

Approximate Message Passing Algorithm with Universal Denoising and Gaussian Mixture Learning

Yanting Ma, *Student Member, IEEE*, Junan Zhu, *Student Member, IEEE*,
and Dror Baron, *Senior Member, IEEE*

Abstract

We study compressed sensing (CS) signal reconstruction problems where an input signal is measured via matrix multiplication under additive white Gaussian noise. Our signals are assumed to be stationary and ergodic, but the input statistics are unknown; the goal is to provide reconstruction algorithms that are universal to the input statistics. We present a novel algorithmic framework that combines: (i) the approximate message passing (AMP) CS reconstruction framework, which solves the matrix channel recovery problem by iterative scalar channel denoising; (ii) a universal denoising scheme based on context quantization, which partitions the stationary ergodic signal denoising into independent and identically distributed (i.i.d.) subsequence denoising; and (iii) a density estimation approach that approximates the probability distribution of an i.i.d. sequence by fitting a Gaussian mixture (GM) model. In addition to the algorithmic framework, we provide three contributions: (i) numerical results showing that state evolution holds for non-separable Bayesian sliding-window denoisers; (ii) an i.i.d. denoiser based on a modified GM learning algorithm; and (iii) a universal denoiser that does not need information about the range where the input takes values from or require the input signal to be bounded. We provide two implementations of our universal CS recovery algorithm with one being faster and the other being more accurate. The two implementations compare favorably with existing universal reconstruction algorithms in terms of both reconstruction quality and runtime.

Index Terms

approximate message passing, compressed sensing, Gaussian mixture model, universal denoising.

I. INTRODUCTION

A. Motivation

Many scientific and engineering problems can be approximated as linear systems of the form

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{z}, \quad (1)$$

This work was supported in part by the National Science Foundation under the Grant CCF-1217749 and in part by the U.S. Army Research Office under the Grant W911NF-14-1-0314. Portions of the work appeared at the 52nd Annual Allerton Conference on Communication, Control, and Computing, Monticello, IL, Oct. 2014 [1].

Yanting Ma, Junan Zhu, and Dror Baron are with the Department of Electrical and Computer Engineering, NC State University, Raleigh, NC 27695. E-mail: {yma7, jzhu9, barondror}@ncsu.edu.

where $\mathbf{x} \in \mathbb{R}^N$ is the unknown input signal, $\mathbf{A} \in \mathbb{R}^{M \times N}$ is the matrix that characterizes the linear system, and $\mathbf{z} \in \mathbb{R}^M$ is measurement noise. The goal is to estimate \mathbf{x} from the measurements \mathbf{y} given \mathbf{A} and statistical information about \mathbf{z} . When $M \ll N$, the setup is known as compressed sensing (CS); by posing a sparsity or compressibility requirement on the signal, it is indeed possible to accurately recover \mathbf{x} from the ill-posed linear system [2, 3]. However, we might need $M > N$ when the signal is dense or the noise is substantial.

One popular scheme to solve the CS recovery problem is LASSO [4] (also known as basis pursuit denoising [5]): $\hat{\mathbf{x}} = \arg \min_{\mathbf{x} \in \mathbb{R}^N} \frac{1}{2} \|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2 + \gamma \|\mathbf{x}\|_1$, where $\|\cdot\|_p$ denotes the ℓ_p -norm, and γ reflects a trade-off between the sparsity $\|\mathbf{x}\|_1$ and residual $\|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2$. This approach does not require statistical information about \mathbf{x} and \mathbf{z} , and can be conveniently solved via standard convex optimization tools or the approximate message passing (AMP) algorithm [6]. However, the reconstruction quality is often far from optimal in terms of mean square error (MSE).

Bayesian CS recovery algorithms based on message passing [7–9] usually achieve better reconstruction quality, but must know the prior for \mathbf{x} . For parametric signals with unknown parameters, one can infer the parameters and achieve the minimum mean square error (MMSE) in some settings; examples include EM-GM-AMP-MOS [10], turboGAMP [11], and adaptive-GAMP [12].

Unfortunately, possible uncertainty about the input statistics may make it difficult to select a model class for empirical Bayes algorithms; a mismatched model can yield excess mean square error (EMSE) above the MMSE, and the EMSE can get amplified in linear inverse problems (1) compared to that in scalar estimation problems [13]. Our goal is to develop universal schemes that approach the optimal Bayesian performance for stationary ergodic signals despite not knowing the input statistics. Although others have worked on CS algorithms for independent and identically distributed (i.i.d.) signals with unknown distributions [10], we are particularly interested in developing algorithms for signals that may not be well approximated by i.i.d. models, because real-world signals often contain dependencies between different entries. For example, we will see in Fig. 6 that a chirp sound clip is reconstructed 1–2 dB better with models that can capture such dependencies than i.i.d. models applied to sparse transform coefficients.

While approaches based on Kolmogorov complexity [14–17] are theoretically appealing for universal signal recovery, they are not computable in practice [18, 19]. Several algorithms based on Markov chain Monte Carlo (MCMC) [20–23] leverage the fact that for stationary ergodic signals, both the per-symbol empirical entropy and Kolmogorov complexity converge asymptotically almost surely to the entropy rate of the signal [18], and aim to minimize the empirical entropy. The best existing implementation of the MCMC approach [23] often achieves an MSE that is within 3 dB of the MMSE, which resembles a result by Donoho for universal denoising [14].

In this paper, we confine our attention to the system model defined in (1), where the input signal \mathbf{x} is stationary and ergodic. We merge concepts from AMP [6], Gaussian mixture (GM) learning [24] for density estimation, and universal denoising for stationary ergodic signals [25, 26]. We call the resulting universal CS recovery algorithm AMP-UD (AMP with a universal denoiser). Two implementations of AMP-UD are provided, and they compare favorably with existing universal approaches in terms of reconstruction quality and runtime.

B. Related work and main results

Approximate message passing: AMP is an iterative algorithm that solves a linear inverse problem by successively

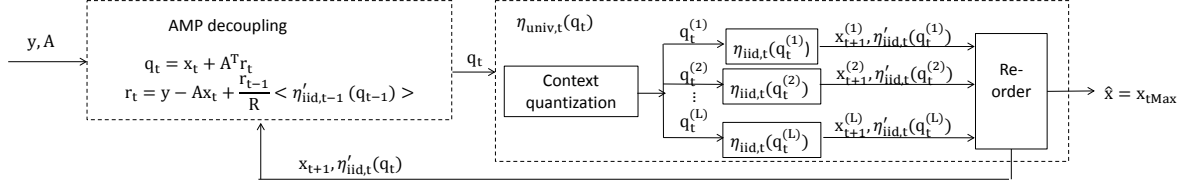


Fig. 1. Flow chart of AMP-UD. AMP (2, 3) decouples the linear inverse problem into scalar channel denoising problems. In the t -th iteration, the universal denoiser $\eta_{\text{univ},t}(\cdot)$ converts stationary ergodic signal denoising into i.i.d. subsequence denoising. Each i.i.d. denoiser $\eta_{\text{id},t}(\cdot)$ (13) outputs the denoised subsequence $\mathbf{x}_{t+1}^{(l)}$ and the derivative of the denoiser $\eta'_{\text{id},t}(\cdot)$ (16). The algorithm stops when the iteration index t reaches the predefined maximum t_{Max} , and outputs $\hat{\mathbf{x}}_{t_{\text{Max}}}$ as the CS recovery result.

converting matrix channel problems into scalar channel denoising problems with additive white Gaussian noise (AWGN). AMP has received considerable attention, because of its fast convergence and the state evolution (SE) formalism [6, 27], which offers a precise characterization of the AWGN denoising problem in each iteration. AMP with separable denoisers has been rigorously proved to obey SE [27].

The focus of this paper is the reconstruction of signals that are not necessarily i.i.d., and so we need to explore non-separable denoisers. Donoho *et al.* [28] provide numerical results demonstrating that SE accurately predicts the phase transition of AMP when some well-behaved non-separable minimax denoisers are applied, and conjecture that SE holds for AMP with a broader class of denoisers. A compressive imaging algorithm that applies non-separable image denoisers within AMP appears in Tan *et al.* [29]. Rush *et al.* [30] apply AMP to sparse superposition decoding, and prove that SE holds for AMP with certain block-separable denoisers and that such an AMP-based decoder achieves channel capacity. A potential challenge of implementing AMP is to obtain the Onsager correction term [6], which involves the calculation of the derivative of a denoiser. Metzler *et al.* [31] leverage a Monte Carlo technique to approximate the derivative of a denoiser when an explicit analytical formulation of the denoiser is unavailable, and provide numerical results showing that SE holds for AMP with their approximation.

Despite the encouraging results for using non-separable denoisers within AMP, a rigorous proof that SE holds for general non-separable denoisers has yet to appear. Consequently, new evidence showing that AMP obeys SE may increase the community's confidence about using non-separable denoisers within AMP. *Our first contribution is that we provide numerical results showing that SE holds for non-separable Bayesian sliding-window denoisers.*

Fitting Gaussian mixture models: Figueiredo and Jain [24] propose an unsupervised GM learning algorithm that fits a given data sequence with a GM model. The algorithm employs a cost function that resembles the minimum message length criterion, and the parameters are learned via expectation-maximization (EM).

Our GM fitting problem involves estimating the probability density function (pdf) of a sequence \mathbf{x} from its AWGN corrupted observations. We modify the GM fitting algorithm [24], so that a GM model can be learned from noisy data. Once the estimated pdf $\hat{p}_{\mathbf{X}}$ of \mathbf{x} is available, we estimate \mathbf{x} by computing the conditional expectation with the estimated pdf $\hat{p}_{\mathbf{X}}$ (recall that MMSE estimators rely on conditional expectation). *Our second contribution is that we modify the GM learning algorithm, and extend it to an i.i.d. denoiser.*

Universal denoising: Our denoiser for stationary ergodic signals is inspired by a context quantization approach [26], where a universal denoiser for a stationary ergodic signal involves multiple i.i.d. denoisers for conditionally i.i.d. subsequences. Sivaramakrishnan and Weissman [26] have shown that their universal denoiser based on context quantization can achieve the MMSE asymptotically for stationary ergodic signals with known bounds.

The boundedness condition of Sivaramakrishnan and Weissman [26] is partly due to their density estimation approach, in which the empirical distribution function is obtained by quantizing the bounded range of the signal. Such boundedness conditions may be undesirable in certain applications. We overcome this limitation by replacing their density estimation approach with GM model learning. *Our third contribution is a universal denoiser that does not need information about the bounds or require the input signal to be bounded; we conjecture that our universal denoiser achieves the MMSE asymptotically under some technical conditions.*

A flow chart of AMP-UD, which employs the AMP framework, along with our modified universal denoiser (η_{univ}) and the GM-based i.i.d. denoiser (η_{iid}), is shown in Fig. 1. Based on the numerical evidence that SE holds for AMP with Bayesian sliding-window denoisers and the conjecture that our universal denoiser can achieve the MMSE, we further conjecture that AMP-UD achieves the MMSE under some technical conditions. The details of AMP-UD, including two practical implementations, are developed in Sections II–V.

The remainder of the paper is arranged as follows. In Section II, we review AMP and provide new numerical evidence that AMP obeys SE with non-separable denoisers. Section III modifies the GM fitting algorithm, and extends it to an i.i.d. denoiser. In Section IV, we extend the universal denoiser based on context quantization to overcome the boundedness condition, and two implementations are provided to improve denoising quality. Our proposed AMP-UD algorithm is summarized in Section V. Numerical results are shown in Section VI, and we conclude the paper in Section VII.

II. APPROXIMATE MESSAGE PASSING WITH SLIDING-WINDOW DENOISERS

In this section, we apply non-separable Bayesian sliding-window denoisers within AMP, and provide numerical evidence that state evolution (SE) holds for AMP with this class of denoisers.

