x^k_{-k}) \\ &= \sum_{x^k_{-k}: x_0 = a} \left[P^T_{Z^k_{-k}} \cdot \left(\Pi^k_{-k}\right)^{-1} \right] (x^k_{-k}) P_{N^k_{-k}}(z^k_{-k} \ominus x^k_{-k}), \end{aligned} \tag{7}$$

where the $\propto$ notation indicates equality up to normalization of the vector whose a-th component is given and $[P^T_{Z^k_{-k}} \cdot \left(\Pi^k_{-k}\right)^{-1}](x^k_{-k})$ denotes the x^k_{-k}-th component of the M^{2k+1}-dimensional (row) vector $P^T_{Z^k_{-k}} \cdot \left(\Pi^k_{-k}\right)^{-1}$. A property of the form on the right side of (7) of key importance in our setting, which is emphasized in the second line of the above display, is that explicitly it only involves the distribution of the noisy $2k+1$-tuple $P_{Z^k_{-k}}$ (and the channel), and not the noiseless source distribution. From (7) it follows that the optimal estimate of X_0 based on Z^k_{-k} under the loss function Λ (in the sense of minimzing expected loss) is given by

$$\hat{X}_0(z^k_{-k}) = \arg\min_{\hat{x}} E\left[\Lambda(X_0, \hat{x}) | Z^k_{-k} = z^k_{-k}\right] \tag{8}$$

$$= \arg\min_{\hat{x}} \sum_a \Lambda(a, \hat{x}) \left[\sum_{x^k_{-k}: x_0 = a} \left[P^T_{Z^k_{-k}} \cdot \left(\Pi^k_{-k}\right)^{-1} \right] (x^k_{-k}) P_{N^k_{-k}}(z^k_{-k} \ominus x^k_{-k}) \right]. \tag{9}$$

[1] E.g., when the HMP is modelled such that the size of the observation space is greater or equal to the size of the state space, then invertibility of the channel matrices required for Assumption 1 results from invertibility of the state-to-observation channel, and invertibility of the underlying state process, separately. The channel associated with almost all (in Lebesgue sense) parameterizations is invertible. The underlying state process can also be shown to be invertible in the required sense for almost all values of the transition probabilities.

This further implies that when X^n is emitted by a stationary source, the best k-th order sliding-window denoiser in the sense of minimizing $E\left[\sum_{i=k+1}^{n-k}\Lambda\left(X_i, f(Z_{i-k}^{i+k})\right)\right]$ is given by $f(z_{-k}^{k}) = \hat{X}_0(z_{-k}^{k})$ (the right side explicitly given in (9)). This denoiser depends on the distribution $P_{Z_{-k}^{k}}$ which is a priori unknown in our universal setting where the noiseless (and hence also noisy) source distribution is assumed unknown. This derivation, however, motivates the following source-distribution-independent n-block denoiser:

$$\hat{X}_i(z^n) = \arg\min_{\hat{x}} \sum_{a} \Lambda(a,\hat{x}) \cdot \left[\sum_{x_{-k}^{k}: x_0 = a} \left[\hat{P}_{Z_{-k}^{k}}(z^n)^T \cdot \left(\Pi_{-k}^{k}\right)^{-1}\right](x_{-k}^{k}) P_{N_{-k}^{k}}(z_{i-k}^{i+k} \ominus x_{-k}^{k})\right], \tag{10}$$

for $k+1 \le i \le n-k$,[2] with $\hat{P}_{Z_{-k}^{k}}(z^n)$ denoting the empirical distribution of a noisy $2k+1$-tuple, i.e.,

$$\hat{P}_{Z_{-k}^{k}}(z^n)[u_{-k}^{k}] = \frac{1}{n-2k}\sum_{i=k+1}^{n-k} \mathbf{1}_{\{z_{i-k}^{i+k} = u_{-k}^{k}\}}. \tag{11}$$

In other words, the unknown distribution of a noisy $2k+1$-tuple is replaced by its empirical estimate (which is based on the observation of the noisy n-tuple). For obvious reasons, we refer to k as the context parameter. A natural implementation of this denoiser will be detailed in Section 4. We mention that the scheme of (10) coincides with the DUDE of [26] when the noise is an i.i.d. process. This can be shown via a computation similar to that in Section 6 of [4], which showed that the scheme of [4] coincides with the DUDE of [26] when the channel input and output alphabets are equal.

2-C. Universal Asymptotic Optimality. To state our main theoretical result we let, as in [26], $D_k(x^n, z^n)$ denote the loss of the best k-th order sliding window denoiser when the clean signal is x^n while the observation is z^n, i.e.,

$$D_k(x^n, z^n) = \min_{f:\mathcal{A}^{2k+1}\to\mathcal{A}} \left[\frac{1}{n-2k}\sum_{i=k+1}^{n-k} \Lambda\left(x_i, f(z_{i-k}^{i+k})\right)\right]. \tag{12}$$

Starting from the statement of the theorem that follows, and on, the "semi-stochastic" setting refers to the case where the noiseless signal is an individual sequence while the "stochastic" setting refers to a probabilistic noiseless signal, independent of the noise process. $\|\cdot\|$, when the argument is either a matrix or a vector, will denote the l_∞ norm, i.e., the maximum of the magnitude of all components.

[2] Similarly as in [26, 4, 7], the reconstruction components located outside the range $k+1 \le i \le n-k$ can be arbitrarily defined and are inconsequential in our analysis which assumes $k \ll n$.

THEOREM 1. *Let Assumption 1 hold and $\hat{X}^{n,k}$ denote the n-block denoiser in (10). Let further $\hat{X}^n_{univ} = \hat{X}^{n,k_n}$ where $\{k_n\}$ satisfies $k_n \to \infty$ and*

$$\frac{1}{n} k_n M^{12k_n} \left(\left\| \left(\Pi^{k_n}_{-k_n} \right)^{-1} \right\| + 1 \right)^2 \longrightarrow 0 \quad \text{as} \quad n \to \infty. \tag{13}$$

1. *Semi-Stochastic Setting: For any sequence $\{x_n\}_{n\geq 1}$, $x^n \in \mathcal{A}^n$,*

$$L_{\hat{X}^n_{univ}}(x^n, Z^n) - D_{k_n}(x^n, Z^n) \longrightarrow 0 \quad \text{in probability.} \tag{14}$$

2. *Stochastic Setting: For any stationary process $\mathbf{X} = (X_1, X_2, \ldots)$,*

$$\lim_{n\to\infty} E L_{\hat{X}^n_{univ}}(X^n, Z^n) = \inf_{n\geq 1} \min_{\hat{X}^n} E L_{\hat{X}^n}(X^n, Z^n), \tag{15}$$

where the minimization on the right side is over all n-block denoisers.

The proof in fact shows that the convergence in (14) is uniform in the sense that, for every $\varepsilon > 0$,

$$\max_{x^n \in \mathcal{A}^n} P\left(\left| L_{\hat{X}^n_{univ}}(x^n, Z^n) - D_{k_n}(x^n, Z^n) \right| > \varepsilon \right) \longrightarrow 0. \tag{16}$$

Before turning to the proof of Theorem 1, let us consider a few examples to put the requirement (13) in perspective:

EXAMPLE 1 (Memoryless noise). *For a memoryless channel, i.e. when $\{N_i\}$ is an i.i.d. process, $\Pi^k_{-k} = (\Pi^0_{-0})^{\otimes(2k+1)}$ (where $\otimes(2k+1)$ denotes the $2k+1$-th order tensor power and Π^0_{-0} is the matrix associated with one input-output pair). Thus in this case $\left\| \left(\Pi^k_{-k}\right)^{-1} \right\| = \left\| \left(\Pi^0_{-0}\right)^{-1} \right\|^{2k+1}$ and it is readily verified that (13) is satisfied by $k_n = c \log n$ for a sufficiently small c (dependent on Π^0_{-0}). For this case, [26, Theorem 1] indeed shows that for $k_n = C \log n$, for C larger than the possible c implied by Theorem 1, (14) still holds (and, in fact, the convergence holds with probability one).*

EXAMPLE 2 (Binary noise modulated by an arbitrarily distributed state process). *Let $\{S_i\}$ be an arbitrarily distributed state process with $\mathcal{S}$-valued components and $\{N_i\}$ be a binary process whose components are independent when conditioned on $\{S_i\}$, where $N_i|S_i = s \sim$ Bernoulli(δ_s) for every $s \in \mathcal{S}$ (the processes of Section 5 are of this form, with $\{S_i\}$ being a binary Markov process). Let $\delta = \sup_{s\in\mathcal{S}} \delta_s$ and assume $\delta < 1/2$. For every $s^k_{-k} \in \mathcal{S}^{2k+1}$ let $\Pi_{s^k_{-k}}$ denote the matrix given by $\Pi_{s^k_{-k}}(x^k_{-k}, z^k_{-k}) = P\left(Z^k_{-k} = z^k_{-k} | x^k_{-k}, S^k_{-k} = s^k_{-k}\right)$. It is then readily checked that $\Pi_{s^k_{-k}} = \bigotimes_{i=-k}^{k} \Pi_{s_i}$, where Π_s is the "single-letter" channel matrix associated with the state s and $\bigotimes$ denotes the tensor product. Since the eigenvalues of Π_s are 1 and $1-2\delta_s$ it follows that the minimum eigenvalue of $\Pi_{s^k_{-k}}$ is lower bounded by $(1-2\delta)^{2k+1}$, for all s^k_{-k}. Furthermore, [16, Theorem 1] implies that $\Pi_{s^k_{-k}}$ has the same diagonalizing transform (i.e., same set of eigenvectors, namely the columns of the $2k+1$-th order Hadamard matrix) for all s^k_{-k}. Since $\Pi^k_{-k} = \int \Pi_{s^k_{-k}} dP(s^k_{-k})$ this implies that, for every $1 \leq$*

$i \le 2^{2k+1}$, $\lambda^{(i)} = \int \lambda^{(i)}_{s^k_{-k}} dP(s^k_{-k})$, where $\lambda^{(i)}$ and $\lambda^{(i)}_{s^k_{-k}}$ denote, respectively, the i-th eigenvalue of Π^k_{-k} and $\Pi_{s^k_{-k}}$. This finally implies that the minimum eigenvalue of Π^k_{-k} is lower bounded by $(1-2\delta)^{2k+1}$ and, hence, that $\left\|\left(\Pi^k_{-k}\right)^{-1}\right\| \le 1/(1-2\delta)^{2k+1}$. Thus, for this case too, (13) is satisfied by $k_n = c\log n$ for appropriate c (dependent on δ).

EXAMPLE 3 (Contagion channels [1]). *Contagion channels are binary additive noise channels often arising in communications, where the noise process is an M-th order Markov process with transition probabilities characterized by*

$$P(N_t = 1 | N_{t-M}^{t-1} = n_{t-M}^{t-1}) = \frac{\varepsilon + w(n_{t-M}^{t-1})\delta}{1+M\delta}, \tag{17}$$

where w denotes Hamming weight, $\varepsilon = P(N_t = 1)$. The distribution of this process is completely characterized by the triplet (M, ε, δ). Note that this family includes all first-order binary Markov processes. Theorem 3 of [16] implies that for this noise process, assuming $\varepsilon < 1/2$, the minimum eigenvalue of Π^k_{-k} is positive and lower bounded by $\left(\frac{1-2\varepsilon}{1+M\delta}\right)^{2k+1}$. Hence $\left\|\left(\Pi^k_{-k}\right)^{-1}\right\| \le \left(\frac{1-2\varepsilon}{1+M\delta}\right)^{-(2k+1)}$ and (13) is satisfied, for appropriate $c = c(M,\varepsilon,\delta) > 0$, by $k_n = c\log n$.

Note that in the above examples it was seen that $k_n = c\log n$, for appropriate c, satisfies (13). This is the largest growth rate allowable if (13) is to be achieved, since clearly a necessary condition for (13) to hold is that $\frac{1}{n}k_n M^{12k_n} \longrightarrow 0$, which already requires that k_n grow not faster than $c\log n$.

Defining now

$$q_k(z^n, x^n)[a, u^k_{-k}] = \frac{1}{n-2k}|\{k+1 \le i \le n-k : x_i = a, z^{i+k}_{i-k} = u^k_{-k}\}|, \tag{18}$$

where $|\cdot|$ here denotes cardinality, and

$$\hat{q}_k(z^n)[a, u^k_{-k}] = \sum_{x^k_{-k}: x_0 = a} [\hat{P}_{Z^k_{-k}}[z^n]^T \cdot \left(\Pi^k_{-k}\right)^{-1}](x^k_{-k}) P_{N^k_{-k}}(u^k_{-k} \ominus x^k_{-k}), \tag{19}$$

it follows from the definition of $\hat{X}^{n,k}$ (recall (10)) and a direct application of [26, Lemma 1] that for all $x^n, z^n \in \mathcal{A}^n$

$$|L_{\hat{X}^{n,k}}(x^n, z^n) - D_k(x^n, z^n)| \le \Lambda_{max} M^{2k+2} \|q_k(z^n, x^n) - \hat{q}_k(z^n)\|, \tag{20}$$

where $\Lambda_{max} = \max_{x,\hat{x}} \Lambda(x,\hat{x})$. It is hence clear that for proving the first item of Theorem 1 it suffices to establish the smallness of $\|q_k(Z^n, x^n) - \hat{q}_k(Z^n)\|$, with high probability. This is done in the following theorem, whose proof is given in Appendix A.