A. Review of AMP

Consider a linear system (1), where the measurement matrix \mathbf{A} has zero mean i.i.d. Gaussian entries with unit-norm columns on average, and \mathbf{z} represents i.i.d. Gaussian noise with pdf $p_{\mathbf{z}}(z_i) = \mathcal{N}(z_i; 0, \sigma_z^2)$, where z_i is the i -th entry of the vector \mathbf{z} , and $\mathcal{N}(x; \mu, \sigma^2)$ denotes a Gaussian pdf:

$$\mathcal{N}(x; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x - \mu)^2}{2\sigma^2}\right).$$

Note that AMP has been proved to follow SE when \mathbf{A} is a zero mean i.i.d. Gaussian matrix, but may diverge otherwise. Several techniques have been proposed to improve the convergence of AMP [32–35]. Moreover, other noise distributions can be supported using generalized AMP (GAMP) [9], and the noise distribution can be estimated in each GAMP iteration [12]. Such generalizations are beyond the scope of this work.

Starting with $\mathbf{x}_0 = \mathbf{0}$, the AMP algorithm [6] proceeds iteratively according to

$$\mathbf{x}_{t+1} = \eta_t(\mathbf{A}^T \mathbf{r}_t + \mathbf{x}_t), \quad (2)$$

$$\mathbf{r}_t = \mathbf{y} - \mathbf{A}\mathbf{x}_t + \frac{1}{R} \mathbf{r}_{t-1} \langle \eta'_{t-1}(\mathbf{A}^T \mathbf{r}_{t-1} + \mathbf{x}_{t-1}) \rangle, \quad (3)$$

where $R = M/N$ represents the measurement rate, t represents the iteration index, $\eta_t(\cdot)$ is a denoising function, and $\langle \mathbf{u} \rangle = \frac{1}{N} \sum_{i=1}^N u_i$ for some vector $\mathbf{u} \in \mathbb{R}^N$. The last term in (3) is called the Onsager correction term in statistical physics. The empirical distribution of \mathbf{x} is assumed to converge to some probability distribution p_X on \mathbb{R} , and the denoising function $\eta_t(\cdot)$ is separable in the original derivation of AMP [6, 27, 36]. That is, $\eta_t(\mathbf{u}) = (\eta_t(u_1), \eta_t(u_2), \dots, \eta_t(u_N))$ and $\eta'_t(\mathbf{u}) = (\eta'_t(u_1), \eta'_t(u_2), \dots, \eta'_t(u_N))$, where $\eta'_t(\cdot)$ denotes the derivative of $\eta_t(\cdot)$. A useful property of AMP is that at each iteration, the vector $\mathbf{A}^T \mathbf{r}_t + \mathbf{x}_t \in \mathbb{R}^N$ in (2) is statistically equivalent to the input signal \mathbf{x} corrupted by AWGN, where the noise variance σ_t^2 evolves following SE in the limit of large systems ($N \rightarrow \infty, M/N \rightarrow R$):

$$\sigma_{t+1}^2 = \sigma_z^2 + \frac{1}{R} \text{MSE}(\eta_t, \sigma_t^2), \quad (4)$$

where $\text{MSE}(\eta_t, \sigma_t^2) = \mathbb{E}_{X,W} \left[(\eta_t(X + \sigma_t W) - X)^2 \right]$, $W \sim \mathcal{N}(w; 0, 1)$, $X \sim p_X$, and $\sigma_0^2 = \sigma_z^2 + \frac{1}{R} \mathbb{E}[X^2]$. Formal statements for SE appear in the reference papers [27, 36]. Additionally, it is convenient to use the following estimator for σ_t^2 [27, 36]:

$$\hat{\sigma}_t^2 = \frac{1}{M} \|\mathbf{r}_t\|_2^2. \quad (5)$$

B. State evolution for Bayesian sliding-window denoisers

SE allows to calculate the asymptotic MSE of linear systems from the MSE of the denoiser used within AMP. Therefore, knowing that SE holds for AMP with the denoisers that we are interested in can help us choose a good denoiser for AMP. It has been conjectured by Donoho *et al.* [28] that AMP with a wide range of non-separable denoisers obeys SE. We now provide new evidence to support this conjecture by constructing non-separable Bayesian denoisers within a sliding-window denoising scheme for two stationary ergodic Markov signal models, and showing that SE accurately predicts the performance of AMP with this class of denoisers for large signal dimension N . Note that for a signal that is generated by a stationary ergodic process, its empirical distribution converges to the stationary distribution, hence the condition on the input signal in the proof for SE [27] is satisfied, and our goal is to numerically verify that SE holds for AMP with non-separable sliding-window denoisers for stationary ergodic signal models. Our rationale for examining the SE performance of sliding-window denoisers is that the context quantization based universal denoiser [26], which will be used in Section IV, resembles a sliding-window denoiser.

The mathematical model for an AWGN channel denoising problem is defined as

$$\mathbf{q} = \mathbf{x} + \mathbf{v}, \quad (6)$$

where $\mathbf{x} \in \mathbb{R}^N$ is the input signal, $\mathbf{v} \in \mathbb{R}^N$ is AWGN with pdf $p_V(v_i) = \mathcal{N}(v_i; 0, \sigma_v^2)$, and $\mathbf{q} \in \mathbb{R}^N$ is a sequence of noisy observations. Note that we are interested in designing denoisers for AMP, and the noise variance of the scalar channel in each AMP iteration can be estimated as $\hat{\sigma}_t^2$ (5). Therefore, throughout the paper we assume that the noise variance σ_v^2 is known when we discuss scalar channels.

In a separable denoiser, x_j is estimated only from its noisy observation q_j . The separable Bayesian denoiser that minimizes the MSE is point-wise conditional expectation,

$$\hat{x}_j = \mathbb{E}[X|Q = q_j] = \int xp(x|q_j)dx, \quad (7)$$

where Bayes' rule yields $p(x|q_j) = \frac{\mathcal{N}(q_j; x, \sigma_v^2)p_X(x)}{p_Q(q_j)}$. If entries of the input signal \mathbf{x} are drawn independently from p_X , then (7) achieves the MMSE.

When there are statistical dependencies among the entries of \mathbf{x} , a sliding-window scheme can be applied to improve the MSE. We consider two Markov sources as examples that contain statistical dependencies, and emphasize that our true motivation is the richer class of stationary ergodic sources.

Example source 1: Consider a two-state Markov state machine that contains states s_0 (zero state in which the signal entries are zero) and s_1 (nonzero state in which entries are nonzero). The transition probabilities are $p_{10} = p(s_0|s_1)$ and $p_{01} = p(s_1|s_0)$. In the steady state, the marginal probability of state s_1 is $\frac{p_{01}}{p_{01}+p_{10}}$. We call our first example source Markov-Gaussian (MGauss for short); it is generated by the two-state Markov machine with $p_{01} = \frac{3}{970}$ and $p_{10} = \frac{1}{10}$, and in the nonzero state the signal value follows a Gaussian distribution $\mathcal{N}(x; \mu_x, \sigma_x^2)$. These state transition parameters yield 3% nonzero entries in an MGauss signal on average.

Example source 2: Our second example is a four-state Markov switching signal (M4 for short) that follows the pattern $+1, +1, -1, -1, +1, +1, -1, -1, \dots$ with 3% error probability in state transitions, resulting in the signal switching from -1 to $+1$ or vice versa either too early or too late; the four states $s_1 = [-1 \ -1]$, $s_2 = [-1 \ +1]$, $s_3 = [+1 \ -1]$, and $s_4 = [+1 \ +1]$ have equal marginal probabilities 0.25 in the steady state.

Bayesian sliding-window denoiser: Let $\boldsymbol{\theta}$ be a binary vector, where $\theta_i = 0$ indicates $x_i = 0$, and $\theta_i = 1$ indicates $x_i \neq 0$. Denoting a block $(u_s, u_{s+1}, \dots, u_t)$ of any sequence \mathbf{u} by \mathbf{u}_s^t for $s < t$, the $(2k+1)$ -Bayesian sliding-window denoiser η_{MGauss} for the MGauss signal is defined as

$$\begin{aligned} \eta_{\text{MGauss},j}(\mathbf{q}_{j-k}^{j+k}) &= \mathbb{E}[X_j | \mathbf{Q}_{j-k}^{j+k} = \mathbf{q}_{j-k}^{j+k}] \\ &= \frac{\sum_{\substack{\boldsymbol{\theta}_{j-k}^{j+k} \in \{s_0, s_1\}^{2k+1} \\ \theta_j = s_1}} \left(\prod_{i=j-k}^{j+k} h(q_i, \theta_i; \mu_x, \sigma_x^2, \sigma_v^2) p_{\boldsymbol{\theta}_{j-k}^{j+k}}(\boldsymbol{\theta}_{j-k}^{j+k}) \right)}{p_{\mathbf{Q}_{j-k}^{j+k}}(\mathbf{q}_{j-k}^{j+k})} \cdot \left(\frac{\sigma_x^2}{\sigma_x^2 + \sigma_v^2} (q_j - \mu_x) + \mu_x \right), \quad (8) \end{aligned}$$

where

$$\begin{aligned} h(q_i, \theta_i; \mu_x, \sigma_x^2, \sigma_v^2) &= \begin{cases} \mathcal{N}(q_i; \mu_x, \sigma_v^2 + \sigma_x^2), & \text{if } \theta_i = s_1 \\ \mathcal{N}(q_i; 0, \sigma_v^2), & \text{if } \theta_i = s_0 \end{cases}, \\ p_{\boldsymbol{\theta}_{j-k}^{j+k}}(\boldsymbol{\theta}_{j-k}^{j+k}) &= p(\theta_{j-k}) \prod_{i=j-k}^{j+k-1} p(\theta_{i+1}|\theta_i), \\ p_{\mathbf{Q}_{j-k}^{j+k}}(\mathbf{q}_{j-k}^{j+k}) &= \sum p_{\boldsymbol{\theta}_{j-k}^{j+k}}(\boldsymbol{\theta}_{j-k}^{j+k}) \prod_{i=j-k}^{j+k} h(q_i, \theta_i; \mu_x, \sigma_x^2), \end{aligned}$$

and the summation is over $\boldsymbol{\theta}_{j-k}^{j+k} \in \{s_0, s_1\}^{2k+1}$.

The MSE of $\eta_{\text{MGauss},j}$ is

$$\begin{aligned} \text{MSE}(\eta_{\text{MGauss}}, \sigma_v^2) &= \mathbb{E} \left[\left(X_j - \eta_{\text{MGauss},j}(\mathbf{Q}_{j-k}^{j+k}) \right)^2 \right] \\ &= \frac{p_{01}(\sigma_x^2 + \mu_x^2)}{p_{01} + p_{10}} - \int_{\mathbb{R}^{2k+1}} \eta_{\text{MGauss},j}^2(\mathbf{q}) p_{\mathbf{Q}_{j-k}^{j+k}}(\mathbf{q}) d\mathbf{q}. \end{aligned} \quad (9)$$

Similarly, the $(2k+1)$ -Bayesian sliding-window denoiser η_{M4} for the M4 signal is defined as

$$\begin{aligned} \eta_{\text{M4},j}(\mathbf{q}_{j-k}^{j+k}) &= \mathbb{E}[X_j | \mathbf{Q}_{j-k}^{j+k} = \mathbf{q}_{j-k}^{j+k}] \\ &= \frac{p_{X_j, \mathbf{Q}_{j-k}^{j+k}}(1, \mathbf{q}_{j-k}^{j+k}) - p_{X_j, \mathbf{Q}_{j-k}^{j+k}}(-1, \mathbf{q}_{j-k}^{j+k})}{p_{X_j, \mathbf{Q}_{j-k}^{j+k}}(1, \mathbf{q}_{j-k}^{j+k}) + p_{X_j, \mathbf{Q}_{j-k}^{j+k}}(-1, \mathbf{q}_{j-k}^{j+k})}, \end{aligned} \quad (10)$$

where

$$\begin{aligned} p_{\mathbf{x}_{j-k}^{j+k}}(\mathbf{x}_{j-k}^{j+k}) &= p(x_{j-k}, x_{j-k+1}) \prod_{i=j-k}^{j+k-2} p(x_{i+2} | x_{i+1}, x_i), \\ p_{X_j, \mathbf{Q}_{j-k}^{j+k}}(x, \mathbf{q}_{j-k}^{j+k}) &= \sum p_{\mathbf{x}_{j-k}^{j+k}}(\mathbf{x}_{j-k}^{j+k}) \prod_{i=j-k}^{j+k} \mathcal{N}(q_i; x_i, \sigma_v^2), \end{aligned}$$

where the summation is over $\mathbf{x}_{j-k}^{j+k} \in \{-1, 1\}^{2k+1}$ with $x_j = x \in \{-1, 1\}$ fixed.