THEOREM 2. *For all $n, k, x^n \in \mathcal{A}^n$ and $\varepsilon > 0$*

(21)

$$P\left(\|\hat{q}_k(Z^n) - q_k(Z^n, x^n)\| \ge \varepsilon\right) \le M^{8k+2} \frac{(4k+1+2\sum_{t=1}^{\infty}\alpha_t^{(N)})\left(\left\|\left(\Pi^k_{-k}\right)^{-1}\right\| + 1\right)^2}{\varepsilon^2(n-2k)}.$$

Proof of Theorem 1 assuming Theorem 2. The combination of (20) and Theorem 2 implies

$$\begin{aligned} & P\left(\left|L_{\hat{X}^{n,k}}(x^n,Z^n)-D_k(x^n,Z^n)\right|>\varepsilon\right) \\ & \leq M^{8k+2}\frac{(4k+1+2\sum_{t=1}^{\infty}\alpha_t^{(N)})\left(\|\left(\Pi_{-k}^{k}\right)^{-1}\|+1\right)^2}{\left(\frac{\varepsilon}{\Lambda_{max}M^{2k+2}}\right)^2(n-2k)} \qquad (22) \\ & = \Lambda_{max}M^{12k+6}\frac{(4k+1+2\sum_{t=1}^{\infty}\alpha_t^{(N)})\left(\|\left(\Pi_{-k}^{k}\right)^{-1}\|+1\right)^2}{\varepsilon^2(n-2k)}. \qquad (23) \end{aligned}$$

Condition (13) guarantees that taking $k=k_n$ on the right side of (23) gives an expression converging to 0, implying (16) (since $\hat{X}^n_{univ}=\hat{X}^{n,k}$). This proves the first item. Proof of the second item is similar, given the first item, to that of [26, Theorem 3]. Specifically, note that for every fixed k and all sufficiently large n

$$D_k(x^n,z^n)\geq D_{k_n}(x^n,z^n) \quad \forall x^n,z^n\in\mathcal{A}^n. \qquad (24)$$

It thus follows from the first item that for any $\varepsilon>0$ and all sufficiently large n

$$P\left(L_{\hat{X}^n_{univ}}(X^n,Z^n)\geq D_k(X^n,Z^n)+\varepsilon\right)\leq\varepsilon \qquad (25)$$

and therefore

$$EL_{\hat{X}^n_{univ}}(X^n,Z^n)\leq ED_k(X^n,Z^n)+\varepsilon+\varepsilon\Lambda_{max}=ED_k(X^n,Z^n)+\varepsilon(1+\Lambda_{max}). \qquad (26)$$

Now, it follows from the joint stationarity of $(\mathbf{X},\mathbf{Z})$, exactly as in the proof of [26, Theorem 3] (cf. display (72) therein), that

$$ED_k(X^n,Z^n)\leq E\left[\min_{\hat{x}\in\mathcal{A}}E\left[\Lambda(X_0,\hat{x})|Z_{-k}^{k}\right]\right]. \qquad (27)$$

Furthermore, Claim 2 and Lemma 4 of [26], along with their proofs, hold *verbatim* for our present setting, implying in particular

$$\lim_{k\to\infty}E\left[\min_{\hat{x}\in\mathcal{A}}E\left[\Lambda(X_0,\hat{x})|Z_{-k}^{k}\right]\right]=\inf_{n\geq1}\min_{\hat{X}^n}EL_{\hat{X}^n}(X^n,Z^n). \qquad (28)$$

Displays (26), (27) and (28), combined with the arbitrariness of ε, imply

$$\limsup_{n\to\infty}EL_{\hat{X}^n_{univ}}(X^n,Z^n)\leq\inf_{n\geq1}\min_{\hat{X}^n}EL_{\hat{X}^n}(X^n,Z^n), \qquad (29)$$

which completes the proof since, trivially,

$$\liminf_{n\to\infty}EL_{\hat{X}^n_{univ}}(X^n,Z^n)\geq\inf_{n\geq1}\min_{\hat{X}^n}EL_{\hat{X}^n}(X^n,Z^n). \qquad (30)$$

□

2-D. General Stationary Channels. Suppose now that, instead of the additive noise channel, we have a general channel characterized by $\{P(\cdot|x_{-\infty}^{\infty})\}_{x_{-\infty}^{\infty}}$ where, for every $x_{-\infty}^{\infty}$, $P(\cdot|x_{-\infty}^{\infty})$ stands for the law of the channel output process when the input is $x_{-\infty}^{\infty}$. In this case, our assumptions on the channel, which are analogous to those of Assumption 1, are:

1. Stationarity: If $Z_{-\infty}^{\infty} \sim P(\cdot|x_{-\infty}^{\infty})$ and $U_{-\infty}^{\infty} \sim P(\cdot|T\left(x_{-\infty}^{\infty}\right))$ then $U_{-\infty}^{\infty} \stackrel{d}{=} T\left(Z_{-\infty}^{\infty}\right)$ (where T denotes the shift transformation).
2. For all m, n, $P(z_{-m}^{n}|x_{-\infty}^{\infty}) = P(z_{-m}^{n}|\tilde{x}_{-\infty}^{\infty})$ whenever $x_{-m}^{n} = \tilde{x}_{-m}^{n}$. We thus write $P(z_{-m}^{n}|x_{-m}^{n})$ instead of $P(z_{-m}^{n}|x_{-\infty}^{\infty})$, as the latter depends on $x_{-\infty}^{\infty}$ only through x_{-m}^{n}.
3. Given the previous assumption, we can define
$$\Pi_{-k}^{k}(x_{-k}^{k}, z_{-k}^{k}) = P(z_{-k}^{k}|x_{-k}^{k}), \tag{31}$$
and we assume that Π_{-k}^{k} is non-singular for every k.
4. $\sum_{t=1}^{\infty} \alpha_t < \infty$, where the α-mixing coefficients are now defined as:
(32)
$$\alpha_t = \sup_{\{k \le l \le m \le n:\ m-l \ge t\}} \max_{x_{-\infty}^{\infty}, z_k^l, z_m^n} \left| P(z_k^l, z_m^n|x_{-\infty}^{\infty}) - P(z_k^l|x_{-\infty}^{\infty}) P(z_m^n|x_{-\infty}^{\infty}) \right|.$$

As an example for a rich family of channels satisfying the above assumptions, consider the case where the channel input-output relationship can be expressed as

$$Z_i = f(x_i, N_{i-l}^{i+l}), \tag{33}$$

for a finite l, and an f whose range is $\mathcal{A}$. The channel matrix would now be defined by

$$\Pi_{-k}^{k}(x_{-k}^{k}, z_{-k}^{k}) = P\left(Z_{-k}^{k} = f(x_{-k}^{k}, N_{-k-l}^{k+l})\right), \tag{34}$$

where with slight abuse of notation $f(x_{-k}^{k}, N_{-k-l}^{k+l})$ denotes the $2k+1$-tuple whose i-th component is $f(x_i, N_{i-l}^{i+l})$, and the probability in the right side of (34) assumes the semi-stochastic setting where x_{-k}^{k} is an individual sequence. Note that (34) reduces to (4) in the additive case. It is readily verified that the channel in (33) satisfies the above four assumptions whenever the noise process $\{N_i\}$ satisfies Assumption 1 (with Π_{-k}^{k} defined via (34) instead of via (4)).

The denoising rule for this more general channel assumes the form

$$\hat{X}_i(z^n) = \arg\min_{\hat{x}} \sum_{a} \Lambda(a, \hat{x}) \cdot \left[\sum_{x_{-k}^{k}: x_0 = a} \left[\hat{P}_{Z_{-k}^{k}}(z^n)^T \cdot \left(\Pi_{-k}^{k}\right)^{-1} \right] (x_{-k}^{k}) \Pi_{-k}^{k}(x_{-k}^{k}, z_{i-k}^{i+k}) \right], \tag{35}$$

which is similar to that in (10), with $P_{N_{-k}^{k}}(z_{i-k}^{i+k} \ominus x_{-k}^{k})$ replaced by the more general $\Pi_{-k}^{k}(x_{-k}^{k}, z_{i-k}^{i+k})$.

Theorem 1 holds *verbatim* in this more general setting (replacing Assumption 1 by the above four assumptions and with $\hat{X}^{n,k}$ denoting the denoiser in (35)). The proof is also readily verified to carry over essentially verbatim, replacing throughout $P_{N^k_{-k}}(u^k_{-k} \ominus y^k_{-k})$ by $\Pi^k_{-k}(y^k_{-k}, u^k_{-k})$.

3. Multi-Dimensional Data Arrays. We now detail how the problem formulation, schemes, and results of the previous section extend to data arranged in a multi-dimensional array. To avoid cumbersome notation we assume the data set is of dimension $d = 2$, i.e., *an image*, with the implication that the extension to any higher number of dimensions is straightforward.

3-A. Problem Setting and Notation. $\mathcal{A} = \{0, \ldots, M-1\}$ will continue to denote the finite alphabet where the components of the clean, the noise-corrupted, and the reconstructed image take their values. Following the notation of [20], for any $\mathcal{S} \subseteq \mathbb{Z}^2$ we denote $x(\mathcal{S}) = \{x_i\}_{i\in\mathcal{S}}$, $z(\mathcal{S}) = \{z_i\}_{i\in\mathcal{S}}$, etc. Thus, $x(\mathcal{S})$ is a $|\mathcal{S}|$-dimensional vector with $\mathcal{A}$-valued components indexed by the elements of $\mathcal{S}$, and we denote by $\mathcal{A}^{\mathcal{S}}$ the set of all such vectors. For $m, n \in \mathbb{N}$ let $V_{m\times n}$ denote the $m \times n$ rectangle $\{(i_x, i_y) \in \mathbb{N}^2 : i_x \le m, i_y \le n\}$. To simplify notation, we shall write $x_{m\times n}$ for $x(V_{m\times n})$, $z_{m\times n}$ for $z(V_{m\times n})$, and $\mathcal{A}^{m\times n}$ for $\mathcal{A}^{V_{m\times n}}$. Also, for $\mathcal{S} \subseteq \mathbb{Z}^2$ and $i \in \mathbb{Z}^2$ we let $\mathcal{S} + i = \{j + i : j \in \mathcal{S}\}$.

A *neighborhood* is a finite subset of $\mathbb{Z}^2$ containing the origin $(0, 0)$ (the *center* of the neighborhood). Analogously as in previous sections, $P_{X(\mathcal{S})}$ will denote the $M^{|\mathcal{S}|}$-dimensional column vector specifying the distribution of $X(\mathcal{S})$, i.e., the $x(\mathcal{S})$-th component of $P_{X(\mathcal{S})}$ according to, say, the lexicographic order[3], is $P(X(\mathcal{S}) = x(\mathcal{S}))$. Analogously as in (11), we let $\hat{P}_{Z(\mathcal{S})}(z_{m\times n})$ denote the empirical distribution of a noisy $\mathcal{S}$-configuration induced by $z_{m\times n}$, i.e.,

$$\hat{P}_{Z(\mathcal{S})}(z_{m\times n})[u(\mathcal{S})] = \frac{|\{i \in V_{m\times n} : \mathcal{S} + i \subseteq V_{m\times n}, z(\mathcal{S} + i) = u(\mathcal{S})\}|}{|\{i \in V_{m\times n} : \mathcal{S} + i \subseteq V_{m\times n}\}|}. \tag{36}$$

We assume that the noiseless image (for which no statistical model is available) is corrupted by additive noise. In other words, the channel model (1) remains intact, this time i being a two-dimensional index and $\{N_i\}$ being the random noise field. With any finite $\mathcal{S} \subseteq \mathbb{Z}^2$ we associate a channel matrix $\Pi_{\mathcal{S}}$, which is a $M^{|\mathcal{S}|} \times M^{|\mathcal{S}|}$ matrix such that the entry indexed by the pair $(x(\mathcal{S}), z(\mathcal{S}))$ is

$$\Pi_{\mathcal{S}}[x(\mathcal{S}), z(\mathcal{S})] = P(N(\mathcal{S}) = z(\mathcal{S}) \ominus x(\mathcal{S})). \tag{37}$$

The definition of the α-mixing coefficients is extended to our current multi-

[3]The lexicographic order on the elements of $\mathbb{Z}^2$ induces a natural order on the elements of any $\mathcal{S} \subseteq \mathbb{Z}^2$ which, in turn, induces a natural order on $\mathcal{A}^{\mathcal{S}}$. The latter is what we refer to as the 'lexicographic order' in this context.

dimensional setting as

$$\alpha_t^{(N)} = \sup_{\{\mathcal{S},\mathcal{S}':\ d(\mathcal{S},\mathcal{S}')\geq t\}} \max_{u(\mathcal{S}),u(\mathcal{S}')} \big|P(N(\mathcal{S})=u(\mathcal{S}),N(\mathcal{S}')=u(\mathcal{S}')) - P(N(\mathcal{S})=u(\mathcal{S}))\cdot P(N(\mathcal{S}')=u(\mathcal{S}'))\big|\,, \tag{38}$$

with $\mathcal{S},\mathcal{S}'$ in the supremum being subsets of $\mathbb{Z}^2$, and d denoting the distance between $\mathcal{S}$ and $\mathcal{S}'$ defined by $d(\mathcal{S},\mathcal{S}') = \min_{i\in\mathcal{S},i'\in\mathcal{S}'}\|i-i'\|$. Assumption 1, when extended to the present two-dimensional setting, assumes the following form

ASSUMPTION 2. *$\{N_i\}$ is spatially stationary and α-mixing with $\sum_{t=1}^{\infty}\alpha_t^{(N)}<\infty$, and $\Pi_{\mathcal{S}}$ is non-singular for every neighborhood $\mathcal{S}$.*

Similarly as was argued for the one-dimensional case, the summability condition is rather benign. In fact, most finite-alphabet noise field models arising in practice have exponentially decaying α-mixing coefficients, including Markov Random Fields (MRFs) with a finite neighborhood structure and positive transition probabilities and Gibbs fields with a summable potential [9, 15].

A $m\times n$ *image denoiser* is a mapping $\hat{X}^{m\times n}:\mathcal{A}^{m\times n}\to\mathcal{A}^{m\times n}$. For $x_{m\times n},z_{m\times n}\in\mathcal{A}^{m\times n}$ we let $L_{\hat{X}^{m\times n}}(x_{m\times n},z_{m\times n})$ denote the normalized denoising loss, as measured by the single-letter loss function Λ, of the image denoiser $\hat{X}^{m\times n}$ when the observed noisy image is $z_{m\times n}$ and the underlying one is $x_{m\times n}$, i.e.,

$$L_{\hat{X}^{m\times n}}(x_{m\times n},z_{m\times n}) = \frac{1}{mn}\sum_{i\in V_{m\times n}}\Lambda(x_i,\hat{X}^{m\times n}(z_{m\times n})[i]), \tag{39}$$

with $\hat{X}^{m\times n}(z_{m\times n})[i]$ denoting the component of $\hat{X}^{m\times n}(z_{m\times n})$ at the i-th location.

3-B. Description of the Denoiser. For a neighborhood $\mathcal{S}\subseteq\mathbb{Z}^2$ define the $m\times n$ image denoiser $\hat{X}_{\mathcal{S}}^{m\times n}$, for locations i such that $\mathcal{S}+i\subseteq V_{m\times n}$, by

$$\hat{X}_{\mathcal{S}}^{m\times n}(z_{m\times n})[i] = \arg\min_{\hat{x}}\sum_a \Lambda(a,\hat{x}) \cdot\left[\sum_{x(\mathcal{S}):x_0=a}\left[\hat{P}_{Z(\mathcal{S})}(z_{m\times n})^T\cdot\Pi_{\mathcal{S}}^{-1}\right]_{x(\mathcal{S})}\cdot P_{N(\mathcal{S})}[z(\mathcal{S})\ominus x(\mathcal{S})]\right], \tag{40}$$

where $[\cdot]_{x(\mathcal{S})}$ denotes the component of the $\mathcal{A}^{\mathcal{S}}$-dimensional argument indexed by $x(\mathcal{S})$. The denoiser output for locations i where $\mathcal{S}+i\not\subseteq V_{m\times n}$ does not affect the validity of the theoretical results below and, for concreteness, can be assumed set to some arbitrary symbol in $\mathcal{A}$.

To define our denoiser let $\mathcal{B}_r$ denote the l_∞ ball[4] of radius r in $\mathbb{Z}^2$ centered at $(0,0)$, i.e., $\mathcal{B}_r=\{i\in\mathbb{Z}^2:\|i\|_1\leq r\}$. Our $m\times n$ image denoiser can now be defined as

$$\hat{X}_{\mathsf{univ}}^{m\times n} = \hat{X}_{\mathcal{B}_{r(m,n)}}^{m\times n}, \tag{41}$$

[4]The particular choice of the l_∞ norm in this context is not crucial. It corresponds to taking square contexts, whereas, e.g., l_1 would have corresponded to diamond-shaped contexts, l_2 to ball-shaped contexts, etc.

i.e., the denoiser defined in (40) when taking for the neighborhood $\mathcal{S} = \mathcal{B}_r$, where the radius of the ball $r = r(m,n)$ depends on the image dimensions in a way to be specified below. The radius $r(m,n)$ can be thought of as the two-dimensional analog of the context length k_n.

3-C. Asymptotic Optimality. A *sliding window denoiser* of radius r is one that determines the denoised value at a location i as a function of $z(\mathcal{B}_r + i)$. Let $D_r(x_{m\times n}, z_{m\times n})$ denote the *r-th order denoisability* of $(x_{m\times n}, z_{m\times n})$, defined by

$$D_r(x_{m\times n}, z_{m\times n}) = \min_{f:\mathcal{A}^{\mathcal{B}_r}\to\mathcal{A}} \left[\frac{1}{mn}\sum_{i:\mathcal{B}_r+i\in V_{m\times n}} \Lambda\left(x_i, f\left(z(\mathcal{B}_r+i)\right)\right)\right]. \tag{42}$$

This can be interpreted as the denoising performance of a "genie-aided" scheme, allowed to select the best sliding-window denoiser of radius $\le r$, based on knowledge of both the noisy *and the underlying noiseless image*. Note that most image denoisers applied in practice, such as median filters, morphological operators , and context-dependent spatial operators (cf., e.g., [11, 25]) are sliding-window denoisers, so the r-th order denoisability is a lower bound on the performance of all such schemes (for r large enough). Theorem 3 below is the two-dimensional version of Theorem 1: its first part guarantees that the image denoiser $\hat{X}^{m\times n}_{\mathsf{univ}}$ does essentially as well as this genie-aided scheme, regardless of the underlying noiseless image. Its second part guarantees that optimum performance is universally achieved also in the fully stochastic setting where the underlying image is a realization of a spatially stationary random field.

This result can also be viewed as the extension of those in [20, Section 4] to the case of channels with memory. Its proof, which is based on straightforward extensions to the multi-dimensional case of the ideas in the proof of Theorem 1, is omitted.

THEOREM 3. *Let Assumption 2 hold and $g:\mathbb{N}\to\mathbb{N}$ be any function satisfying $\lim_{l\to\infty} g(l) = \infty$ yet slowly enough so that*

$$\frac{1}{l^2} g(l)^2 M^{12g(l)^2} \left\|\Pi_{\mathcal{B}_{g(l)}}\right\|^2 \longrightarrow 0 \quad \text{as } \ l\to\infty. \tag{43}$$

Let $\hat{X}^{m\times n}_{\mathsf{univ}}$ be the denoiser defined in (41) taking $r(m,n) = g(\min\{m,n\})$.

1. *Semi-Stochastic Setting: For any collection of images $\{x_{m\times n}\}_{m,n}$, $x_{m\times n}\in\mathcal{A}^{m\times n}$,*

$$L_{\hat{X}^{m\times n}_{\mathsf{univ}}}(x_{m\times n}, Z_{m\times n}) - D_{r(m,n)}(x_{m\times n}, Z_{m\times n}) \longrightarrow 0 \quad \text{in probability} \tag{44}$$

   as $m,n\to\infty$.

2. *Stochastic Setting: For any spatially stationary process $\mathbf{X} = \{X_i\}$,*

$$\lim_{m,n\to\infty} E\left[L_{\hat{X}^{m\times n}_{\mathsf{univ}}}(X_{m\times n}, Z_{m\times n})\right] = \inf_{m,n\ge 1}\min_{\hat{X}^{m\times n}} E\left[L_{\hat{X}^{m\times n}}(X_{m\times n}, Z_{m\times n})\right], \tag{45}$$

   where the minimization on the right side is over all $m\times n$-image denoisers.

4. Complexity and Implementation. For concreteness below, our discussion refers to the scheme $\hat{X}^{n,k}$ of Section 2. It applies also to the scheme $\hat{X}^{m\times n}_{\mathcal{B}_r}$ of Section 3 under the association $n \to mn$ and $k \to r^2$.

4-A. Algorithm Description. The proposed denoising scheme is described in the following steps. A rough count of the computation time-complexity in each step is given in terms of the number of arithmetic operations required.

- **Pre-processing.** Before the data is read, the inverse of the channel transition matrix, $(\Pi^k_{-k})^{-1}$, is computed for being used in Computation of Decoding Rule (to follow). The matrix dimension of Π^k_{-k} is $M^{2k+1} \times M^{2k+1}$, so a standard computation of its inverse requires $O(M^{6k})$ operations (cf., e.g., [13]). For additive noise channels, we show in the next subsection that this complexity can be significantly reduced to $O(kM^{2k})$ operations.
- **Computation of Counts.** The noise-corrupted data is scanned and the $2k+1$-tuple empirical distribution of the noisy data, $\hat{P}_{Z^k_{-k}}(z^n)$, is computed through counting the number of appearances of the different $2k+1$-tuples in one pass, as they appear in the noisy data. This requires $O(kn)$ operations.
- **Computation of Decoding Rule.** The decoding rule of the denoiser in (10) (or more generally in (35)) is determined. With $\hat{P}_{Z^k_{-k}}(z^n)$, the $2k+1$-tuple distribution of the noise-free source, $P^T_{X^k_{-k}}$, is estimated as $\hat{P}_{Z^k_{-k}}[z^n]^T \cdot (\Pi^k_{-k})^{-1}$, which requires $2M^{4k+2}$ number of operations (for the special case of additive noise, this complexity can be further reduced by the efficient algorithm presented in the next subsection). Next, the decoding rule for estimating the source symbol given its associated $2k+1$-tuple is determined. Because each $2k+1$-tuple requires at most $2M^{2k+1}$ operations for computing its associated decoding rule in (10), the total number of operations required for all possible $2k+1$-tuples observed is at most $2M^{4k+2}$. Adding $2M^{4k+2}$ operations in the estimation of $P_{X^k_{-k}}$, the total number of operations required in this step is $4M^{4k+2}$, i.e., $O(M^{4k})$.
- **Denoising.** The noise-corrupted data is scanned in a second time. At each location, the source symbol is decoded according to its associated $2k+1$-tuple in the observed data and the decoding rule developed. This requires a number of operations, similarly as in the Computation of Counts stage, $O(kn)$.

To sum up, the total computational time-complexity is $O(kn + M^{4k})$, excluding the computation in the Pre-processing stage (which need not be done in "real-time", and is performed once, after which time the same algorithm can be reapplied on different data sets). By taking k_n to be say[5] $\leq \frac{1}{4}\log n$, we get total time-complexity

[5]Note that for the theoretical performance guarantees only an upper bound on the growth rate of k_n is required, and any lesser growth rate will do, so long as $k_n \to \infty$. Thus, even when $k_n = C\log n$ is allowed for $C > 1/4$ from the viewpoint of Theorem 1, taking $k_n = \frac{1}{4}\log n$ will still comply with

$O(n \log n)$.

4-B. Efficient Computation of $\hat{P}_{X_{-k}^k}$ for Additive Noise. In this subsection, we present an efficient algorithm for computing the estimated empirical distribution of the $2k+1$-tuple noiseless source signal, $\hat{P}^T_{X_{-k}^k} = \hat{P}^T_{Z_{-k}^k} \cdot (\Pi_{-k}^k)^{-1}$, for the case of additive noise. As already noted, when the noise is additive, the $2k+1$-tuple channel transition matrix, Π_{-k}^k, satisfies

$$\Pi_{-k}^k\left(x_{-k}^k, z_{-k}^k\right) = \Pi_{-k}^k\left(\tilde{x}_{-k}^k, \tilde{z}_{-k}^k\right) \quad \text{whenever} \quad z_{-k}^k \ominus x_{-k}^k = \tilde{z}_{-k}^k \ominus \tilde{x}_{-k}^k,$$

a property we shall refer to as *lexicographically circulant*. For matrices with this property we have the following result, whose proof is given in Appendix B.