It can be shown that

$$\begin{aligned} \text{MSE}(\eta_{\text{M4}}, \sigma_v^2) &= \mathbb{E} \left[\left(X_j - \eta_{\text{M4},j}(\mathbf{Q}_{j-k}^{j+k}) \right)^2 \right] \\ &= 4 \int_{\mathbb{R}^{2k+1}} \frac{p_{X_j, \mathbf{Q}_{j-k}^{j+k}}(-1, \mathbf{q}) p_{X_j, \mathbf{Q}_{j-k}^{j+k}}(1, \mathbf{q})}{p_{\mathbf{Q}_{j-k}^{j+k}}(\mathbf{q})} d\mathbf{q}. \end{aligned} \quad (11)$$

If AMP with η_{MGauss} or η_{M4} obeys SE, then the noise variance σ_t^2 should evolve according to (4). As a consequence, the reconstruction error at iteration t can be predicted by evaluating (9) or (11) with σ_v^2 being replaced by σ_t^2 .

Numerical evidence: We apply η_{MGauss} (8) within AMP for MGauss signals, and η_{M4} (10) within AMP for M4 signals. The window size $2k+1$ is chosen to be 1 or 3 for η_{MGauss} , and 1 or 5 for η_{M4} . Note that when the window size is 1, η_{MGauss} and η_{M4} become separable denoisers. The MSE predicted by SE is compared to the empirical MSE at each iteration where the input signal to noise ratio ($\text{SNR} = 10 \log_{10}[(N\mathbb{E}[X^2])/(M\sigma_z^2)]$) is 10 dB for both MGauss and M4. It is shown in Fig. 2 for AMP with η_{MGauss} and η_{M4} that the markers representing the empirical MSE track the lines predicted by SE, and that side-information from neighboring entries helps improve the MSE.

Our SE results for the two Markov sources increase our confidence that AMP with non-separable denoisers that incorporate information from neighboring entries will track SE.

The reader may have noticed from Fig. 1 that the universal denoiser $\eta_{\text{univ}}(\cdot)$ is acting as a set of separable denoisers $\eta_{\text{hid}}(\cdot)$. However, the statistical information used by $\eta_{\text{hid}}(\cdot)$ is learned from subsequences $\mathbf{q}_t^{(1)}, \dots, \mathbf{q}_t^{(L)}$ of the noisy sequence \mathbf{q}_t , and the subsequencing result is determined by the neighborhood of each entry. The SE results for the Bayesian sliding-window denoisers motivate us to apply the universal denoiser within AMP for CS reconstruction of stationary ergodic signals with unknown input statistics. Indeed, the numerical results in Section VI show that AMP with a universal denoiser leads to a promising universal CS recovery algorithm.

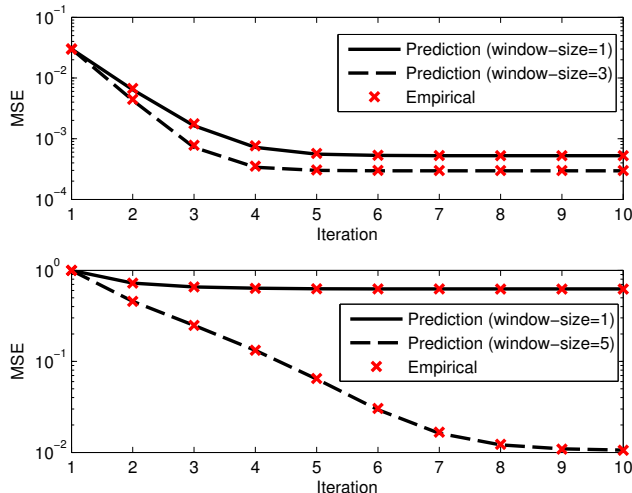


Fig. 2. Top: Numerical verification of SE for AMP with η_{MGauss} (8) when the input is an MGauss signal. ($N = 20000$, $R = M/N = 0.4$, SNR = 10 dB.) Bottom: Numerical verification of SE for AMP with η_{M4} (10) when the input is an M4 signal. ($N = 20000$, $R = 0.4$, SNR = 10 dB.)

III. I.I.D. DENOISING VIA GAUSSIAN MIXTURE FITTING

We will see in Section IV that context quantization maps the non-i.i.d. sequence \mathbf{q} into conditionally independent subsequences, and now focus our attention on denoising the resulting i.i.d. subsequences.

A. Background

The pdf of a Gaussian mixture (GM) has the form:

$$p(x) = \sum_{s=1}^S \alpha_s \mathcal{N}(x; \mu_s, \sigma_s^2), \quad (12)$$

where S is the number of Gaussian components, and $\sum_{s=1}^S \alpha_s = 1$, so that $p(x)$ is a proper pdf.

Figueiredo and Jain [24] propose to fit a GM model for a given data sequence by starting with some arbitrarily large S , and inferring the structure of the mixture by letting the mixing probabilities α_s of some components be zero. This leads to an unsupervised learning algorithm that automatically determines the number of Gaussian components from data. This approach resembles the concept underlying the minimum message length (MML) criterion that selects the best overall model from the entire model space, which differs from model class selection based on the best model within each class.¹ This criterion can be interpreted as posing a Dirichlet prior on the mixing probability and perform maximum a posteriori estimation [24]. A component-wise EM algorithm that updates $\{\alpha_s, \mu_s, \sigma_s^2\}$ sequentially in s is used to implement the MML-based approach. The main feature of the component-wise EM algorithm is that if α_s is estimated as 0, then the s -th component is immediately removed, and the expectation is recalculated before moving to the estimation of the next component.

¹All models with the same number of components belong to one model class, and different models within a model class have different parameters for each component.

B. Extension to denoising

Consider the scalar channel denoising problem defined in (6) with an i.i.d. input. We propose to estimate \mathbf{x} from its Gaussian noise corrupted observations \mathbf{q} by posing a GM prior on \mathbf{x} , and learning the parameters of the GM model with a modified version of the algorithm by Figueiredo and Jain [24].

Initialization of EM: The EM algorithm must be initialized for each parameter, $\{\alpha_s, \mu_s, \sigma_s^2\}$, $s = 1, \dots, S$. One may choose to initialize the Gaussian components with equal mixing probabilities and equal variances, and the initial value of the means are randomly sampled from the input data sequence [24], which in our case is the sequence of noisy observations \mathbf{q} . However, in CS recovery problems, the input signal is often sparse, and it becomes difficult to correct the initial value if the initialized values are far from the truth. To see why a poor initialization might be problematic, consider the following scenario: a sparse binary signal that contains a few ones and is corrupted by Gaussian noise is sent to the algorithm. If the initialization levels of the μ_s 's are all around zero, then the algorithm is likely to fit a Gaussian component with near-zero mean and large variance rather than two narrow Gaussian components, one of which has mean close to zero while the other has mean close to one.

To address this issue, we modify the initialization to examine the maximal distance between each symbol of the input data sequence and the current initialization of the μ_s 's. If the distance is greater than $0.1\sigma_q$, then we add a Gaussian component whose mean is initialized as the value of the symbol being examined, where σ_q^2 is the estimated variance of the noisy observations \mathbf{q} . We found in our simulations that the modified initialization improves the accuracy of the density estimation, and speeds up the convergence of the EM algorithm; the details of the simulation are omitted for brevity.

Parameter estimation from noisy data: Two possible modifications can be made to the original GM learning algorithm [24] that is designed for clean data.

We first notice that the model for the noisy data is a GM convolved with Gaussian noise, which is a new GM with larger component variances. Hence, one approach is to use the original algorithm [24] to fit a GM to the noisy data, but to remove a component immediately during the EM iterations if the estimated component variance is much smaller than the noise variance σ_v^2 . Specifically, during the parameter learning process, if a component has variance that is less than $0.2\sigma_v^2$, we assume that this low-variance component is spurious, and remove it from the mixture model. However, if the component variance is between $0.2\sigma_v^2$ and $0.9\sigma_v^2$, then we force the component variance to be $0.9\sigma_v^2$ and let the algorithm keep tracking this component. For component variance greater than $0.9\sigma_v^2$, we do not adjust the algorithm. The parameters 0.2 and 0.9 are chosen, because they provide reasonable MSE performance for a wide range of signals that we tested. These parameters are then fixed for our algorithm to generate the numerical results in Section VI. At the end of the parameter learning process, all remaining components with variances less than σ_v^2 are set to have variances equal to σ_v^2 . That said, when subtracting the noise variance σ_v^2 from the Gaussian components of \hat{p}_Q to obtain the components of \hat{p}_X , we could have components with zero-valued variance, which yields deltas in \hat{p}_X . Note that deltas are in general difficult to fit with a limited amount of observations, and our modification helps the algorithm estimate deltas.

Another approach is to introduce latent variables that represent the underlying clean data, and estimate the

parameters of the GM for the latent variables directly. Hence, similar to the original algorithm, a component is removed only when the estimated mixing probability is non-positive. It can be shown that the GM parameters are estimated as

$$\begin{aligned}\hat{\alpha}_s(t+1) &= \frac{\max\left\{\sum_{i=1}^N w_i^{(s)}(t) - 1, 0\right\}}{\sum_{s:\alpha_s>0} \max\left\{\sum_{i=1}^N w_i^{(s)}(t) - 1, 0\right\}}, \\ \hat{\mu}_s(t+1) &= \frac{\sum_{i=1}^N w_i^{(s)}(t) a_i^{(s)}(t)}{\sum_{i=1}^N w_i^{(s)}(t)}, \\ \hat{\sigma}_s^2(t+1) &= \frac{\sum_{i=1}^N w_i^{(s)}(t) \left(v_i^{(s)}(t) + \left(a_i^{(s)}(t) - \hat{\mu}_s(t+1)\right)^2\right)}{\sum_{i=1}^N w_i^{(s)}(t)},\end{aligned}$$

where

$$\begin{aligned}w_i^{(s)}(t) &= \frac{\hat{\alpha}_s(t) \mathcal{N}(q_i; \hat{\mu}_s(t), \sigma_v^2 + \hat{\sigma}_s(t)^2)}{\sum_{m=1}^S \hat{\alpha}_m(t) \mathcal{N}(q_i; \hat{\mu}_m(t), \sigma_v^2 + \hat{\sigma}_m(t)^2)}, \\ a_i^{(s)}(t) &= \frac{\hat{\sigma}_s^2(t)}{\hat{\sigma}_s^2(t) + \sigma_v^2} (q_i - \hat{\mu}_s(t)) + \hat{\mu}_s(t), \\ v_i^{(s)}(t) &= \frac{\sigma_v^2 \sigma_s^2}{\sigma_v^2 + \hat{\sigma}_s^2(t)}.\end{aligned}$$

Detailed derivations appear in the Appendix.

We found in our simulation that the first approach converges faster and leads to lower reconstruction error, especially for discrete-valued inputs. Therefore, the simulation results presented in Section VI use the first approach.