THEOREM 4. *Let $\mathcal{F}_M$ denote the $M \times M$ Fourier matrix*

$$\mathcal{F}_M(l,m) = \frac{1}{\sqrt{M}} \exp\left\{-j\frac{2\pi}{M} lm\right\} \qquad 0 \le l \le M-1,\ 0 \le m \le M-1, \tag{46}$$

and

$$\mathcal{H}_n = \mathcal{F}_M^{\otimes n}. \tag{47}$$

Then:

1. *$\mathcal{H}_{2k+1}$ diagonalizes Π_{-k}^k, i.e., $\Pi_{-k}^k = \mathcal{H}_{2k+1}^* \Gamma \mathcal{H}_{2k+1}$, where Γ is diagonal and $*$ denotes conjugate transpose.*
2. *$diag(\Gamma) = \mathcal{H}_{2k+1} \cdot P_{N_{-k}^k}$, where $diag(\mathcal{X})$ denotes a column vector consisting of the diagonal elements of a square matrix $\mathcal{X}$.*

For the *binary* additive noise case, the diagonalizing matrix, $\mathcal{H}_{2k+1}$, becomes the well-known Walsh-Hadamard matrix [16], and this leads to an efficient algorithm for computing $(\Pi_{-k}^k)^{-1}$ that requires $O(k2^{2k})$ number of operations [12], as compared to the $O(2^{6k})$ which would be required by direct computation. Here we generalize the results of [16, 12] to additive noise over a general finite alphabet. $\hat{P}_{X_{-k}^k}$ can be now computed as

$$\hat{P}_{X_{-k}^k} = \left(\Pi_{-k}^k\right)^{-T} \cdot \hat{P}_{Z_{-k}^k} = \mathcal{H}_{2k+1} \cdot \left[\left(\mathcal{H}_{2k+1}^* \cdot \hat{P}_{Z_{-k}^k}\right) \oslash \left(\mathcal{H}_{2k+1} \cdot P_{N_{-k}^k}\right)\right], \tag{48}$$

where $\oslash$ denotes component-wise division, i.e., $(X \oslash Y)[i] = X_i/Y_i$, and the rightmost equality in (48) follows from Theorem 4. Let $\mathbf{H}_{2k+1}(X)$ and $\mathbf{H}^{\mathtt{inv}}_{2k+1}(X)$ denote, respectively, the "generalized" Fourier transform and inverse Fourier transform of a vector X, i.e., $\mathbf{H}_{2k+1}(X) = \mathcal{H}_{2k+1} \cdot X$, and $\mathbf{H}^{\mathtt{inv}}_{2k+1}(X) = \mathcal{H}^*_{2k+1} \cdot X$. It is noted that when $k = 0$, these two transforms become standard Fourier transform and inverse Fourier transform, respectively. Now, $\hat{P}_{X_{-k}^k}$ in (48) can be obtained through computing two generalized Fourier transforms and one generalized inverse Fourier transform. It is shown in Lemma 4 of Appendix B that both $\mathbf{H}_{2k+1}(X)$ and $\mathbf{H}^{\mathtt{inv}}_{2k+1}(X)$ can be

the requirement of that theorem, while being preferable from a computational viewpoint.

computed by a fast algorithm that requires only $O(kM^{2k})$ operations. Therefore, the total time-complexity for computing $\hat{P}_{X^k_{-k}}$ becomes also $O(kM^{2k})$, instead of the $O(M^{6k})$ that direct computation would require.

4-C. Considerations for a Modified Version.

4-C.1. Context-length Selection. Both the theoretical results of Section 2-C and the complexity bounds mentioned in Subsection 4-A provide guidelines for a reasonable choice of asymptotic growth order for k with n. However, for the case where n is of moderate value, the choice of k has a considerable effect on the denoising performance. Two main considerations in selecting the value of k are:

- **Lack of Sufficient Counts.** The denoiser first counts the noise-corrupted data to obtain the $2k+1$-tuple empirical distribution, $\hat{P}_{Z^k_{-k}}(z^n)$, and then uses it to estimate the $2k+1$-tuple source distribution, $P_{X^k_{-k}}$. On the one hand, a very large k tends to render the empirical distribution, $\hat{P}_{Z^k_{-k}}(z^n)$, a less reliable estimate of the true distribution because there may not be a sufficient number of samples counted for many of the $2k+1$-tuples. As a result, the inaccuracy in $\hat{P}_{Z^k_{-k}}(z^n)$ propagates into the estimation of $P_{X^k_{-k}}$ and causes the degradation of the denoising performance. On the other hand, if the true source distribution, $P_{X^k_{-k}}$, is a priori known instead of being estimated from the noisy data, a larger k is always preferable because the denoiser estimates the source signal at each location based on more information. Therefore, there is a tradeoff in choosing k between an accurate distribution estimation and a large context for our sliding-window based denoiser. This tradeoff is clearly demonstrated by the DUDE over DMC channels [26, Section 8-A] (cf. also [21]), where it is seen that for a fixed data length, n, the denoising performance improves when k increases, but starts to degrade when k exceeds a critical value.
- **Matrix Inversion.** One key step in implementing the denoiser is the computation of the inverse of the $2k+1$-tuple channel transition matrix, $(\Pi^k_{-k})^{-1}$, which, as discussed, requires, when done brute force, $O(M^{6k})$ multiplication and summation operations. As is shown in Section 4-B, for the special case of additive noises, this complexity can be significantly reduced to be $O(kM^{2k})$, i.e., complexity essentially linear in the matrix dimension, allowing the use of significantly larger window sizes than would otherwise be practical. In any case, there seems to be no avoiding the exponential dependence on k of the required complexity (as the size of the channel matrix has such dependence on k).

4-C.2. A Modified Denoiser. With the above two points in mind, we now introduce a modified version of the original denoiser for one-dimensional data array. The extension of the proposed modified denoiser to multi-dimensional data arrays is

also possible after necessary modifications. The basic idea of the modified denoiser is as follows: Suppose that the original denoiser is designed for a context-length parameter k. The modified denoiser first starts with a smaller context-length parameter k', where $k' < k$, and, as is in the original scheme, obtains an empirical estimate of the $2k'+1$-tuple distribution of the noisy data, $\hat{P}_{Z_{-k'}^{k'}}[z^n]$. The value of k' is selected such that: 1. There are sufficient counts for the estimate of the empirical distribution $\hat{P}_{Z_{-k'}^{k'}}$ to be reliable; 2. The inverse of the $2k'+1$-tuple channel transition matrix, $(\Pi_{-k'}^{k'})^{-1}$, can be computed with a moderate amount of computational effort. The denoiser then obtains an estimate of the $2k'+1$-tuple source distribution as

$$\hat{P}_{X_{-k'}^{k'}} = \hat{P}_{X_{-k'}^{k'}}[z^n] = \left(\Pi_{-k'}^{k'}\right)^{-T} \cdot \hat{P}_{Z_{-k'}^{k'}}[z^n]. \tag{49}$$

Next, the denoiser proceeds to estimate the $2k+1$-tuple source distribution, $P_{X_{-k}^{k}}$, by extending $\hat{P}_{X_{-k'}^{k'}}$ from both left and right sides, assuming that the source signal is a *Markov process of order no greater than* $2k'$. More specifically,

$$\hat{P}_{X_{-k}^{k}}(x_{-k}^{k}) = \hat{P}_{X_{-k'}^{k'}}\left(x_{-k'}^{k'}\right) \prod_{i=1}^{k-k'} \left[\hat{P}_{X_{-k'}^{k'}}\left(x_{k'+i}|x_{-k'+i}^{k'+i-1}\right) \hat{P}_{X_{-k'}^{k'}}\left(x_{-k'-i}|x_{-k'-i+1}^{k'-i}\right)\right], \tag{50}$$

where the first term in the square brackets denotes the conditional distribution of a symbol given a $2k'$-tuple on its left, and the second denotes the conditional distribution of a symbol given a $2k'$-tuple on its right, both as induced by the distribution on a $2k'+1$-tuple $\hat{P}_{X_{-k'}^{k'}}$. The denosing algorithm in (10) is now modified to be

$$\hat{X}_i(z^n) = \arg\min_{\hat{x}} \sum_a \Lambda(a,\hat{x}) \left[\sum_{x_{-k}^{k}:x_0=a} \hat{P}_{X_{-k}^{k}}(x_{-k}^{k}) P_{N_{-k}^{k}}(z_{i-k}^{i+k} \ominus x_{-k}^{k})\right], \tag{51}$$

i.e., we use $\hat{P}_{X_{-k}^{k}}$ as defined in (50) in lieu of $\hat{P}_{Z_{-k}^{k}}^{T} \cdot \left(\Pi_{-k}^{k}\right)^{-1}$. The idea is that the denoiser first achieves an accurate estimate of the $2k'+1$-tuple source distribution with a smaller k' that overcomes the problem of lacking sufficient counts when the larger k is implemented directly. Secondly, the actual denoising is implemented with the larger k by extending the $2k'+1$-tuple source distribution into the needed $2k+1$-tuple source distribution. This is the modified denoiser's way of handling the conflict between an accurate distribution estimation and a large context for the sliding-window denoising. While initial experimentation indicates that this modified scheme can significantly improve denoising over the original one, basic theoretical questions are still under investigation. For example, it is not clear whether for a fixed k' there exists a way to increase k with n such that universality will be guaranteed at least with respect to the class of $2k'$-th order Markov processes. Recent results for the filtering (causal denoising) problem [18] seem to hint that this will indeed be the case, at least under a mild positivity assumption on the Markov transition kernel.

5. Experimental Results and Discussion. In this section we report on experimental results obtained by applying the proposed denoisers to data sets corrupted by a burst-noise channel [10], which is often encountered in practice.

5-A. 1D Denoising.

5-A.1. The 1D Burst Noise Channel Model. The noise sequence $\{N_i\}$ in 1D burst noise channel can be modeled as a finite-state hidden-Markov-process (FS-HMP). At each time i, the said FS-HMP is characterized by a channel state, S_i, where $\{S_i\}$ is an irreducible, aperiodic and stationary Markov chain with a finite state-space $\mathcal{C} = \{1, \ldots, C\}$ and a state transition probability matrix P_s. The noise components are independent given the state sequence. To each channel state corresponds a different noise distribution p_c, where $p_c(m) = P(N_i = m | S_i = c)$, $m \in \mathcal{A}$, and $c \in \mathcal{C}$. Also given is $\pi(c)$, which denotes the stationary distribution of the cth channel state, so that $P_s^T \pi = \pi$. Therefore, the $2k+1$-tuple distribution of N_{-k}^k can be expressed as

$$P(N_{-k}^k = n_{-k}^k) = \sum_{s_{-k}^k} P(N_{-k}^k = n_{-k}^k | S_{-k}^k = s_{-k}^k) P(S_{-k}^k = s_{-k}^k) \tag{52}$$

$$= \sum_{s_{-k}^k} p_{s_{-k}}(n_{-k}) \pi(s_{-k}) \prod_{i=-k+1}^{k} p_{s_i}(n_i) P_s(s_{i-1}, s_i). \tag{53}$$

A typical FS-HMP generates a burst-like noise process because the channel propagates through C different channel states, each having some persistent memory and being characterized by a different noise distribution. We assume the channel parameters are such that Assumption 1 is satisfied, which can be shown to hold for "most" points of the parameter space [6]. The burst noise channel becomes memoryless if and only if $P_s(c', c) = P_s(c'', c)$, $\forall c, c', c'' \in \mathcal{C}$. In this case, the channel becomes an equivalent DMC, i.e., with i.i.d. additive noise components distributed as $\sum_{c=1}^{C} \pi(c) p_c$.

5-A.2. 1D Denoising Performance. We implemented our denoiser for a binary burst noise channel for which the noise process is a binary HMP with two channel states, i.e., $M = C = 2$, which is the well-known Gilbert-Elliot Channel [10]. The channel state "1" corresponds to a "Good" binary symmetric channel (BSC) with a crossover probability, $p_1(1) \triangleq \varepsilon_G$, while the channel state "2" corresponds to a "Bad" BSC with a crossover probability, $p_2(1) \triangleq \varepsilon_B$. $0 \le \varepsilon_G < \varepsilon_B \le 1$. The state transition probability from the "Good" channel to the "Bad" channel is denoted P_{GB} while the transition probability from the "Bad" to the "Good" is denoted P_{BG}. Therefore, the vector, $\beta_C \triangleq [\varepsilon_G, \varepsilon_B, P_{GB}, P_{BG}]$, completely characterizes the binary burst noise channel. A total number of four different burst noise channels are considered in the simulation: (C1) $\beta_{C1} = [0.01\ 0.2\ 0.01\ 0.1]$; (C2) $\beta_{C2} = [0.01\ 0.8\ 0.01\ 0.1]$; (C3) $\beta_{C3} = [0.01\ 0.2\ 0.01\ 0.01]$; (C4) $\beta_{C4} = [0.01\ 0.8\ 0.01\ 0.01]$. Compared with channels C1 and C2 that have $P_{BG} = 0.1$, channels C3 and C4 with $P_{BG} = 0.01$ have a higher

tendency to persist when in a "Bad" channel state. Compared with channels C1 and C3 that have $\varepsilon_B = 0.2$, channels C2 and C4 with $\varepsilon_B = 0.8$ have more noisy "Bad" channels. The source signal is a first-order symmetric binary Markov process with probability of transition from one state to the other $p = 0.01$. In each experiment, only one realization of the source and the noise is generated and the data sequence length, n, is 10^6.