Denoising: Once the parameters in (12) are estimated, we define a denoiser for i.i.d. signals as conditional expectation:

$$\begin{aligned}\eta_{\text{iid}}(q) &= \mathbb{E}[X|Q = q] \\ &= \sum_{s=1}^S \mathbb{E}[X|Q = q, \text{comp} = s] P(\text{comp} = s|Q = q) \\ &= \sum_{s=1}^S \left(\frac{\sigma_s^2}{\sigma_s^2 + \sigma_v^2} (q - \mu_s) + \mu_s \right) \frac{\alpha_s \mathcal{N}(q; \mu_s, \sigma_s^2 + \sigma_v^2)}{\sum_{s=1}^S \alpha_s \mathcal{N}(q; \mu_s, \sigma_s^2 + \sigma_v^2)},\end{aligned}\tag{13}$$

where comp is the component index, and

$$\mathbb{E}[X|Q = q, \text{comp} = s] = \left(\frac{\sigma_s^2}{\sigma_s^2 + \sigma_v^2} (q - \mu_s) + \mu_s \right)$$

is the Wiener filter for component s .

We have verified numerically for several distributions and low to moderate noise levels that the denoising results obtained by the GM-based i.i.d. denoiser (13) approach the MMSE within a few hundredths of a dB. For example, the favorable reconstruction results for i.i.d. sparse Laplace signals in Fig. 3 show that the GM-based denoiser approaches the MMSE.

IV. UNIVERSAL DENOISING

We have seen in Section III that an i.i.d. denoiser based on GM learning can denoise i.i.d. signals with unknown distributions. Our goal in this work is to reconstruct stationary ergodic signals that are not necessarily i.i.d. Sivaramakrishnan and Weissman [26] have proposed a universal denoising scheme for stationary ergodic signals with known bounds based on context quantization, where a stationary ergodic signal is partitioned into i.i.d. subsequences. In this section, we modify the context quantization scheme and apply the GM-based denoiser (13) to the i.i.d. subsequences, so that our universal denoiser can denoise stationary ergodic signals that are unbounded or with unknown bounds.

A. Background

Consider the denoising problem (6), where the input \mathbf{x} is stationary ergodic. The main idea of the context quantization scheme [26] is to quantize the noisy symbols \mathbf{q} to generate quantized contexts that are used to partition the unquantized symbols into subsequences. That is, given the noisy observations $\mathbf{q} \in \mathbb{R}^N$, define the context of q_j as $\mathbf{c}_j = [\mathbf{q}_{j-k}^{j-1}; \mathbf{q}_{j+1}^{j+k}] \in \mathbb{R}^{2k}$ for $j = 1+k, \dots, N-k$, where $[\mathbf{a}; \mathbf{b}]$ denotes the concatenation of the sequences \mathbf{a} and \mathbf{b} . For $j \leq k$ or $j \geq N-k+1$, the median value q_{med} of \mathbf{q} is used as the missing symbols in the contexts. As an example for $j = k$, we only have $k-1$ symbols in \mathbf{q} before q_k , and so the first symbol in \mathbf{c}_k is missing; we define $\mathbf{c}_k = [q_{\text{med}}; \mathbf{q}_1^{k-1}; \mathbf{q}_{k+1}^{2k}]$. Vector quantization can then be applied to the context set $\mathcal{C} = \{\mathbf{c}_j : j = 1, \dots, N\}$, and each \mathbf{c}_j is assigned a label $l_j \in \{1, \dots, L\}$ that represents the cluster that \mathbf{c}_j belongs to. Finally, the L subsequences that consist of symbols from \mathbf{q} with the same label are obtained by taking $\mathbf{q}^{(l)} = \{q_j : l_j = l\}$, for $l = 1, \dots, L$.

The symbols in each subsequence $\mathbf{q}^{(l)}$ are regarded as approximately conditionally identically distributed given the common quantized contexts. The rationale underlying this concept is that a sliding-window denoiser uses information from the contexts to estimate the current symbol, and symbols with similar contexts in the noisy output of the scalar channel have similar contexts in the original signal. Therefore, symbols with similar contexts can be grouped together and denoised using the same denoiser. Note that Sivaramakrishnan and Weissman [26] propose a second subsequencing step, which further partitions each subsequence into smaller subsequences such that a symbol in a subsequence does not belong to the contexts of any other symbols in this subsequence. This step ensures that the symbols within each subsequence are mutually independent, which is crucial for theoretical analysis. However, for finite-length signals, small subsequences may occur, and they may not contain enough symbols to learn its empirical pdf well. Therefore, we omit this second subsequencing step in our implementations.

In order to estimate the distribution of $\mathbf{x}^{(l)}$, which is the clean subsequence corresponding to $\mathbf{q}^{(l)}$, Sivaramakrishnan and Weissman [26] first estimate the pdf $\hat{p}_Q^{(l)}$ of $\mathbf{q}^{(l)}$ via kernel density estimation. They then quantize the range that x_i 's take values from and the levels of the empirical distribution function of \mathbf{x} , and find a quantized distribution function that matches $\hat{p}_Q^{(l)}$ well. Once the distribution function of $\mathbf{x}^{(l)}$ is obtained, the conditional expectation of the symbols in the l -th subsequence can be calculated.

For error metrics that satisfy some mild technical conditions, Sivaramakrishnan and Weissman [26] have proved for stationary ergodic signals with bounded components that their universal denoiser asymptotically achieves the

optimal estimation error among all sliding-window denoising schemes despite not knowing the prior for the signal. When the error metric is square error, the optimal error is the MMSE.

B. Extension to unbounded signals and signals with unknown bounds

Sivaramakrishnan and Weissman [26] have shown that one can denoise a stationary ergodic signal by (i) grouping together symbols with similar contexts and (ii) applying an i.i.d. denoiser to each group. Such a scheme is optimal in the limit of large signal dimension N . However, their denoiser assumes an input with known bounds, which might make it inapplicable to some real-world settings. In order to be able to estimate signals that take values from the entire real line, in step (ii), we apply the GM learning algorithm for density estimation, which has been discussed in detail in Section III, and compute the conditional expectation with the estimated density as our i.i.d. denoiser.

We now provide details about a modification made to step (i). The context set \mathcal{C} is acquired in the same way as described in Section IV-A. Because the symbols in the context $\mathbf{c}_j \in \mathcal{C}$ that are closer in index to q_j are likely to provide more information about x_j than the ones that are located further away, we add weights to the contexts before clustering. That is, for each $\mathbf{c}_j \in \mathcal{C}$ of length $2k$, the weighted context is defined as

$$\mathbf{c}'_j = \mathbf{c}_j \odot \mathbf{w},$$

where \odot denotes a point-wise product, and the weights take values,

$$w_{k_i} = \begin{cases} e^{-\beta(k-k_i)}, & k_i = 1, \dots, k \\ e^{-\beta(k_i-k-1)}, & k_i = k+1, \dots, 2k \end{cases}, \quad (14)$$

for some $\beta \geq 0$. While in noisier channels, it might be necessary to use information from longer contexts, comparatively short contexts could be sufficient for cleaner channels. Therefore, the exponential decay rate β is made adaptive to the noise level in a way such that β increases with SNR. Specifically, β is chosen to be linear in SNR:

$$\beta = b_1 \log_{10}((\|\mathbf{q}\|_2^2/N - \sigma_v^2)/\sigma_v^2) + b_2, \quad (15)$$

where $b_1 > 0$ and b_2 can be determined numerically. Specifically, we run the algorithm with a sufficiently large range of β values for various input signals at various SNR levels and measurement rates. For each setting, we select the β that achieves the best reconstruction quality. Then, b_1 and b_2 are obtained using the least squares approach. Note that b_1 and b_2 are fixed for all the simulation results presented in Section VI. If the parameters were tuned for each individual input signal, then the optimal parameter values might vary for different input signals, and the reconstruction quality might be improved. The simulation results in Section VI with fixed parameters show that when the parameters are slightly off from the individually optimal ones, the reconstruction quality of AMP-UD is still comparable or better than the prior art. We choose the linear relation because it is simple and fits well with our empirical optimal values for β ; other choices for β might be possible. The weighted context set $\mathcal{C}' = \{\mathbf{c}'_j : j = 1, \dots, N\}$ is then sent to a k -means algorithm [37], and $\mathbf{q}^{(l)}, l = 1, \dots, L$, are obtained according to the labels determined via clustering. We can now apply the GM-based i.i.d. denoiser (13) to each subsequence

separately. However, one potential problem is that the GM fitting algorithm might not provide a good estimate of the model when the number of data points is small. We propose two approaches to address this small cluster issue.

Approach 1: Borrow members from nearby clusters. A post-processing step can be added to ensure that the pdf of $\mathbf{q}^{(l)}$ is estimated from no less than T symbols. That is, if the size of $\mathbf{q}^{(l)}$, which is denoted by B , is less than T , then $T - B$ symbols in other clusters whose contexts are closest to the centroid of the current cluster are included to estimate the empirical pdf of $\mathbf{q}^{(l)}$, while after the pdf is estimated, the extra symbols are removed, and only $\mathbf{q}^{(l)}$ is denoised with the currently estimated pdf. We call UD with Approach 1 “UD1.”²

Approach 2: Merge statistically similar subsequences. An alternative approach is to merge subsequences iteratively according to their statistical characterizations. The idea is to find subsequences with pdfs that are close in Kullback-Leibler (KL) distance [18], and decide whether merging them can yield a better model according to the minimum description length (MDL) [38] criterion. Denote the iteration index for the merging process by h . After the k -means algorithm, we have obtained a set of subsequences $\{\mathbf{q}_h^{(l)} : l = 1, \dots, L_h\}$, where L_h is the current number of subsequences. A GM pdf $\hat{p}_{Q,h}^{(l)}$ is learned for each subsequence $\mathbf{q}_h^{(l)}$. The MDL cost c_h^{MDL} for the current model is calculated as:

$$c_h^{\text{MDL}} = - \sum_{l=1}^{L_h} \sum_{i=1}^{|\mathbf{q}_h^{(l)}|} \log \left(\hat{p}_{Q,h}^{(l)}(q_{i,h}^{(l)}) \right) + \sum_{l=1}^{L_h} \frac{3 \cdot m_h^{(l)}}{2} \log \left(|\mathbf{q}_h^{(l)}| \right) + 2 \cdot L_h + L_0 \cdot \sum_{l=1}^{L_h} \frac{n_h^{(l)}}{L_0} \log \left(\frac{L_0}{n_h^{(l)}} \right),$$

where $q_{i,h}^{(l)}$ is the i -th entry of the subsequence $\mathbf{q}_h^{(l)}$, $m_h^{(l)}$ is the number of Gaussian components in the mixture model for subsequence $\mathbf{q}_h^{(l)}$, L_0 is the number of subsequences before the merging procedure, and $n_h^{(l)}$ is the number of subsequences in the initial set $\{\mathbf{q}_0^{(l)} : l = 1, \dots, L_0\}$ that are merged to form the subsequence $\mathbf{q}_h^{(l)}$. The four terms in c_h^{MDL} are interpreted as follows. The first term is the negative log likelihood of the entire noisy sequence \mathbf{q}_h given the current GM models. The second term is the penalty for the number of parameters used to describe the model, where we have 3 parameters (α, μ, σ^2) for each Gaussian component, and $m_h^{(l)}$ components for the subsequence $\mathbf{q}_h^{(l)}$. The third term arises from 2 bits that are used to encode $m_h^{(l)}$ for $l = 1, \dots, L_h$, because our numerical results have shown that the number of Gaussian components rarely exceeds 4. In the fourth term, $\sum_{l=1}^{L_h} \frac{n_h^{(l)}}{L_0} \log \left(\frac{L_0}{n_h^{(l)}} \right)$ is the uncertainty that a subsequence from the initial set is mapped to $\mathbf{q}_h^{(l)}$ with probability $n_h^{(l)}/L_0$, for $l = 1, \dots, L_h$. Therefore, the fourth term is the coding length for mapping the L_0 subsequences from the initial set to the current set.