Table 5-A.2 shows the bit error rate (BER), expressed as a multiple of δ, of the denoised signal obtained by different denoising schemes, where δ is the raw BER before any denoising. The denoising schemes are listed as follows:

- **Median Filter.** The $2k+1$ sliding-window median filter with binary input and output alphabets decodes the source symbol at each location by a majority vote from the value of the observed symbol at that location and the values of $2k$ observed symbols in its context. The minimum BER obtained through the median filtering is shown in Table 1 and the associated best filter order k in each case is also shown in the bracket following the BER.
- **Genie-aided**$[k]$**.** The genie-aided $2k+1$ sliding-window denoiser decodes the source symbol using the achiever of the minimum in (12), i.e., based on knowledge of both the noisy and noise-free signals. As such, it provides the performance bound on all sliding window denoisers of order $2k+1$.
- **Proposed**$[k]$**.** Refers to the proposed $2k+1$-tuple denoiser. For $k = 4$, the original denoiser in (10) is used while for $k = 7$, the modified denoiser of Subsection 4-C.2 is used with $k' = 2$.
- **DUDE**$[k]$**.** The $2k+1$ sliding-window denoiser (DUDE) in [26] for DMC is applied here, by ignoring the memory in the burst noise process and taking the burst noise channel as a DMC with crossover probability $p_e = p_1(1)\pi(1) + p_2(1)\pi(2)$.
- **BCJR.** The BCJR-based denoiser has perfect knowledge of the source statistics and decodes the source symbol at each location based on all the observed symbols using the BCJR algorithm [2] (agglomerating the noiseless signal component and the channel state into one state). In other words, it is an implementation of the optimal distribution-dependent denoiser.

We make the following observations:

- The median filter that is easily implemented without any knowledge of the source or the channel can perform reasonably well for channels C1, C2 and C3, but fails in improving the BER for channel C4. Furthermore, the optimum values of the filter order appear quite random and a sound rule for designing the filter order seems unlikely.
- The proposed denoiser designed for the burst noise channel achieves a significant BER improvement compared with the DUDE of [26], which is applied here by ignoring the memory in the channel.

TABLE 1

Bit error rate in denoising sequences emitted by a Markov source and corrupted by a binary burst noise channel.

Denoising Schemes	C1 $\delta = 0.0269$	C2 $\delta = 0.0808$	C3 $\delta = 0.1038$	C4 $\delta = 0.4011$
Median Filter	0.1491δ[4]	0.6720δ[16]	0.1349δ[7]	1.0000δ[0]
Genie-aided[4]	0.1190δ	0.6733δ	0.1618δ	0.4291δ
Proposed[4]	0.1190δ	0.6733δ	0.1618δ	0.4298δ
DUDE[4]	0.4764δ	0.9208δ	0.4461δ	1.1653δ
Genie-aided[7]	0.0669δ	0.4975δ	0.0790δ	0.3089δ
Proposed[7]	0.0929δ	0.5371δ	0.1012δ	0.3219δ
DUDE[7]	0.4647δ	0.9084δ	0.4644δ	1.1735δ
BCJR	0.0855δ	0.2859δ	0.0790δ	0.0738δ

- The proposed denoiser essentially attains the performance of the best genie-aided sliding-window denoiser for $k = 4$. For $k = 7$, the modified sliding-window denoiser provides a consistent BER improvement. However, a notable BER gap is observed as compared with the genie-aided denoiser. This is because even assuming the modified denoiser can achieve the same performance as that of the actual proposed denoiser with the same context length, the data block length, $n(= 10^6)$, becomes insufficient for the applied context length, $k(= 7)$.
- The best genie-aided sliding-window denoiser for the semi-stochastic setting can outperform the BCJR-based denoiser that is the optimal denoiser for the stochastic setting, e.g., for channel C1. However, for channels that have longer consecutive burst errors, e.g., channel C2 and C4, there is a notable BER gap between the sliding-window denoiser and the BCJR-based denoiser that operates optimally based on all the data and complete knowledge of the source and channel. This can be explained by the fact that the BCJR-based denoiser implemented via the backward-forward recursions, jointly estimates the source symbol and the channel state at each location. The sliding-window based denoiser, on the other hand, determines the decoding rule at each location inevitably by mixing the statistics of "Good" and "Bad" states since, for the small window-lengths used, it is unable to "lock in on" the true state, as the BCJR-based denoiser that has access to all the noisy data is typically able to do.

5-B. Binary Image Denoising.

5-B.1. 2D Burst Noise Channel Model. The noise field[6] $N_{m\times n} = \{N_{i,j}\}_{(i,j)\in V_{m\times n}}$ in a two-dimensional burst noise channel can be modelled as a finite-state hidden- Markov-Random-Field (FS-HMRF). This field is characterized by $S_{m\times n}$, the channel state field, i.e., $S_{i,j}$ denotes the channel state at the location $(i,j) \in V_{m\times n}$ and takes a value from the finite state-space $\mathcal{C} = \{1,\dots,C\}$. $S_{m\times n}$ is a MRF, which means that the conditional distribution of $S_{i,j}$ given the channel states at all the other locations in the 2D data array satisfies:

(54)
$$P\left(S_{i,j} = s_{i,j} | S\left(V_{m\times n} \setminus (i,j)\right) = s\left(V_{m\times n} \setminus (i,j)\right)\right) = P(S_{i,j} = s_{i,j} | S_{\mathcal{N}_{i,j}} = s_{\mathcal{N}_{i,j}}),$$

where $\mathcal{N}_{i,j}$ is the set of points neighboring (i,j), and the neighboring relationship has the following two properties: (1) $(i,j) \notin \mathcal{N}_{i,j}$; (2)$(i,j) \in \mathcal{N}_{k,l} \Leftrightarrow (k,l) \in \mathcal{N}_{i,j}$. The noise components are independent given the state values. As in the one dimensional case, each channel state is associated with a noise distribution, p_c, where $p_c(a) = P(N_{i,j} = a | S_{i,j} = c)$, $a \in \mathcal{A}$, and $c \in \mathcal{C}$.

The joint distribution for the MRF in (54) is well known to be given by the Gibbs distribution [17, 9, 15], which takes the form:

$$P(S_{m\times n} = s_{m\times n}) = Z^{-1} \exp^{-\frac{1}{T}U(s_{m\times n})}, \tag{55}$$

where $Z = \sum_{s_{m\times n}} \exp^{-\frac{1}{T}U(s_{m\times n})}$ is the normalization factor called the partition function, T is referred to as the temperature which we shall take to be 1, and $U(s_{m\times n})$ is the energy function. For example, for the '8-nearest-neighbor' neighborhood $\mathcal{N}_{i,j} = \{(i, j\pm 1), (i\pm 1, j), (i\pm 1, j\pm 1)\}$, the energy function can be written as

$$U(s_{m\times n}) = \sum_{(i,j)} V_1(s_{i,j}) + \sum_{(i,j)} \sum_{(k,l)\in\mathcal{N}_{i,j}} V_2(s_{i,j}, s_{k,l}), \tag{56}$$

where V_1 and V_2 are clique potential functions. The conditional distribution in (54) can then be brought to the form

$$P(S_{i,j} = s_{i,j} | S_{\mathcal{N}_{i,j}} = s_{\mathcal{N}_{i,j}}) = \frac{\exp^{-\left[V_1(s_{i,j}) + \sum_{(i,j)} \sum_{(k,l)\in\mathcal{N}_{i,j}} V_2(s_{i,j}, s_{k,l})\right]}}{\sum_{s_{i,j}} \exp^{-\left[V_1(s_{i,j}) + \sum_{(i,j)} \sum_{(k,l)\in\mathcal{N}_{i,j}} V_2(s_{i,j}, s_{k,l})\right]}}. \tag{57}$$

5-B.2. 2D Denoising Performance. We have implemented our denoiser for a burst noise channel for which the noise process is a binary HMRF with two channel states, i.e., $M = C = 2$. The channel state "1" corresponds to a "Good" BSC with

[6]Recall notation for 2D data from Subsection 3-A which we use throughout this subsection, with the exception of using a double index (i,j) (one for each coordinate), rather than a single (two-dimensional) index, to denote a location in the image.

TABLE 2
Bit error rate in denoising images corrupted by a two-dimensional binary burst noise channel.

Images	Denoising Schemes	C1 $\delta = 0.0337$	C2 $\delta = 0.1046$	C3 $\delta = 0.1117$
Shannon 1000×1000	Genie-aided	0.2077δ	0.2170δ	0.5838δ
	Proposed	0.2107δ	0.2170δ	0.6124δ
	DUDE	0.5371δ	0.4178δ	1.0090δ
	Median Filter	0.3442δ	0.2639δ	1.0143δ
	Morphological Filter	1.5282δ	0.7945δ	0.8478δ
Einstein 900×900	Genie-aided	0.8012δ	0.8202δ	0.8422δ
	Proposed	0.8012δ	0.8221δ	0.8457δ
	DUDE	0.8392δ	0.8382δ	1.0717δ
	Median Filter	4.1988δ	1.6660δ	1.7049δ
	Morphological Filter	7.0673δ	2.9791δ	2.5753δ
Lenna 256×256	Genie-aided	0.2945δ	0.2570δ	0.4702δ
	Proposed	0.2946δ	0.2696δ	0.4919δ
	DUDE	0.4189δ	0.3395δ	0.9611δ
	Median Filter	0.6243δ	0.3240δ	0.5325δ
	Morphological Filter	1.2021δ	0.8513δ	0.8087δ

a crossover probability $p_1(1) \triangleq \varepsilon_G$, while the channel state "2" corresponds to a "Bad" BSC with a crossover probability $p_2(1) \triangleq \varepsilon_B$. We assume that $V_1(s_{i,j}) = \alpha_{s_{i,j}}$ and $V_2(s_{i,j}, s_{k,l}) = 2\gamma(s_{i,j}, s_{k,l}) - 1$, where $\gamma(a, b) = 1$ if $a = b$ and zero otherwise. Therefore, the vector $\beta_C \triangleq [\varepsilon_G, \varepsilon_B, \alpha_1, \alpha_2]$ completely characterizes the burst noise channel. Three different burst noise channels are considered in the simulation: (C1) $\beta_{C1} = [0.01\ 0.2\ 0.2\ 0]$; (C2) $\beta_{C2} = [0.01\ 0.2\ 0\ 0]$; (C3) $\beta_{C3} = [0.01\ 0.8\ 0.2\ 0]$. Compared with channel C1, C2 and C3 are more noisy channels.

The source signals are three binary images: (1) a scanned copy of the first page of [24] with the size of $10^3 \times 10^3$, i.e., $m = n = 10^3$; (2) a 900×900 half-toned portrait of a famous physicist; (3) a 256×256 image of "Lenna". The binary MRF is generated by the Gibbs sampling method [17] with 50 iterations.

Table 2 shows the bit error rate (BER), expressed as a multiple of δ, of the denoised signal obtained by different denoising schemes, where δ is the raw BER before any denoising. The denoising schemes are listed as follows:

- **Genie-aided.** The genie-aided 3×3 sliding-window denoiser decodes using the achiever of the minimum in (42), with $r = 1$.
- **Proposed.** Refers to $\hat{X}_{\mathsf{univ}}^{m \times n}$, as defined in (41), with $r = 1$. In the experiment we have used an estimate of the joint distribution of a 3×3 square of noise components, taken as the empirical distribution induced by a randomly

generated $10^3 \times 10^3$ hidden MRF (independent of the noise field that corrupted the data). This estimate was taken in lieu of the true distribution of such a 3×3 square (required in the denoising rule (40)), which would be difficult to obtain precisely (ideally needing to marginalize the distribution in (55) to a 3×3 square). The inverse of the estimated channel matrix (associated with the estimated noise distribution) was used, as is, in the denoising rule, though we believe methods for regularizing the channel inverse could yield performance gains. Such methods are well developed and widely applied in statistics (cf., e.g., [5]) and communications (cf., e.g., [23]).