We then compute the KL distance between the pdf of $\mathbf{q}_h^{(s)}$ and that of $\mathbf{q}_h^{(t)}$, for $s, t = 1, \dots, L_h$:

$$D \left(\hat{p}_{Q,h}^{(s)} \parallel \hat{p}_{Q,h}^{(t)} \right) = \int \hat{p}_{Q,h}^{(s)}(q) \log \left(\frac{\hat{p}_{Q,h}^{(s)}(q)}{\hat{p}_{Q,h}^{(t)}(q)} \right) dq.$$

A symmetric $L_h \times L_h$ distance matrix \mathbf{D}_h is obtained by letting its s -th row and t -th column be

$$D \left(\hat{p}_{Q,h}^{(s)} \parallel \hat{p}_{Q,h}^{(t)} \right) + D \left(\hat{p}_{Q,h}^{(t)} \parallel \hat{p}_{Q,h}^{(s)} \right).$$

²A related approach is k -nearest neighbors, where for each symbol in \mathbf{q} , we find T symbols whose contexts are nearest to that of the current symbol and estimate its pdf from the T symbols. The k -nearest neighbors approach requires to run the GM learning algorithm [24] N times in each AMP iteration, which significantly slows down the algorithm.

Suppose the smallest entry in the upper triangular part of \mathbf{D}_h (not including the diagonal) is located in the s^* -th row and t^* -th column, then $\mathbf{q}_h^{(s^*)}$ and $\mathbf{q}_h^{(t^*)}$ are temporarily merged to form a new subsequence, and a new GM pdf is learned for the merged subsequence. We now have a new model with $L_{h+1} = L_h - 1$ GM pdfs, and the MDL criterion c_{h+1}^{MDL} is calculated for the new model. If c_{h+1}^{MDL} is smaller than c_h^{MDL} , then we accept the new model, and calculate a new $L_{h+1} \times L_{h+1}$ distance matrix \mathbf{D}_{h+1} ; otherwise we keep the current model, and look for the next smallest entry in the upper triangular part of the current $L_h \times L_h$ distance matrix. The number of subsequences is decreased by at most one after each iteration, and the merging process ends when there is only one subsequence left, or the smallest KL distance between two GM pdfs is greater than some threshold, which is determined numerically. We call UD with Approach 2 “UD2.”

We will see in Section VI that UD2 is more reliable than UD1 in terms of MSE performance, whereas UD1 is faster than UD2. This is because UD2 applies a more complicated (and thus slower) subsequencing procedure, which allows more accurate GM models to be fitted to subsequences.

V. PROPOSED UNIVERSAL CS RECOVERY ALGORITHM

Combining the three components that have been discussed in Sections II–IV, we are now ready to introduce our proposed universal CS recovery algorithm AMP-UD. Note that the AMP-UD algorithm is designed for medium to large size problems. Specifically, the problem size should be large enough, such that the decoupling effect of AMP, which converts the compressed sensing problem to a series of scalar channel denoising problems, approximately holds, and that the statistical information about the input can be approximately estimated by the universal denoiser.

Consider a linear system (1), where the input signal \mathbf{x} is stationary and ergodic with unknown distributions, and the matrix \mathbf{A} has i.i.d. Gaussian entries. To estimate \mathbf{x} from \mathbf{y} given \mathbf{A} , we apply AMP as defined in (2) and (3). In each iteration, AWGN corrupted observations, $\mathbf{q}_t = \mathbf{x}_t + \mathbf{A}^T \mathbf{r}_t = \mathbf{x} + \mathbf{v}$, are obtained, where σ_v^2 is estimated by $\hat{\sigma}_t^2$ (5). A subsequencing approach is applied to generate i.i.d. subsequences, where Approach 1 and Approach 2 (Section IV-B) are two possible implementations. The GM-based i.i.d. denoiser (13) is then utilized to denoise each i.i.d. subsequence.

To obtain the Onsager correction term in (3), we need to calculate the derivative of η_{iid} (13). For $q \in \mathbb{R}$, denoting

$$\begin{aligned} f(q) &= \sum_{s=1}^S \alpha_s \mathcal{N}(q; \mu_s, \sigma_s^2 + \sigma_v^2) \left(\frac{\sigma_s^2}{\sigma_s^2 + \sigma_v^2} (q - \mu_s) + \mu_s \right), \\ g(q) &= \sum_{s=1}^S \alpha_s \mathcal{N}(q; \mu_s, \sigma_s^2 + \sigma_v^2), \end{aligned}$$

we have that

$$\begin{aligned} f'(q) &= \sum_{s=1}^S \alpha_s \mathcal{N}(q; \mu_s, \sigma_s^2 + \sigma_v^2) \cdot \left(\frac{\sigma_s^2 + \mu_s^2 - q\mu_s}{\sigma_s^2 + \sigma_v^2} - \left(\frac{\sigma_s(q - \mu_s)}{\sigma_s^2 + \sigma_v^2} \right)^2 \right), \\ g'(q) &= \sum_{s=1}^S \alpha_s \mathcal{N}(q; \mu_s, \sigma_s^2 + \sigma_v^2) \left(-\frac{q - \mu_s}{\sigma_s^2 + \sigma_v^2} \right). \end{aligned}$$

Therefore,

$$\eta'_{\text{iid}}(q) = \frac{f'(q)g(q) - f(q)g'(q)}{(g(q))^2}. \quad (16)$$

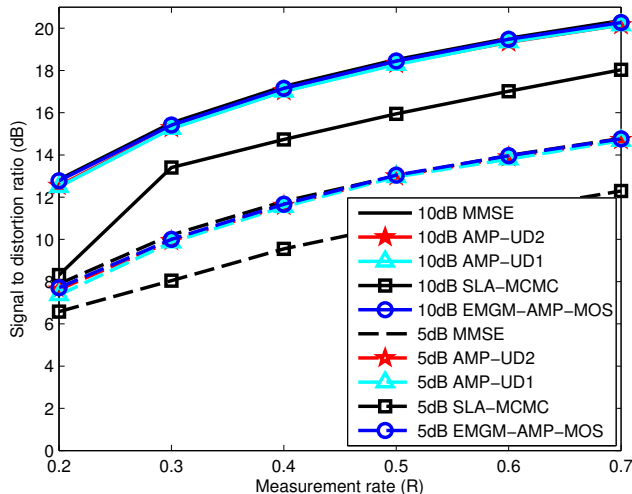


Fig. 3. Two AMP-UD implementations, SLA-MCMC, and EM-GM-AMP-MOS reconstruction results for an i.i.d. sparse Laplace signal as a function of measurement rate ($R = M/N$). Note that the SDR curves for the two AMP-UD implementations and EM-GM-AMP-MOS overlap the MMSE curve. ($N = 10000$, SNR = 5 dB or 10 dB.)

We highlight that AMP-UD is unaware of the input SNR and also unaware of the input statistics. The noise variance σ_v^2 in the scalar channel denoising problem is estimated by the average energy of the residual (5). The input's statistical structure is learned by the universal denoiser without any prior assumptions.

It has been proved [26] that the context quantization universal denoising scheme can asymptotically achieve the MMSE for stationary ergodic signals with known bounds. We have extended the scheme to unbounded signals in Sections III and IV, and conjecture that our modified universal denoiser can asymptotically achieve the MMSE for unbounded stationary ergodic signals. AMP with MMSE-achieving separable denoisers has been proved to asymptotically achieve the MMSE in linear systems for i.i.d. inputs [27]. In Section II-B, we have provided numerical evidence that shows that SE holds for AMP with Bayesian sliding-window denoisers. Bayesian sliding-window denoisers with proper window-sizes are MMSE-achieving non-separable denoisers [26]. Given that our universal denoiser resembles a Bayesian sliding-window denoiser, we conjecture that AMP-UD can achieve the MMSE for stationary ergodic inputs in the limit of large linear systems where the matrix has i.i.d. random entries. Note that we have optimized the window-size for inputs of length $N = 10000$ via numerical experiments. We believe that the window size should increase with N , and leave the characterization of the optimal window size for future work.

VI. NUMERICAL RESULTS

We run AMP-UD1 (AMP with UD1) and AMP-UD2 (AMP with UD2) in MATLAB on a Dell OPTIPLEX 9010 running an Intel(R) Core™ i7-3770 with 16GB RAM, and test them utilizing different types of signals, including synthetic signals, a chirp sound clip, and a speech signal, at various measurement rates and SNR levels, where we remind the reader that SNR is defined in Section II-B. The input signal length N is 10000 for synthetic signals and roughly 10000 for the chirp sound clip and the speech signal. The context size $2k$ is chosen to be 12, and

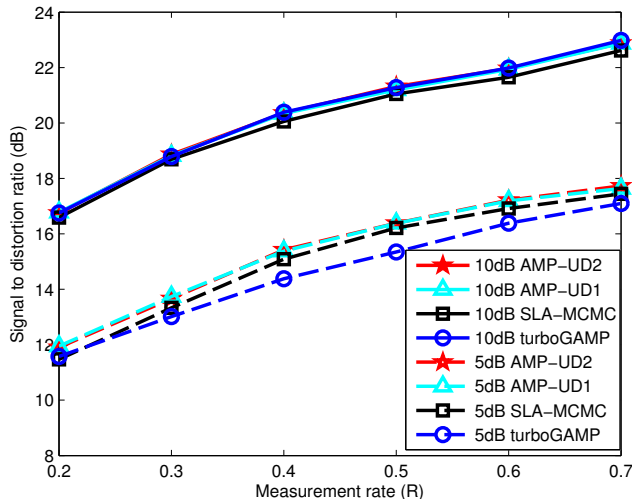


Fig. 4. Two AMP-UD implementations, SLA-MCMC, and turboGAMP reconstruction results for a two-state Markov signal with nonzero entries drawn from a uniform distribution $U[0, 1]$ as a function of measurement rate. Note that the SDR curves for the two AMP-UD implementations overlap at SNR = 5 dB, and they both overlap turboGAMP at SNR = 10 dB. ($N = 10000$, SNR = 5 dB or 10 dB.)

the contexts are weighted according to (14) and (15). The context quantization is implemented via the k -means algorithm [37]. In order to avoid possible divergence of AMP-UD, possibly due to a bad GM fitting, we employ a damping technique [32] to slow down the evolution. Specifically, damping is an extra step in the AMP iteration (3); instead of updating the value of \mathbf{x}_{t+1} by the output of the denoiser $\eta_t(\mathbf{A}^T \mathbf{r}_t + \mathbf{x}_t)$, a weighted sum of $\eta_t(\mathbf{A}^T \mathbf{r}_t + \mathbf{x}_t)$ and \mathbf{x}_t is taken as follows,

$$\mathbf{x}_{t+1} = \lambda \eta_t(\mathbf{A}^T \mathbf{r}_t + \mathbf{x}_t) + (1 - \lambda) \mathbf{x}_t,$$

for some $\lambda \in (0, 1]$.

Parameters for AMP-UD1: The number of clusters L is initialized as 10, and may become smaller if empty clusters occur. The lower bound T on the number of symbols required to learn the GM parameters is 256. The damping parameter λ is 0.1, and we run 100 AMP iterations.

Parameters for AMP-UD2: The initial number of clusters is set to be 30, and these clusters will be merged according to the scheme described in Section IV. Because each time when merging occurs, we need to apply the GM fitting algorithm one more time to learn a new mixture model for the merged cluster, which is computationally demanding, we apply adaptive damping [34] to reduce the number of iterations required; the number of AMP iterations is set to be 30. The damping parameter is initialized to be 0.5, and will increase (decrease) within the range $[0.01, 0.5]$ if the value of the scalar channel noise estimator $\hat{\sigma}_t^2$ (5) decreases (increases).

The recovery performance is evaluated by signal to distortion ratio ($\text{SDR} = 10 \log_{10}(\mathbb{E}[X^2]/\text{MSE})$), where the MSE is averaged over 50 random draws of \mathbf{x} , \mathbf{A} , and \mathbf{z} .