- **DUDE.** The 3×3 sliding-window denoiser of [20], assuming a DMC with crossover probability p_e. Because it is difficult to obtain closed-form expressions for marginal distributions of "Good" and "Bad" channels in a 2D Markov random field, i.e., $\pi(1)$ and $\pi(2)$, the equivalent raw BER, $p_e = p_1(1)\pi(1) + p_2(1)\pi(2)$, is also not available. Therefore, in the simulation, we instead take p_e to be the number of bit errors divided by the total number of bits in each observed noisy binary image.
- **Median Filter.** The 3×3 sliding-window median filter decodes by majority vote.
- **Morphological Filter.** A Morphological filter, available in MATLAB, uses a 3×3 structure element and implements the CLOSE and then the OPEN operation to the noise corrupted image.

The proposed image denoiser is observed to achieve a better BER improvement for the tested images and all the channels simulated compared with more conventional filters like the Median and Morphological filters. It also approaches the performance of the best genie-aided sliding-window based denoiser, and outperforms the DUDE in [26] that takes into account the channel crossover probability assuming it is a DMC. Portions of the noiseless image, the noisy image, the image denoised by the proposed denoiser, and the image as denoised by the DUDE of [26], are shown in Figure 1 for the experiment of the text image corrupted by channel C1, Figure 2 for the half-toned image corrupted by channel C3, and Figure 3 for the black and white image corrupted by channel C3. It is observed that the proposed denoising scheme improves not only the BER, but also the visual quality of the noise-corrupted images. Of course, this in no way indignifies the DUDE of [26], which was not designed to accommodate memory in the noise. It does, however, exemplify the gain in taking the channel memory into account. Further empirical evidence supporting this conclusion, for sources and channels of types different than those experimented with here, is reported on in [12].

6. Conclusion. Discrete denoising for channels with memory was considered, with particular focus on the case of additive noise. A sequence of denoisers that

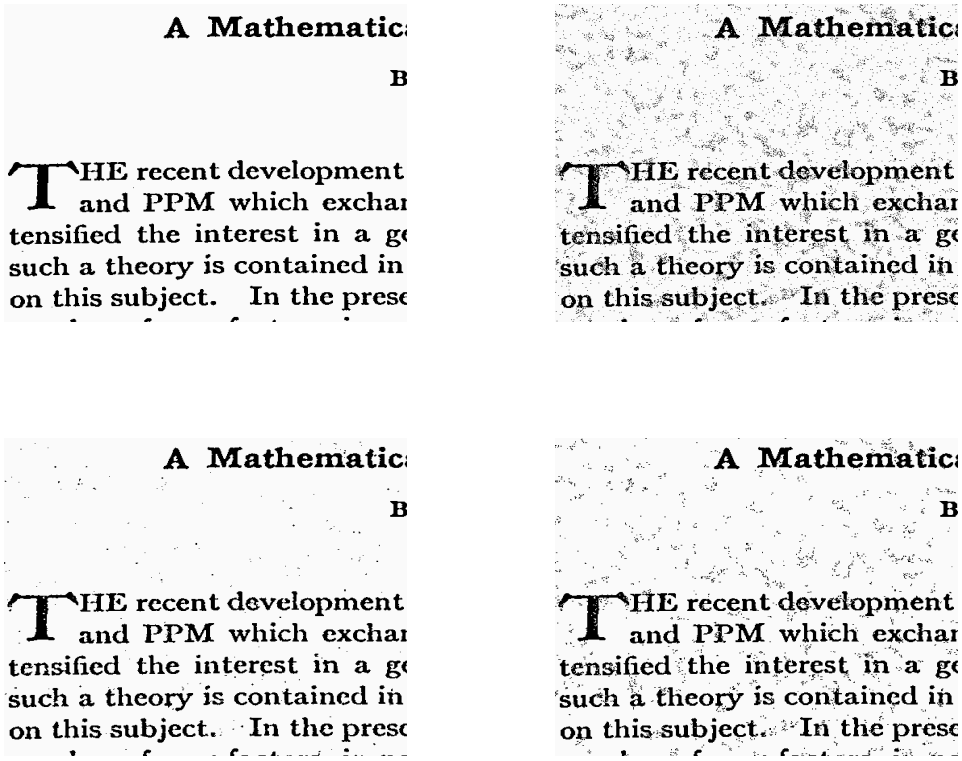

FIG. 1. *Denoising of a scanned text image. top-left: noiseless image; top-right: noisy image; bottom-left: denoised image by the proposed denoiser; bottom-right: denoised image by DUDE in [26].*

operates without knowledge of the noiseless data or its distribution was derived, and shown to be universal under a mild mixing condition on the channel noise. Algorithmic aspects were also considered, including a variation on the first scheme, which was argued likely to improve performance in practice. Experimental results for binary data corrupted by burst noise were presented, where it was found that the suggested schemes outperform current popular denoisers.

On the theoretical front, an attempt has not been made to refine the analysis beyond the asymptotics, and to get the tightest possible non-asymptotic performance bounds. For example, it is possible that a bound tighter than that in Theorem 2 (decaying faster than $\sim 1/n$) can be obtained. Furthermore, under a requirement for exponential decay of the mixing coefficients (under possibly a slightly stronger form of mixing), the bound should be improvable to exponential decay in n, similarly as was done in [7].

It will be interesting to explore further the modified denoising scheme of Section 4-C.2, from both experimental and theoretical viewpoints. It should be noted that the idea on which this scheme is based can be also applied (with obvious modifications)

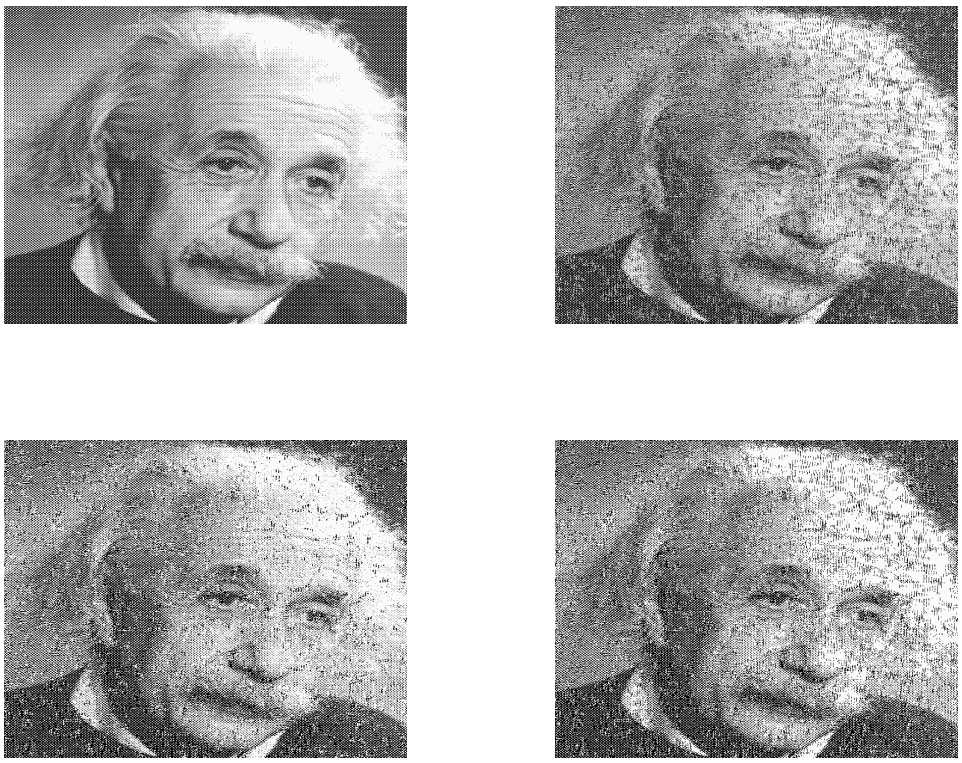

FIG. 2. *Denoising of a half-toned image. top-left: noiseless image; top-right: noisy image; bottom-left: denoised image by the proposed denoiser; bottom-right: denoised image by DUDE in [26].*

in other discrete denoising settings, including the original one of [26], and those of [4, 22]. Initial theoretical results justifying this approach are developed in [18].

Acknowledgment. The authors gratefully acknowledge helpful discussions with Erik Ordentlich, Ronny Roth, Gadiel Seroussi, Sergio Verdú, and Marcelo Weinberger on various aspects of this work.

Appendix A. Proof of Theorem 2.

In this Appendix, we prove Theorem 2 in the main text. Two facts will be needed in the proof. The first follows directly from the definition of α-mixing (recall (3)):

FACT 1. *Let $\{S_i\}$ be a process with $ES_i = 0$, $|S_i| \leq M$, and α-mixing coefficients $\{\alpha_t^{(S)}\}$. Then for all i, j*

$$|ES_iS_j| \leq M^2\alpha_{|i-j|}^{(S)}.$$

The second is:

LEMMA 1. *Let $\{V_i\}$ be a sequence of random variables satisfying $|V_i| \leq M$ and*

FIG. 3. *Denoising of a black and white image. top-left: noiseless image; top-right: noisy image; bottom-left: denoised image by the proposed denoiser; bottom-right: denoised image by DUDE in [26].*

$|EV_iV_j| \le R(i-j)$ *for all* i, j, *where* $R(\cdot)$ *satisfies* $C \stackrel{\triangle}{=} \sum_{i=1}^{\infty} R(i) < \infty$. *Then*

$$Var\left(\frac{1}{n}\sum_{i=1}^{n} V_i\right) \le \frac{M^2 + 2C}{n}.$$

Proof.

$$\begin{aligned}
\mathrm{Var}\left(\frac{1}{n}\sum_{i=1}^{n} V_i\right) &\le E\left[\left(\frac{1}{n}\sum_{i=1}^{n} V_i\right)^2\right] \\
&= \frac{1}{n^2} E\left[\sum_{i=1}^{n} V_i^2 + 2\sum_{i=2}^{n}\sum_{j=1}^{i-1} V_iV_j\right] \\
&\le \frac{1}{n^2}\left[nM^2 + 2\sum_{i=2}^{n}\sum_{j=1}^{i-1} R(i-j)\right] \\
&\le \frac{1}{n^2}\left[nM^2 + 2n\sum_{j=1}^{\infty} R(i-j)\right] \\
&= \frac{M^2 + 2C}{n}.
\end{aligned}$$

□

Proof of Theorem 2. Assume throughout this proof a fixed x^n. For each a, u_{-k}^k,

$$
\begin{aligned}
&\left|\hat{q}_k(Z^n)[a,u_{-k}^k]-q_k(z^n,x^n)[a,u_{-k}^k]\right| \\
&=\left|\left[\sum_{x_{-k}^k:x_0=a}[\hat{P}_{Z_{-k}^k}[Z^n]^T\cdot\left(\Pi_{-k}^k\right)^{-1}](x_{-k}^k)P_{N_{-k}^k}(u_{-k}^k\ominus x_{-k}^k)\right]-q_k(Z^n,x^n)[a,u_{-k}^k]\right| \\
&=\left|\left[\sum_{x_{-k}^k:x_0=a}\left[\sum_{v_{-k}^k}\hat{P}_{Z_{-k}^k}[Z^n](v_{-k}^k).\left(\Pi_{-k}^k\right)^{-1}[v_{-k}^k,x_{-k}^k]\right]P_{N_{-k}^k}(u_{-k}^k\ominus x_{-k}^k)\right]\right. \\
&\quad\left.-\frac{1}{n-2k}\sum_{i=k+1}^{n-k}\mathbf{1}_{\left\{x_i=a,Z_{i-k}^{i+k}=u_{-k}^k\right\}}\right| \\
&=\left|\left[\sum_{y_{-k}^k:y_0=a}\left[\sum_{v_{-k}^k}\frac{\sum_{i=k+1}^{n-k}\mathbf{1}_{\left\{Z_{i-k}^{i+k}=v_{-k}^k\right\}}}{n-2k}\cdot\left(\Pi_{-k}^k\right)^{-1}[v_{-k}^k,y_{-k}^k]\right]P_{N_{-k}^k}(u_{-k}^k\ominus y_{-k}^k)\right]\right. \\
&\quad\left.-\sum_{y_{-k}^k:y_0=a}\sum_{i=k+1}^{n-k}\frac{\mathbf{1}_{\left\{x_{i-k}^{i+k}=y_{-k}^k,Z_{i-k}^{i+k}=u_{-k}^k\right\}}}{n-2k}\right| \\
&\le\frac{1}{n-2k}\sum_{y_{-k}^k:y_0=a}\left|\sum_{i=k+1}^{n-k}\left[\sum_{v_{-k}^k}\mathbf{1}_{\left\{Z_{i-k}^{i+k}=v_{-k}^k\right\}}\cdot\left(\Pi_{-k}^k\right)^{-1}[v_{-k}^k,y_{-k}^k]\right]\right. \\
&\quad\left.\cdot P_{N_{-k}^k}(u_{-k}^k\ominus y_{-k}^k)-\mathbf{1}_{\left\{x_{i-k}^{i+k}=y_{-k}^k,Z_{i-k}^{i+k}=u_{-k}^k\right\}}\right|
\end{aligned}
\tag{58}
$$