We compare the performance of the two AMP-UD implementations to (i) the universal CS recovery algorithm SLA-MCMC [23]; and (ii) the empirical Bayesian message passing approaches EM-GM-AMP-MOS [10] for i.i.d. inputs and turboGAMP [11] for non-i.i.d. inputs. Note that EM-GM-AMP-MOS assumes during recovery that the

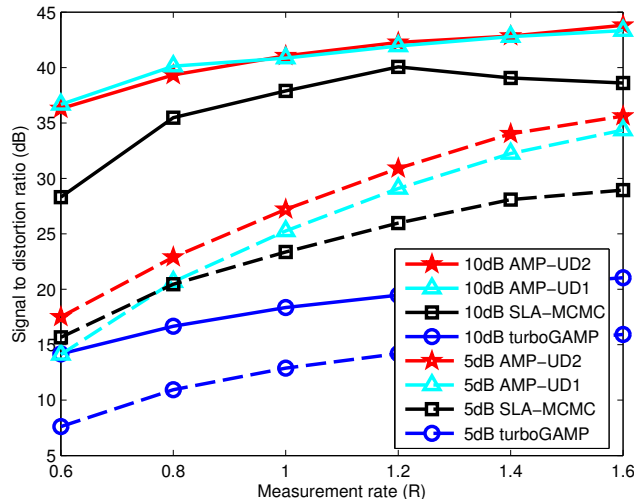


Fig. 5. Two AMP-UD implementations, SLA-MCMC, and turboGAMP reconstruction results for a dense two-state Markov signal with nonzero entries drawn from a Rademacher (± 1) distribution as a function of measurement rate. ($N = 10000$, SNR = 10 dB or 15 dB.)

input is i.i.d., whereas turboGAMP is designed for non-i.i.d. inputs with a known statistical model. We do not include results for other well-known CS algorithms such as compressive sensing matching pursuit (CoSaMP) [39], gradient projection for sparse reconstruction (GPSR) [40], or ℓ_1 minimization [2, 3], because their SDR performance is consistently weaker than the three algorithms being compared.

Sparse Laplace signal (i.i.d.): We tested i.i.d. sparse Laplace signals that follow the distribution $p_X(x) = 0.03\mathcal{L}(0, 1) + 0.97\delta(x)$, where $\mathcal{L}(0, 1)$ denotes a Laplacian distribution with mean zero and variance one, and $\delta(\cdot)$ is the delta function [41]. It is shown in Fig. 3 that the two AMP-UD implementations and EM-GM-AMP-MOS achieve the MMSE [42, 43], whereas SLA-MCMC has weaker performance, because the MCMC approach is expected to sample from the posterior and its MSE is twice the MMSE [14, 23].

Markov-uniform signal: Consider the two-state Markov state machine defined in Section II-B with $p_{01} = \frac{3}{970}$ and $p_{10} = \frac{1}{10}$. A Markov-uniform signal (MUnif for short) follows a uniform distribution $U[0, 1]$ at the nonzero state s_1 . These parameters lead to 3% nonzero entries in an MUnif signal on average. It is shown in Fig. 4 that at low SNR, the two AMP-UD implementations achieve higher SDR than SLA-MCMC and turboGAMP. At high SNR, the two AMP-UD implementations and turboGAMP have similar SDR performance, and are slightly better than SLA-MCMC. We highlight that turboGAMP needs side information about the Markovian structure of the signal, whereas the two AMP-UD implementations and SLA-MCMC do not.

Dense Markov-Rademacher signal: Consider the two-state Markov state machine defined in Section II-B with $p_{01} = \frac{3}{70}$ and $p_{10} = \frac{1}{10}$. A dense Markov Rademacher signal (MRad for short) takes values from $\{-1, +1\}$ with equal probability at s_1 . These parameters lead to 30% nonzero entries in an MRad signal on average. Because the MRad signal is dense (non-sparse), we must measure it with somewhat larger measurement rates and SNRs than before. It is shown in Fig. 5 that the two AMP-UD implementations and SLA-MCMC have better overall performance than turboGAMP. AMP-UD1 outperforms SLA-MCMC except for the lowest tested measurement rate

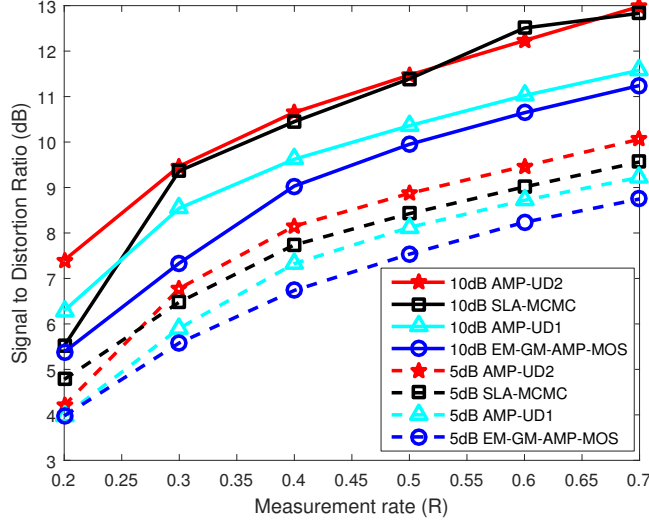


Fig. 6. Two AMP-UD implementations, SLA-MCMC, and EM-GM-AMP-MOS reconstruction results for a chirp sound clip as a function of measurement rate. ($N = 9600$, SNR = 5 dB or 10 dB.)

at low SNR, whereas AMP-UD2 outperforms SLA-MCMC consistently.

Chirp sound clip and speech signal: Our experiments up to this point use synthetic signals. We now evaluate the reconstruction quality of AMP-UD for two real-world signals. A “Chirp” sound clip and a speech signal are used. We cut a segment with length 9600 out of the “Chirp” and a segment with length 10560 out of the speech signal (denoted by \mathbf{x}), and performed a short-time discrete cosine transform (DCT) with window size, number of DCT points, and hop size all being 32. The resulting short-time DCT coefficients matrix are then vectorized to form a coefficient vector $\boldsymbol{\theta}$. Denoting the short-time DCT matrix by \mathbf{W}^{-1} , we have $\boldsymbol{\theta} = \mathbf{W}^{-1}\mathbf{x}$. Therefore, we can rewrite (1) as $\mathbf{y} = \boldsymbol{\Phi}\boldsymbol{\theta} + \mathbf{z}$, where $\boldsymbol{\Phi} = \mathbf{A}\mathbf{W}$. Our goal is to reconstruct $\boldsymbol{\theta}$ from the measurements \mathbf{y} and the matrix $\boldsymbol{\Phi}$. After we obtain the estimated coefficient vector $\hat{\boldsymbol{\theta}}$, the estimated signal is calculated as $\hat{\mathbf{x}} = \mathbf{W}\hat{\boldsymbol{\theta}}$. Although the coefficient vector $\boldsymbol{\theta}$ may exhibit some type of memory, it is not readily modeled in closed form, and so we cannot provide a valid model for turboGAMP [11]. Therefore, we use EM-GM-AMP-MOS [10] instead of turboGAMP [11]. The SDRs for the two AMP-UD implementations, SLA-MCMC and EM-GM-AMP-MOS [10] for the “Chirp” are plotted in Fig. 6 and the speech signal in Fig. 7. We can see that both AMP-UD implementations outperform EM-GM-AMP-MOS consistently, which implies that the simple i.i.d. model is suboptimal for these two real-world signals. Moreover, AMP-UD2 provides comparable and in most cases higher SDR than SLA-MCMC, which indicates that AMP-UD2 is more reliable in learning various statistical structures than SLA-MCMC. AMP-UD1 is the fastest among the four algorithms, but it may have lower reconstruction quality than AMP-UD2 and SLA-MCMC, owing to poor selection of the subsequences. It is worth mentioning that we have also run simulations on an electrocardiograph (ECG) signal, and EM-GM-AMP-MOS achieved similar SDR as the two AMP-UD implementations, which indicates that an i.i.d. model might be adequate to represent the coefficients of the ECG signal; the plot is omitted for brevity.

Runtime: The runtime of AMP-UD1 and AMP-UD2 for MUnif, MRad, and the speech signal is typically under 5 minutes and 10 minutes, respectively, but somewhat more for signals such as sparse Laplace and the chirp sound

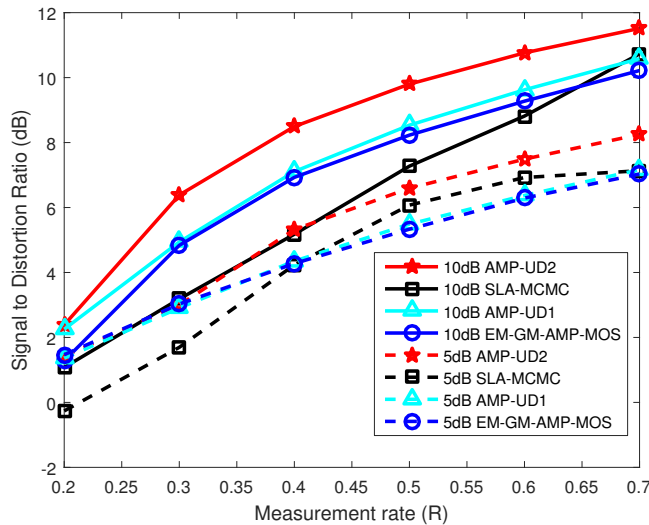


Fig. 7. Two AMP-UD implementations, SLA-MCMC, and EM-GM-AMP-MOS reconstruction results for a speech signal as a function of measurement rate. ($N = 10560$, SNR = 5 dB or 10 dB.)

clip that require a large number of Gaussian components to be fit. For comparison, the runtime of SLA-MCMC is typically an hour, whereas typical runtimes of EM-GM-AMP-MOS and turboGAMP are 30 minutes. To further accelerate AMP, we could consider parallel computing. That is, after clustering, the Gaussian mixture learning algorithm can be implemented simultaneously in different processors.

VII. CONCLUSION

In this paper, we introduced a universal compressed sensing recovery algorithm AMP-UD that applies our proposed universal denoiser (UD) within approximate message passing (AMP). AMP-UD is designed to reconstruct stationary ergodic signals from noisy linear measurements. The performance of two AMP-UD implementations was evaluated via simulations, where it was shown that AMP-UD achieves favorable signal to distortion ratios compared to existing universal algorithms, and that its runtime is typically faster.

AMP-UD combines three existing schemes: (i) AMP [6]; (ii) universal denoising [26]; and (iii) a density estimation approach based on Gaussian mixture (GM) fitting [24]. In addition to the algorithmic framework, we provided three specific contributions. First, we provided numerical results showing that SE holds for non-separable Bayesian sliding-window denoisers. Second, we modified the GM learning algorithm, and extended it to an i.i.d. denoiser. Third, we designed a universal denoiser that does not need to know the bounds of the input or require the input signal to be bounded. Two implementations of the universal denoiser were provided, with one being faster and the other achieving better reconstruction quality in terms of signal to distortion ratio.

There are numerous directions for future work. First, our current algorithm was designed to minimize the square error, and the denoiser could be modified to minimize other error metrics [44]. Second, AMP-UD was designed to reconstruct one-dimensional signals. In order to support applications that process multi-dimensional signals such as images, it might be instructive to employ universal image denoisers within AMP. Third, the relation between

the input length and the optimal window-size, as well as the exponential decay rate of the context weights, can be investigated. Finally, we can modify our work to support measurement noise with unknown distributions as an extension to adaptive generalized AMP [12].

APPENDIX

We follow the derivation in Figueiredo and Jain [24]. Denoting $\boldsymbol{\theta} = \{\alpha_s, \mu_s, \sigma_s^2\}_{s=1}^S$, the MML-based criterion is

$$\mathcal{L}(\mathbf{q}, \boldsymbol{\theta}) = \frac{n}{2} \sum_{s:\alpha_s>0} \log(N\alpha_s) + \frac{S_{nz}}{2} \log(N) - \log(p(\mathbf{q}|\boldsymbol{\theta})), \quad (17)$$

where $n = 2$ is the number of parameters per Gaussian component, and S_{nz} is the number of components with nonzero mixing probability α_s . The first term is the coding length of $\{\mu_s, \sigma_s^2\}_{s=1}^S$, because the expected number of data points that are from the s -th component is $N\alpha_s$, hence the effective sample size for estimating $\{\mu_s, \sigma_s^2\}$ is $N\alpha_s$. The second term is the coding length of α_s 's, because α_s 's are estimated from N data points. The third term is the coding length of the data sequence \mathbf{q} .