where $\left(\Pi_{-k}^k\right)^{-1}[v_{-k}^k,x_{-k}^k]$ denotes the (v_{-k}^k,x_{-k}^k)-th element of $\left(\Pi_{-k}^k\right)^{-1}$. Now, for each i, u_{-k}^k and y_{-k}^k,

$$
\begin{aligned}
&E\left\{\left[\sum_{v_{-k}^k}\mathbf{1}_{\left\{Z_{i-k}^{i+k}=v_{-k}^k\right\}}\cdot\left(\Pi_{-k}^k\right)^{-1}[v_{-k}^k,y_{-k}^k]\right]P_{N_{-k}^k}(u_{-k}^k\ominus y_{-k}^k)\right\} \\
&=P_{N_{-k}^k}(u_{-k}^k\ominus y_{-k}^k)\sum_{v_{-k}^k}E\mathbf{1}_{\left\{Z_{i-k}^{i+k}=v_{-k}^k\right\}}\cdot\left(\Pi_{-k}^k\right)^{-1}[v_{-k}^k,y_{-k}^k] \\
&=P_{N_{-k}^k}(u_{-k}^k\ominus y_{-k}^k)\sum_{v_{-k}^k}E\left[\sum_{s_{-k}^k}\mathbf{1}_{\left\{x_{i-k}^{i+k}=s_{-k}^k,Z_{i-k}^{i+k}=v_{-k}^k\right\}}\right]\cdot\left(\Pi_{-k}^k\right)^{-1}[v_{-k}^k,y_{-k}^k] \\
&=P_{N_{-k}^k}(u_{-k}^k\ominus y_{-k}^k)\sum_{v_{-k}^k}\sum_{s_{-k}^k}\mathbf{1}_{\left\{x_{i-k}^{i+k}=s_{-k}^k\right\}}P_{N_{-k}^k}(v_{-k}^k\ominus s_{-k}^k)\cdot\left(\Pi_{-k}^k\right)^{-1}[v_{-k}^k,y_{-k}^k] \\
&=P_{N_{-k}^k}(u_{-k}^k\ominus y_{-k}^k)\sum_{s_{-k}^k}\mathbf{1}_{\left\{x_{i-k}^{i+k}=s_{i-k}^{i+k}\right\}}\sum_{v_{-k}^k}\Pi_{-k}^k[s_{-k}^k,v_{-k}^k]\cdot\left(\Pi_{-k}^k\right)^{-1}[v_{-k}^k,y_{-k}^k]
\end{aligned}
$$

$$
\begin{aligned}
&= P_{N^k_{-k}}(u^k_{-k} \ominus y^k_{-k}) \sum_{s^k_{-k}} \mathbf{1}_{\{x^{i+k}_{i-k}=s^k_{-k}\}} \cdot \mathbf{1}_{\{y^k_{-k}=s^k_{-k}\}} \\
&= P_{N^k_{-k}}(u^k_{-k} \ominus y^k_{-k}) \mathbf{1}_{\{x^{i+k}_{i-k}=y^k_{-k}\}} \\
&= \Pr(Z^{i+k}_{i-k} = u^k_{-k}) \mathbf{1}_{\{x^{i+k}_{i-k}=y^k_{-k}\}} \\
&= E\mathbf{1}_{\{x^{i+k}_{i-k}=y^k_{-k}, Z^{i+k}_{i-k}=u^k_{-k}\}}. \qquad (59)
\end{aligned}
$$

Evidently, defining

$$
\begin{aligned}
T_i = T_i(N^{i+k}_{i-k}, u^k_{-k}, y^k_{-k}) = &\left[\sum_{v^k_{-k}} \mathbf{1}_{\{Z^{i+k}_{i-k}=v^k_{-k}\}} \cdot \left(\Pi^k_{-k}\right)^{-1}[v^k_{-k}, y^k_{-k}]\right] \\
&\cdot P_{N^k_{-k}}(u^k_{-k} \ominus y^k_{-k}) - \mathbf{1}_{\{x^{i+k}_{i-k}=y^k_{-k}, Z^{i+k}_{i-k}=u^k_{-k}\}},
\end{aligned} \qquad (60)
$$

the sum over i in (58) is

$$
\sum_{i=k+1}^{n-k} T_i,
$$

which is a sum of *zero mean* variables, bounded in magnitude by $\| \left(\Pi^k_{-k}\right)^{-1} \| + 1$, the i-th variable being a deterministic function of N^{i+k}_{i-k}. It is therefore also clear that the mixing coefficients of the process $\{T_i\}$, $\{\alpha^{(T)}_t\}$, satisfy $\alpha^{(T)}_t \le \alpha^{(N)}_{t-2k}$ for all $t \ge 2k$ (and trivially $\alpha^{(T)}_t \le 1$ for $t < 2k$). Summarizing, $\{T_i\}$ is a sequence of zero-mean variables, bounded by $\| \left(\Pi^k_{-k}\right)^{-1} \| + 1$, with α-mixing coefficients satisfying $\alpha^{(T)}_t \le \alpha^{(N)}_{t-2k}$. Combined with Fact 1 and Lemma 1 this implies

$$
\begin{aligned}
&\operatorname{Var}\left(\frac{1}{n-2k}\sum_{i=k+1}^{n-k} T_i\right) \\
&\le \frac{\left(\| \left(\Pi^k_{-k}\right)^{-1} \| + 1\right)^2 + 2\left(2k + \sum_{t=1}^{\infty} \alpha_t\right)\left(\| \left(\Pi^k_{-k}\right)^{-1} \| + 1\right)^2}{n-2k} \qquad (61) \\
&= \frac{(4k + 1 + 2\sum_{t=1}^{\infty} \alpha_t)\left(\| \left(\Pi^k_{-k}\right)^{-1} \| + 1\right)^2}{n-2k}, \qquad (62)
\end{aligned}
$$

where in (61) and on $\alpha_t = \alpha^{(N)}_t$. Applying Chebychev's inequality gives

$$
\Pr\left(\left|\frac{1}{n-2k}\sum_{i=k+1}^{n-k} T_i\right| \ge \varepsilon\right) \le \frac{(4k + 1 + 2\sum_{t=1}^{\infty} \alpha_t)\left(\| \left(\Pi^k_{-k}\right)^{-1} \| + 1\right)^2}{\varepsilon^2(n-2k)}. \qquad (63)
$$

Now, in terms of the random variables $T_i(N^{i+k}_{i-k}, u^k_{-k}, y^k_{-k})$, the expression in (58) becomes

$$
\sum_{y^k_{-k}: y_0 = a} \left|\frac{1}{n-2k}\sum_{i=k+1}^{n-k} T_i(N^{i+k}_{i-k}, u^k_{-k}, y^k_{-k})\right| \qquad (64)
$$

and

$$\begin{aligned}
&\Pr\left(\sum_{y_{-k}^k: y_0=a}\left|\frac{1}{n-2k}\sum_{i=k+1}^{n-k} T_i(N_{i-k}^{i+k}, u_{-k}^k, y_{-k}^k)\right| \geq \varepsilon\right) \\
&\leq \sum_{y_{-k}^k: y_0=a} \Pr\left(\left|\frac{1}{n-2k}\sum_{i=k+1}^{n-k} T_i(N_{i-k}^{i+k}, u_{-k}^k, y_{-k}^k)\right| > \varepsilon/M^{2k}\right) \\
&\leq M^{2k}\frac{(4k+1+2\sum_{t=1}^{\infty}\alpha_t)\left(\|\left(\Pi_{-k}^k\right)^{-1}\|+1\right)^2}{(\varepsilon/M^{2k})^2(n-2k)} \qquad (65) \\
&= M^{6k}\frac{(4k+1+2\sum_{t=1}^{\infty}\alpha_t)\left(\|\left(\Pi_{-k}^k\right)^{-1}\|+1\right)^2}{\varepsilon^2(n-2k)}, \qquad (66)
\end{aligned}$$

where (65) follows by applying (63) on each summand. Combining (66) with (58) gives

$$\begin{aligned}
&\Pr\left(\left|\hat{q}_k(Z^n)[a, u_{-k}^k] - q_k(Z^n, x^n)[a, u_{-k}^k]\right| \geq \varepsilon\right) \qquad (67) \\
&\leq M^{6k}\frac{(4k+1+2\sum_{t=1}^{\infty}\alpha_t)\left(\|\left(\Pi_{-k}^k\right)^{-1}\|+1\right)^2}{\varepsilon^2(n-2k)}.
\end{aligned}$$

So

$$\begin{aligned}
&\Pr\left(\|\hat{q}_k(Z^n) - q_k(Z^n, x^n)\| \geq \varepsilon\right) \\
&\leq \sum_{a, u_{-k}^k} \Pr\left(\left|\hat{q}_k(Z^n)[a, u_{-k}^k] - q_k(Z^n, x^n)[a, u_{-k}^k]\right| \geq \varepsilon\right) \qquad (68) \\
&\leq M^{2k+2}\cdot M^{6k}\frac{(4k+1+2\sum_{t=1}^{\infty}\alpha_t)\left(\|\left(\Pi_{-k}^k\right)^{-1}\|+1\right)^2}{\varepsilon^2(n-2k)}. \qquad (69)
\end{aligned}$$

□

Appendix B. Proof of Theorem 4. In this Appendix, we first prove Theorem 4 (of Section 4-B) and then derive the time-complexity required for computing the "generalized" Fourier and inverse Fourier transforms, both used for the efficient algorithm presented in Section 4-B (recall (48) therein).

Consider a matrix $A_n \in \mathbb{R}^{N\times N}$, with $N = M^n$. The element of A_n at the ith row and the jth column is denoted as $A_n(i, j)$, $i = 0, \ldots, M^n - 1, j = 0, \ldots, M^n - 1$. We represent the row and the column index of each matrix element of A_n by an equivalent vector, $\underline{i}$ and $\underline{j}$, respectively. Each element of $\underline{i}$ and $\underline{j}$ takes values from a finite alphabet, $\mathcal{A} = \{0, \ldots, M-1\}$ and has the following correspondence with i and j: $\underline{i} = [\underline{i}_0, \ldots, \underline{i}_{n-1}], \underline{j} = [\underline{j}_0, \ldots, \underline{j}_{n-1}]$, and $i = \sum_{k=0}^{n-1}\underline{i}_k M^k, j = \sum_{k=0}^{n-1}\underline{j}_k M^k$ (known as a lexicographic correspondence). We thus have $A_n(i, j) = A_n(\underline{i}, \underline{j})$. We assume further that A_n is lexicographically circulant, i.e., $A_n(\underline{i}, \underline{j}) = A_n(\underline{i}', \underline{j}')$, if $\underline{j}_k \ominus \underline{i}_k = \underline{j}_k' \ominus \underline{i}_k', \ \forall k = 0, \ldots, n-1$, where $\ominus$ denotes modulo-M subtraction. It is

noted that the $2k+1$-tuple channel transition matrix, Π_{-k}^{k}, defined in (4) for additive noise channels with modulo-M addition, has this property. Therefore, we prove the result in this Appendix for a general lexicographically circulant matrix, A_n, which will then be directly applicable to the channel transition matrix of our original interest.

We begin by stating two lemmas concerning the cyclic decomposition of lexicographically circulant matrices, and the special structure of the "generalized" Fourier transform matrix, $\mathcal{H}_n$ defined in (47).

LEMMA 2. *If A_n is lexicographically circulant, it has the following cyclic decomposition:*

$$A_n = \begin{bmatrix} B_{n-1}^{(0)} & B_{n-1}^{(M-1)} & \cdots & \cdots & B_{n-1}^{(1)} \\ B_{n-1}^{(1)} & B_{n-1}^{(0)} & B_{n-1}^{(M-1)} & \cdots & B_{n-1}^{(2)} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ B_{n-1}^{(M-2)} & \cdots & B_{n-1}^{(1)} & B_{n-1}^{(0)} & B_{n-1}^{(M-1)} \\ B_{n-1}^{(M-1)} & \cdots & \cdots & B_{n-1}^{(1)} & B_{n-1}^{(0)} \end{bmatrix} \tag{70}$$

where $B_{n-1}^{(k)}$ is a lexicographically circulant matrix of dimension $M^{n-1} \times M^{n-1}$, $k = 0, \ldots, M-1$.