The *complete data* expression for $\log(p(\mathbf{q}|\boldsymbol{\theta}))$ is

$$\begin{aligned} \log(p(\mathbf{q}, \mathbf{x}, \mathbf{z}|\boldsymbol{\theta})) &= \sum_{i=1}^N \log\left(p(q_i, x_i, z_i^{(1)}, \dots, z_i^{(S)}|\boldsymbol{\theta})\right) \\ &= \sum_{i=1}^N \log\left(p(q_i, x_i|z_i^{(1)}, \dots, z_i^{(S)}, \boldsymbol{\theta})p(z_i^{(1)}, \dots, z_i^{(S)})\right) \\ &= \sum_{i=1}^N \log\left(\prod_{s=1}^k p(q_i, x_i|\{\mu_s, \sigma_s^2\})^{z_i^{(s)}} \prod_{s=1}^S \alpha_s^{z_i^{(s)}}\right) \\ &= \sum_{i=1}^N \log\left(\prod_{s=1}^S (\alpha_s p(q_i, x_i|\{\mu_s, \sigma_s^2\}))^{z_i^{(s)}}\right) \\ &= \sum_{i=1}^N \sum_{s=1}^S z_i^{(s)} \log(\alpha_s p(q_i, x_i|\{\mu_s, \sigma_s^2\})) \\ &= \sum_{i=1}^N \sum_{s=1}^S z_i^{(s)} \log(\alpha_s p(q_i|x_i)p(x_i|\{\mu_s, \sigma_s^2\})) \\ &= \sum_{i=1}^N \sum_{s=1}^S z_i^{(s)} \log(\alpha_s \mathcal{N}(q_i; x_i, \sigma_v^2) \mathcal{N}(x_i; \mu_s, \sigma_s^2)). \end{aligned}$$

Replace $\log(p(\mathbf{q}|\boldsymbol{\theta}))$ in (17) with $\log(p(\mathbf{q}, \mathbf{x}, \mathbf{z}|\boldsymbol{\theta}))$:

$$\mathcal{L}(\mathbf{q}, \mathbf{x}, \mathbf{z}, \boldsymbol{\theta}) = \frac{n}{2} \sum_{s:\alpha_s>0} \log(N\alpha_s) + \frac{S_{nz}}{2} \log(N) - \sum_{i=1}^N \sum_{s=1}^S z_i^{(s)} \log(\alpha_s \mathcal{N}(q_i; x_i, \sigma_v^2) \mathcal{N}(x_i; \mu_s, \sigma_s^2)). \quad (18)$$

Suppose $\widehat{\boldsymbol{\theta}}(t) = (\widehat{\alpha}_1(t), \dots, \widehat{\alpha}_S(t), \widehat{\mu}_1(t), \dots, \widehat{\mu}_S(t), \widehat{\sigma}_1^2(t), \dots, \widehat{\sigma}_S^2(t))$ is the estimate of the $\boldsymbol{\theta}$ at the t -th iteration.

$$\begin{aligned}
& \mathbb{E}[\log(p(\mathbf{q}, \mathbf{X}, \mathbf{Z}|\boldsymbol{\theta})) | \mathbf{q}, \widehat{\boldsymbol{\theta}}(t)] \\
&= \sum_{i=1}^N \sum_{s=1}^S \log(\alpha_s) \mathbb{E} \left[Z_i^{(s)} | \mathbf{q}, \widehat{\boldsymbol{\theta}}(t) \right] \\
&+ \sum_{i=1}^N \sum_{s=1}^S \mathbb{E} \left[Z_i^{(s)} \left(-\frac{1}{2} \log(2\pi\sigma_v^2) - \frac{(X_i - q_i)^2}{2\sigma_v^2} \right) | \mathbf{q}, \widehat{\boldsymbol{\theta}}(t) \right] \\
&+ \sum_{i=1}^N \sum_{s=1}^S \mathbb{E} \left[Z_i^{(s)} \left(-\frac{1}{2} \log(2\pi\sigma_s^2) - \frac{(X_i - \mu_s)^2}{2\sigma_s^2} \right) | \mathbf{q}, \widehat{\boldsymbol{\theta}}(t) \right] \\
&= \sum_{i=1}^N \sum_{s=1}^S \log(\alpha_s) \mathbb{E} \left[Z_i^{(s)} | \mathbf{q}, \widehat{\boldsymbol{\theta}}(t) \right] \\
&- \frac{1}{2} \sum_{i=1}^N \sum_{s=1}^S \log(2\pi\sigma_v^2) \mathbb{E} \left[Z_i^{(s)} | \mathbf{q}, \widehat{\boldsymbol{\theta}}(t) \right] - \frac{1}{2\sigma_v^2} \sum_{i=1}^N \sum_{s=1}^S \mathbb{E} \left[Z_i^{(s)} (X_i - q_i)^2 | \mathbf{q}, \widehat{\boldsymbol{\theta}}(t) \right] \\
&- \frac{1}{2} \sum_{i=1}^N \sum_{s=1}^S \log(2\pi\sigma_s^2) \mathbb{E} \left[Z_i^{(s)} | \mathbf{q}, \widehat{\boldsymbol{\theta}}(t) \right] - \frac{1}{2} \sum_{i=1}^N \sum_{s=1}^S \frac{1}{\sigma_s^2} \mathbb{E} \left[Z_i^{(s)} (X_i - \mu_s)^2 | \mathbf{q}, \widehat{\boldsymbol{\theta}}(t) \right] \\
&= \sum_{i=1}^N \sum_{s=1}^S \log(\alpha_s) \mathbb{E} \left[Z_i^{(s)} | \mathbf{q}, \widehat{\boldsymbol{\theta}}(t) \right] \\
&- \frac{1}{2} \sum_{i=1}^N \sum_{s=1}^S \log(2\pi\sigma_s^2) \mathbb{E} \left[Z_i^{(s)} | \mathbf{q}, \widehat{\boldsymbol{\theta}}(t) \right] - \frac{1}{2} \sum_{i=1}^N \sum_{s=1}^S \frac{1}{\sigma_s^2} \mathbb{E} \left[Z_i^{(s)} (X_i - \mu_s)^2 | \mathbf{q}, \widehat{\boldsymbol{\theta}}(t) \right] + C,
\end{aligned}$$

where C is a constant that does not depend on $\boldsymbol{\theta}$.

$$\begin{aligned}
\mathbb{E} \left[Z_i^{(s)} | \mathbf{q}, \widehat{\boldsymbol{\theta}}(t) \right] &= P \left(Z_i^{(s)} = 1 | \mathbf{q}, \widehat{\boldsymbol{\theta}}(t) \right) \\
&= \frac{\widehat{\alpha}_s(t) \mathcal{N}(q_i; \widehat{\mu}_s(t), \sigma_v^2 + \widehat{\sigma}_s(t)^2)}{\sum_{m=1}^S \widehat{\alpha}_m(t) \mathcal{N}(q_i; \widehat{\mu}_m(t), \sigma_v^2 + \widehat{\sigma}_m(t)^2)}, \\
\mathbb{E} \left[Z_i^{(s)} X_i | \mathbf{q}, \widehat{\boldsymbol{\theta}}(t) \right] &= \mathbb{E} \left[X_i | Z_i^{(s)} = 1, \mathbf{q}, \widehat{\boldsymbol{\theta}}(t) \right] P \left(Z_i^{(s)} = 1 | \mathbf{q}, \widehat{\boldsymbol{\theta}}(t) \right) \\
&= \left(\frac{\widehat{\sigma}_s^2(t)}{\widehat{\sigma}_s^2(t) + \sigma_v^2} (q_i - \widehat{\mu}_s(t)) + \widehat{\mu}_s(t) \right) P \left(Z_i^{(s)} = 1 | \mathbf{q}, \widehat{\boldsymbol{\theta}}(t) \right), \\
\mathbb{E} \left[Z_i^{(s)} X_i^2 | \mathbf{q}, \widehat{\boldsymbol{\theta}}(t) \right] &= \mathbb{E} \left[X_i^2 | Z_i^{(s)} = 1, \mathbf{q}, \widehat{\boldsymbol{\theta}}(t) \right] P \left(Z_i^{(s)} = 1 | \mathbf{q}, \widehat{\boldsymbol{\theta}}(t) \right) \\
&= \left(\frac{\sigma_v^2 \widehat{\sigma}_s^2(t)}{\widehat{\sigma}_s^2(t) + \sigma_v^2} + \left(\frac{\widehat{\sigma}_s^2(t)}{\widehat{\sigma}_s^2(t) + \sigma_v^2} (q_i - \widehat{\mu}_s(t)) + \widehat{\mu}_s(t) \right)^2 \right) P \left(Z_i^{(s)} = 1 | \mathbf{q}, \widehat{\boldsymbol{\theta}}(t) \right).
\end{aligned}$$

Denote

$$\begin{aligned}
w_i^{(s)}(t) &= \mathbb{E} \left[Z_i^{(s)} | \mathbf{q}, \widehat{\boldsymbol{\theta}}(t) \right], \\
a_i^{(s)}(t) &= \frac{\widehat{\sigma}_s^2(t)}{\widehat{\sigma}_s^2(t) + \sigma_v^2} (q_i - \widehat{\mu}_s(t)) + \widehat{\mu}_s(t), \\
v_i^{(s)}(t) &= \frac{\sigma_v^2 \widehat{\sigma}_s^2(t)}{\widehat{\sigma}_s^2(t) + \sigma_v^2}.
\end{aligned}$$

$$\begin{aligned} & \mathbb{E}[\log(p(\mathbf{q}, \mathbf{X}, \mathbf{Z}|\boldsymbol{\theta}))|\mathbf{q}, \widehat{\boldsymbol{\theta}}(t)] \\ &= \sum_{i=1}^N \sum_{s=1}^S \log(\alpha_s) w_i^{(s)}(t) - \frac{1}{2} \sum_{i=1}^N \sum_{s=1}^S \log(2\pi\sigma_s^2) w_i^{(s)}(t) - \frac{1}{2} \sum_{i=1}^N \sum_{s=1}^S \frac{w_i^{(s)}(t)}{\sigma_s^2} \left(v_i^{(s)}(t) + \left(a_i^{(s)}(t) - \widehat{\mu}_s(t) \right)^2 \right) + C. \end{aligned}$$

Therefore,

$$\begin{aligned} & \mathbb{E}[\mathcal{L}(\mathbf{q}, \mathbf{X}, \mathbf{Z}, \boldsymbol{\theta})|\mathbf{q}, \widehat{\boldsymbol{\theta}}(t)] \\ &= \frac{n}{2} \sum_{s:\alpha_s>0} \log(\alpha_s) - \sum_{i=1}^N \sum_{s:\alpha_s>0} \log(\alpha_s) w_i^{(s)} \\ &+ \frac{1}{2} \sum_{i=1}^N \sum_{s:\alpha_s>0} \log(2\pi\sigma_s^2) w_i^{(s)}(t) + \frac{1}{2} \sum_{i=1}^N \sum_{s:\alpha_s>0} \frac{w_i^{(s)}(t)}{\sigma_s^2} \left(v_i^{(s)}(t) + \left(a_i^{(s)}(t) - \widehat{\mu}_s(t) \right)^2 \right) + C', \quad (19) \end{aligned}$$

where C' is a constant that does not depend on $\boldsymbol{\theta}$.

Denote $Q(\boldsymbol{\theta}, \widehat{\boldsymbol{\theta}}(t)) = \mathbb{E}[\mathcal{L}(\mathbf{q}, \mathbf{x}, \mathbf{z}, \boldsymbol{\theta})|\mathbf{q}, \widehat{\boldsymbol{\theta}}(t)]$.

$$\begin{aligned} \frac{\partial}{\partial \mu_s} Q(\boldsymbol{\theta}, \widehat{\boldsymbol{\theta}}(t)) &= -\frac{1}{2} \sum_{i=1}^N \frac{w_i^{(s)}(t)}{\sigma_s^2} \left(2\mu_s - 2a_i^{(s)}(t) \right) = 0 \\ \widehat{\mu}_s(t+1) &= \frac{\sum_{i=1}^N w_i^{(s)}(t) a_i^{(s)}(t)}{\sum_{i=1}^N w_i^{(s)}(t)}. \\ \frac{\partial}{\partial \sigma_s^2} Q(\boldsymbol{\theta}, \widehat{\boldsymbol{\theta}}(t)) &= -\frac{1}{2} \sum_{i=1}^N \frac{w_i^{(s)}(t)}{\sigma_s^2} + \frac{1}{2} \sum_{i=1}^N \frac{w_i^{(s)}(t)}{\sigma_s^4} \left(v_i^{(s)}(t) + \left(a_i^{(s)}(t) - \widehat{\mu}_s(t+1) \right)^2 \right) = 0 \\ \widehat{\sigma}_s^2(t+1) &= \frac{\sum_{i=1}^N w_i^{(s)}(t) \left(v_i^{(s)}(t) + \left(a_i^{(s)}(t) - \widehat{\mu}_s(t+1) \right)^2 \right)}{\sum_{i=1}^N w_i^{(s)}(t)}. \end{aligned}$$

To estimate $\{\alpha_s\}$, notice that we have the constraints $0 \leq \alpha_s \leq 1, \forall s$ and $\sum_{s=1}^S \alpha_s = 1$. Collecting the terms in (19) that contain $\{\alpha_s\}$, we have

$$\sum_{s:\alpha_s>0} \log(\alpha_s) \left(\frac{n}{2} - \sum_{i=1}^N w_i^{(s)} \right) = -\log \left(\prod_{s:\alpha_s>0} \alpha_s^{\sum_{i=1}^N w_i^{(s)} - \frac{n}{2}} \right),$$

which is the negative log likelihood of a quantity that is proportional to a Dirichlet pdf of $(\alpha_s, \dots, \alpha_{S_{nz}})$, and its mode appears at

$$\alpha_s = \frac{\sum_{i=1}^N w_i^{(s)} - \frac{n}{2}}{\sum_{s=1}^{S_{nz}} \left(\sum_{i=1}^N w_i^{(s)} - \frac{n}{2} \right)}, \quad \sum_{i=1}^N w_i^{(s)} - \frac{n}{2} > 0.$$

Hence,

$$\widehat{\alpha}_s(t+1) = \frac{\max \left\{ \sum_{i=1}^N w_i^{(s)}(t) - \frac{n}{2}, 0 \right\}}{\sum_{s=1}^{S_{nz}} \max \left\{ \sum_{i=1}^N w_i^{(s)}(t) - \frac{n}{2}, 0 \right\}}.$$

ACKNOWLEDGEMENTS

We thank Mario Figueiredo and Tsachy Weissman for informative discussions; and Ahmad Beirami and Marco Duarte for detailed suggestions on the manuscript.

REFERENCES

- [1] Y. Ma, J. Zhu, and D. Baron, "Compressed sensing via universal denoising and approximate message passing," in *Proc. Allerton Conference Commun., Control, and Comput.*, Oct. 2014.
- [2] D. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.
- [3] E. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Inf. Theory*, vol. 52, no. 2, pp. 489–509, Feb. 2006.
- [4] R. Tibshirani, "Regression shrinkage and selection via the LASSO," *J. Royal Stat. Soc. Series B (Methodological)*, vol. 58, no. 1, pp. 267–288, 1996.
- [5] S. Chen, D. Donoho, and M. Saunders, "Atomic decomposition by basis pursuit," *SIAM J. Sci. Comp.*, vol. 20, no. 1, pp. 33–61, 1998.
- [6] D. L. Donoho, A. Maleki, and A. Montanari, "Message passing algorithms for compressed sensing," *Proc. Nat. Academy Sci.*, vol. 106, no. 45, pp. 18914–18919, Nov. 2009.
- [7] D. Baron, S. Sarvotham, and R. G. Baraniuk, "Bayesian compressive sensing via belief propagation," *IEEE Trans. Signal Process.*, vol. 58, pp. 269–280, Jan. 2010.
- [8] D. L. Donoho, A. Maleki, and A. Montanari, "Message Passing Algorithms for Compressed Sensing: I. Motivation and Construction," in *IEEE Inf. Theory Workshop*, Jan. 2010.
- [9] S. Rangan, "Generalized approximate message passing for estimation with random linear mixing," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, July 2011, pp. 2168–2172.
- [10] J. Vila and P. Schniter, "Expectation-maximization Gaussian-mixture approximate message passing," *IEEE Trans. Signal Process.*, vol. 61, no. 19, pp. 4658–4672, Oct. 2013.
- [11] J. Ziniel, S. Rangan, and P. Schniter, "A generalized framework for learning and recovery of structured sparse signals," in *Proc. IEEE Stat. Signal Process. Workshop (SSP)*, Aug. 2012, pp. 325–328.
- [12] U. Kamilov, S. Rangan, A. K. Fletcher, and M. Unser, "Approximate message passing with consistent parameter estimation and applications to sparse learning," *IEEE Trans. Inf. Theory*, vol. 60, no. 5, pp. 2969–2985, May 2014.
- [13] Y. Ma, D. Baron, and A. Beirami, "Mismatched estimation in large linear systems," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, July 2015, pp. 760–764.
- [14] D. L. Donoho, "The Kolmogorov sampler," Stanford University, Stanford, CA, Department of Statistics Technical Report 2002-4, Jan. 2002.
- [15] D. Donoho, H. Kakavand, and J. Mammen, "The simplest solution to an underdetermined system of linear equations," in *Proc. Int. Symp. Inf. Theory (ISIT)*, July 2006, pp. 1924–1928.
- [16] S. Jalali and A. Maleki, "Minimum complexity pursuit," in *Proc. Allerton Conference Commun., Control, Comput.*, Sept. 2011, pp. 1764–1770.
- [17] S. Jalali, A. Maleki, and R. G. Baraniuk, "Minimum complexity pursuit for universal compressed sensing," *IEEE Trans. Inf. Theory*, vol. 60, no. 4, pp. 2253–2268, Apr. 2014.
- [18] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York, NY, USA: Wiley-Interscience, 2006.
- [19] M. Li and P. M. B. Vitanyi, *An introduction to Kolmogorov complexity and its applications*. Springer-Verlag, New York, 2008.
- [20] D. Baron, "Information complexity and estimation," in *Workshop Inf. Theoretic Methods Sci. Eng. (WITMSE)*, Aug. 2011.
- [21] D. Baron and M. F. Duarte, "Universal MAP estimation in compressed sensing," in *Proc. Allerton Conference Commun., Control, and Comput.*, Sept. 2011, pp. 768–775.
- [22] J. Zhu, D. Baron, and M. F. Duarte, "Complexity-adaptive universal signal estimation for compressed sensing," in *Proc. IEEE Stat. Signal Process. Workshop (SSP)*, June 2014, pp. 416–419.
- [23] —, "Recovery from linear measurements with complexity-matching universal signal estimation," *IEEE Trans. Signal Process.*, vol. 63, no. 6, pp. 1512–1527, Mar. 2015.
- [24] M. Figueiredo and A. Jain, "Unsupervised learning of finite mixture models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, pp. 381–396, Mar. 2002.
- [25] K. Sivaramakrishnan and T. Weissman, "Universal denoising of discrete-time continuous-amplitude signals," *IEEE Trans. Inf. Theory*, vol. 54, no. 12, pp. 5632–5660, Dec. 2008.
- [26] —, "A context quantization approach to universal denoising," *IEEE Trans. Signal Process.*, vol. 57, no. 6, pp. 2110–2129, June 2009.

- [27] M. Bayati and A. Montanari, "The dynamics of message passing on dense graphs, with applications to compressed sensing," *IEEE Trans. Inf. Theory*, vol. 57, no. 2, pp. 764–785, Feb. 2011.
- [28] D. Donoho, I. Johnstone, and A. Montanari, "Accurate prediction of phase transitions in compressed sensing via a connection to minimax denoising," *IEEE Trans. Inf. Theory*, vol. 59, no. 6, pp. 3396–3433, June 2013.
- [29] J. Tan, Y. Ma, and D. Baron, "Compressive imaging via approximate message passing with image denoising," *IEEE Trans. Signal Process.*, vol. 63, no. 8, pp. 2085–2092, Apr. 2015.
- [30] C. Rush, A. Greig, and R. Venkataramanan, "Capacity-achieving sparse superposition codes via approximate message passing decoding," *Proc. Int. Symp. Inf. Theory (ISIT)*, June 2015.
- [31] C. Metzler, A. Maleki, and R. G. Baraniuk, "From denoising to compressed sensing," *Arxiv preprint arxiv:1406.4175v2*, June 2014.
- [32] S. Rangan, P. Schniter, and A. Fletcher, "On the convergence of approximate message passing with arbitrary matrices," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, July 2014, pp. 236–240.
- [33] A. Manoel, F. Krzakala, E. W. Tramel, and L. Zdeborová, "Sparse estimation with the swept approximated message-passing algorithm," *Arxiv preprint arxiv:1406.4311*, June 2014.
- [34] J. Vila, P. Schniter, S. Rangan, F. Krzakala, and L. Zdeborová, "Adaptive damping and mean removal for the generalized approximate message passing algorithm," in *IEEE Int. Conf. Acoustics, Speech, Signal Process. (ICASSP)*, Apr. 2015, pp. 2021–2025.
- [35] S. Rangan, A. K. Fletcher, P. Schniter, and U. S. Kamilov, "Inference for generalized linear models via alternating directions and bethe free energy minimization," in *Proc. Int. Symp. Inf. Theory (ISIT)*, June 2015, pp. 1640–1644.
- [36] A. Montanari, "Graphical models concepts in compressed sensing," *Compressed Sensing: Theory and Applications*, pp. 394–438, 2012.
- [37] J. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proc. 5th Berkeley Symp. Math. Stat. & Prob.*, vol. 1, no. 14, 1967, pp. 281–297.
- [38] A. Barron, J. Rissanen, and B. Yu, "The minimum description length principle in coding and modeling," *IEEE Trans. Inf. Theory*, vol. 44, no. 6, pp. 2743–2760, Oct. 1998.
- [39] D. Needell and J. A. Tropp, "CoSaMP: Iterative signal recovery from incomplete and inaccurate samples," *Appl. Computational Harmonic Anal.*, vol. 26, no. 3, pp. 301–321, May 2009.
- [40] M. Figueiredo, R. Nowak, and S. J. Wright, "Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems," *IEEE J. Select. Topics Signal Proces.*, vol. 1, pp. 586–597, Dec. 2007.
- [41] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*. McGraw Hill Book Co., 1991.
- [42] D. Guo, D. Baron, and S. Shamai, "A single-letter characterization of optimal noisy compressed sensing," in *Proc. 47th Allerton Conf. Commun., Control, and Comput.*, Sept. 2009, pp. 52–59.
- [43] S. Rangan, A. K. Fletcher, and V. K. Goyal, "Asymptotic analysis of MAP estimation via the replica method and applications to compressed sensing," *IEEE Trans. Inf. Theory*, vol. 58, no. 3, pp. 1902–1923, Mar. 2012.
- [44] J. Tan, D. Carmon, and D. Baron, "Signal estimation with additive error metrics in compressed sensing," *IEEE Trans. Inf. Theory*, vol. 60, no. 1, pp. 150–158, Jan. 2014.