Proof. Let $A_n[l,m]$ denote the $M^{n-1} \times M^{n-1}$ sub-matrix of A_n defined by $(A_n[l,m])(\underline{i},\underline{j}) = A_n(\underline{i}',\underline{j}')$, where $\underline{i}' = [\underline{i},l], \underline{j}' = [\underline{j},m], l = 0,\ldots,M-1, m = 0,\ldots,M-1$. The fact that A_n is lexicographically circulant implies that $A_n[l,m]$ depends on l,m only through $l \ominus m$. Thus, A_n decomposes according to (70), with $B_{n-1}^{(l\ominus m)} = A_n[l,m]$. □

Consider the "generalized" Fourier transform matrix, $\mathcal{H}_n$ defined in (47), i.e., $\mathcal{H}_n = \mathcal{F}_M^{\otimes n}$, where $\otimes n$ denotes the n-th tensor-power, and $\mathcal{F}_M$ is the $M \times M$ Fourier matrix.

LEMMA 3. *$\mathcal{H}_n$ is unitary, i.e., $\mathcal{H}_n^{-1} = \mathcal{H}_n^* \ \forall n$, where $*$ denotes conjugate transpose.*

Proof. For $n = 1$ this is the well-known property of the Fourier matrix. The case $n > 1$ easily follows by induction, using the tensor product properties $(A \otimes B)^* = A^* \otimes B^*$ and $(A \otimes B)^{-1} = A^{-1} \otimes B^{-1}$. □

We are now ready to prove Theorem 4 for a general lexicographically circulant matrix A_n. It is noted that the proof for the binary alphabet, $M = 2$, has been given in [16]. To recapitulate, we need to prove the following:

1. $\mathcal{H}_n$ diagonalizes A_n, i.e., $A_n = \mathcal{H}_n^* \Lambda_n \mathcal{H}_n$, where Λ_n is diagonal.
2. $\mathrm{diag}(\Lambda_n) = \mathcal{H}_n \cdot A_n(:,0)$, where $A_n(:,0)$ denotes the first column of A_n.

Proof. Consider first $n = 1$. In this case, $\mathcal{H}_1 = \mathcal{F}_M$ and the assertions are well known properties of circulant matrices (cf., e.g., [14]). Suppose that the theorem is true for a lexicographically circulant matrix, A_{n-1}, of size $M^{n-1} \times M^{n-1}$. By Lemma 2, $B_{n-1}^{(k)}$ is lexicographically circulant and of size $M^{n-1} \times M^{n-1}$, therefore,

$B_{n-1}^{(k)} = \mathcal{H}_{n-1}^* \Lambda_{n-1}^{(k)} \mathcal{H}_{n-1}$, for $k = 1, \ldots, M-1$, and $\mathrm{diag}(\Lambda_{n-1}^{(k)}) = \mathcal{H}_{n-1} \cdot B_{n-1}^{(k)}(:, 0)$. Then

$$(\mathcal{H}_n \cdot A_n)[l, m] = \sum_{k=0}^{M-1} \mathcal{F}_M(l, k) H_{n-1} \cdot A_n[k, m] \tag{71}$$

$$= \sum_{k=0}^{M-1} \mathcal{F}_M(l, k) H_{n-1} \cdot B_{n-1}^{(k \ominus m)} \tag{72}$$

$$= \sum_{k=0}^{M-1} \mathcal{F}_M(l, k) H_{n-1} \cdot \left(\mathcal{H}_{n-1}^* \Lambda_{n-1}^{(k \ominus m)} \mathcal{H}_{n-1} \right) \tag{73}$$

$$= \sum_{k=0}^{M-1} \mathcal{F}_M(l, k) \Lambda_{n-1}^{(k \ominus m)} \cdot \mathcal{H}_{n-1} \tag{74}$$

$$= \left(\sum_{k=0}^{M-1} \mathcal{F}_M(l, k \ominus m) \Lambda_{n-1}^{(k \ominus m)} \right) \mathcal{F}_M(l, m) \mathcal{H}_{n-1} \tag{75}$$

$$= \underbrace{\sum_{k=0}^{M-1} \mathcal{F}_M(l, k) \Lambda_{n-1}^{(k)}}_{\Lambda_n[l,l]} \cdot \mathcal{H}_n[l, m] \tag{76}$$

Therefore, $\mathcal{H}_n \cdot A_n = \Lambda_n \cdot \mathcal{H}_n$. Since $\mathcal{H}_n^{-1} = \mathcal{H}_n^*$, $A_n = \mathcal{H}_n^* \Lambda_n \mathcal{H}_n$ is established. Furthermore, from (76),

$$\mathrm{diag}(\Lambda_n[l, l]) = \sum_{k=0}^{M-1} \mathcal{F}_M(l, k) \mathrm{diag}(\Lambda_{n-1}^{(k)}) \tag{77}$$

$$= \sum_{k=0}^{M-1} \mathcal{F}_M(l, k) \mathrm{diag}\left(\mathcal{H}_{n-1} \cdot B_{n-1}^{(k)}(:, 0) \right) \tag{78}$$

$$= \mathcal{H}_n[l, :] \cdot A_n(:, 0). \tag{79}$$

Therefore, $\mathrm{diag}(\Lambda_n) = \mathcal{H}_n \cdot A_n(:, 0)$, completing the induction and the proof. □

Finally, we provide the time-complexity analysis for computing the "generalized" Fourier transform and inverse Fourier transform, i.e., $\mathbf{H}_n(X) = \mathcal{H}_n \cdot X$, and $\mathbf{H}_n^{\mathrm{inv}}(X) = \mathcal{H}_n^* \cdot X$, which are used in the efficient algorithm presented in Section 4-B. Because of the special structure of $\mathcal{H}_n$ and $\mathcal{H}_n^*$ we can develop a fast algorithm for computing the matrix transforms.

Consider first $\mathbf{H}_n(X) = \mathcal{H}_n \cdot X$ and let $\mathcal{C}_n$ denote the time-complexity associated with this computation. The fast algorithm first considers $X^T = [X_0^T, \ldots, X_{M-1}^T]$, where X_l is of size $M^{n-1} \times 1$, for $l = 0, \ldots, M-1$. Since $\mathcal{H}_n[l, m] = \mathcal{F}_M(l, m) \mathcal{H}_{n-1}$, we can first compute $\mathcal{H}_{n-1} \cdot X_l$, each having a time-complexity of $\mathcal{C}_{n-1}$. Then $\mathcal{H}_n \cdot X$ can be obtained by multiplying the Fourier matrix coefficients, $\mathcal{F}_M(l, m)$, for each $\mathcal{H}_{n-1} \cdot X_l$ and then summing up those vectors after multiplication corresponding to the same l, which requires at most $2M^{n+1}$ arithmetic operations. To sum up, the

time-complexity of this recursive algorithm satisfies

$$\mathcal{C}_n \leq M\mathcal{C}_{n-1} + 2M^{n+1}, \tag{80}$$

from which we derive $\mathcal{C}_n \leq M^{n-1}\mathcal{C}_1 + 2M^{n+1}(n-1) = O(nM^n)$, where $\mathcal{C}_1$ corresponds to the number of operations required for the standard Fourier transform for M-dimensional vectors, and can be computed with $O(M \log M)$ operations, e.g., using the well-known fast Fourier transform (FFT). A similar fast recursive algorithm, and analysis, can be carried through for the computation of $\mathbf{H}_n^{\mathtt{inv}}(X) = \mathcal{H}_n^* \cdot X$. We have thus established the following:

LEMMA 4. *Both the "generalized" Fourier transform, $\mathbf{H}_n(X) = \mathcal{H}_n \cdot X$, and the "genaralized" inverse Fourier transform, $\mathbf{H}_n^{\mathtt{inv}}(X) = \mathcal{H}_n^* \cdot X$, can be implemented with $O(nM^n)$ arithmetic operations.*

REFERENCES

[1] F. ALAJAJI AND T. FUJA, *A communication channel modeled on contagion*, IEEE Trans. Inform. Theory, IT-40(1994), pp. 2035-2041.

[2] L. R. BAHL, J. COCKE, F. JELINEK, AND J. RAVIV, *Optimal decoding of linear codes for minimizing symbol error rate*, IEEE Trans. Inform. Theory, IT-20(1974), pp. 284-287.

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

[4] A. DEMBO AND T. WEISSMAN, *Universal denoising for the finite-input-general-output channel*, IEEE Trans. Inform. Theory, to appear.

[5] H. W. ENGL, M. HANKE, AND A. NEUBAUER, *Regularization of Inverse Problems*, Kluwer Academic Publishers, Dordrecht - The Netherlands, 1996.

[6] Y. EPHRAIM AND N. MERHAV, *Hidden Markov processes*, IEEE Trans. Inform. Theory, 48:6(2002), pp. 1518–1569.

[7] G. GEMELOS, S. SIGURJÒNSSON, AND T. WEISSMAN, *Universal minimax discrete denoising under channel uncertainty*, Int. Symp. Inf. Th., p. 199, Chicago, IL, June-July 2004.

[8] G. GEMELOS, S. SIGURJÒNSSON, AND T. WEISSMAN, *Universal minimax binary image denoising under channel uncertainty*, Proc. IEEE 11th Int. Conf. on Image Processing, p. 997–1000, Singapore, October 24-27, 2004.

[9] H. O. GEORGII, *Gibbs Measures and Phase Transitions*, Walter de Gruyter, Berlin - New York, 1988.

[10] E. N. GILBERT, *Capacity of a burst-noise channel*, Bell Syst. Tech. J., 39(1960), pp. 1253-1265.

[11] R. C. GONZALEZ AND R. E. WOODS, *Digital Image Processing*, Addison Wesley, New York, 1992.

[12] C. D. GIURCĂNEANU AND B. YU, *Efficient algorithms for discrete universal denoising for channels with memory*, Proc. of Int. Symp. Inf. Th., Adelaide, Australia, September 2005.

[13] G. H. GOLUB AND C. F. VAN LOAN, *Matrix Computations*, Ed. Johns Hopkins, third edition, 1996.

[14] R. M. GRAY, *Toeplitz and Circulant Matrices*, Information Systems Laboratory Technical Report, Stanford University, 1971. Available at: http://www-ee.stanford.edu/g̃ray/toeplitz.html.

[15] X. GUYON, *Random Fields on a Network*, Springer-Verlag, New York, 1995.

[16] R. Iordache, I. Tăbus, and J. Astola, *Robust index assignment using Hadamard transform for vector quantization transmission over finite-memory contagion channels*, Circuits, Systems, and Signal Processing, 21:5(2002), p. 485–509.

[17] S. Z. Li, *Markov Random Field Modeling in Image Analysis*, Springer-Verlag, 2001

[18] T. Moon and T. Weissman, *Discrete universal filtering via hidden Markov modelling*, Proc. of Int. Symp. Inf. Th., Adelaide, Australia, September 2005.

[19] E. Ordentlich, G. Seroussi, S. Verdú, K. Viswanathan, M. Weinberger, and T. Weissman, *Channel decoding of systematically encoded unknown redundant sources*, Int. Symp. Inf. Th. , p. 165, Chicago, IL, June-July 2004.

[20] E. Ordentlich, G. Seroussi, S. Verdú, M.J. Weinberger, and T. Weissman, *A universal discrete image denoiser and its application to binary images*, in: Proc. IEEE International Conference on Image Processing, p. 117-120, Barcelona, Catalonia, Spain, September 2003.

[21] E. Ordentlich, M. Weinberger, and T. Weissman, *Efficient pruning of bi-directional context trees with applications to universal denoising and compression*, IEEE Inf. Th. workshop, San Antonio, Texas, October 24-29, 2004.

[22] E. Ordentlich, T. Weissman, M. Weinberger, A. Somekh-Baruch, and N. Merhav, *Discrete universal filtering through incremental parsing*, Data Compression Conference (DCC 2004), p. 352–361, Snowbird, Utah March 23-25, 2004.

[23] C. Peel, B. Hochwald, and L. Swindlehurst, *A Vector-Perturbation Technique for Near-Capacity Multi-Antenna Multi-User Communication*, 41st Annual Allerton Conf. on Communications, Control, and Computing, Monticello, IL, Oct. 2003.

[24] C. E. Shannon, *A mathematical theory of communication*, Bell Sys. Tech. Journal, 27(1948), pp. 379–423, 623–656.

[25] P. Soille, *Morphological Image Analysis: Principles and Applications*, Springer-Verlag, 1999.

[26] T. Weissman, E. Ordentlich, G. Seroussi, S. Verdú, and M. Weinberger, *Universal discrete denoising: Known channel*, IEEE Trans. Inform. Theory, 51:1(2005), pp. 5–28.

[27] R. Zhang and T. Weissman, *On discrete denoising for the burst-noise channel*, 42nd Annual Allerton Conf. on Communications, Control, and Computing, Monticello, IL, Sept. 29th – Oct. 1st, 2004.