An Overview of Multi-Processor Approximate Message Passing

Junan Zhu,* Ryan Pilgrim,† and Dror Baron†
*JPMorgan Chase & Co., New York, NY 10001, Email: jzhu9@ncsu.edu
†Department of Electrical and Computer Engineering, NC State University, Raleigh, NC 27695
Email: {rzpilgri,barondror}@ncsu.edu

***Abstract*—Approximate message passing (AMP) is an algorithmic framework for solving linear inverse problems from noisy measurements, with exciting applications such as reconstructing images, audio, hyper spectral images, and various other signals, including those acquired in compressive signal acquisiton systems. The growing prevalence of big data systems has increased interest in large-scale problems, which may involve huge measurement matrices that are unsuitable for conventional computing systems. To address the challenge of large-scale processing, multi-processor (MP) versions of AMP have been developed. We provide an overview of two such MP-AMP variants. In row-MP-AMP, each computing node stores a subset of the rows of the matrix and processes corresponding measurements. In column-MP-AMP, each node stores a subset of columns, and is solely responsible for reconstructing a portion of the signal. We will discuss pros and cons of both approaches, summarize recent research results for each, and explain when each one may be a viable approach. Aspects that are highlighted include some recent results on state evolution for both MP-AMP algorithms, and the use of data compression to reduce communication in the MP network.**

***Index Terms*—Approximate message passing, compressed sensing, distributed linear systems, inverse problems, lossy compression, optimization.**

I. Introduction

Many scientific and engineering problems can be modeled as solving a regularized linear inverse problem of the form

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{w}, \tag{1}$$

where the goal is to estimate the unknown $\mathbf{x} \in \mathbb{R}^N$ given the matrix $\mathbf{A} \in \mathbb{R}^{M\times N}$ and statistical information about the signal $\mathbf{x}$ and the noise $\mathbf{w} \in \mathbb{R}^M$. These problems have received significant attention in the compressed sensing literature [1, 2] with applications to image reconstruction [3], communication systems [4], and machine learning problems [5].

In recent years, many applications have seen explosive growth in the sizes of data sets. Some linear inverse problems, for example in hyper spectral image reconstruction [3, 6, 7], are so large that the $M \times N$ matrix elements cannot be stored on conventional computing systems. To solve these large-scale problems, it is possible to partition the matrix $\mathbf{A}$ among multiple computing nodes in multi-processor (MP) systems.

The matrix $\mathbf{A}$ can be partitioned in a column-wise or row-wise fashion, and the corresponding sub-matrices are stored at different processors. The partitioning style depends on data availability, computational considerations, and privacy concerns. Both types of partitioning result in reduced storage requirements per node and faster computation [8–16].

Row-wise partitioning: When the matrix is partitioned into rows, there are P *distributed nodes* (processor nodes) and a *fusion center*. Each distributed node stores $\frac{M}{P}$ rows of the matrix $\mathbf{A}$, and acquires the corresponding linear measurements of the underlying signal $\mathbf{x}$. Without loss of generality, we model the measurement system in distributed node $p \in \{1, ..., P\}$ as

$$y_i = \mathbf{a}_i\mathbf{x} + w_i,\ i \in \left\{\frac{M(p-1)}{P} + 1, ..., \frac{Mp}{P}\right\}, \tag{2}$$

where $\mathbf{a}_i$ is the i-th row of $\mathbf{A}$, and y_i and w_i are the i-th entries of $\mathbf{y}$ and $\mathbf{w}$, respectively. Once every y_i is collected, we run distributed algorithms among the fusion center and P distributed nodes to reconstruct the signal $\mathbf{x}$. Prior studies on solving row-wise partitioned linear inverse problems include extending existing algorithms such as least absolute shrinkage and selection operator (LASSO) [5] and iterative hard thresholding (IHT) to a distributed setting [8, 12].

Column-wise partitioning: Columns of the matrix $\mathbf{A}$ may correspond to features in feature selection problems [5]. In some applications, for example in healthcare when rows of the matrix correspond to patients, privacy concerns or other constraints prevent us from storing entire rows (corresponding to all the data about a patient) in individual processors, and column-wise partitioning becomes preferable. The (non-overlapping) column-wise partitioned linear inverse problem can be modeled as follows,

$$\mathbf{y} = \sum_{p=1}^{P} \mathbf{A}^p\mathbf{x}^p + \mathbf{w}, \tag{3}$$

where $\mathbf{A}^p \in \mathbb{R}^{M\times N_p}$ is the sub-matrix that is stored in processor p, and $\sum_{p=1}^{P} N_p = N$.

Many studies on solving the column-wise partitioned linear inverse problem (3) have been in the context of distributed feature selection. For example, Zhou *et al.* [17] modeled feature selection as a parallel group testing problem. Wang *et al.* [18] proposed to de-correlate the data matrix before partitioning, so that each processor can work independently using the de-correlated matrix without communication with other processors. Peng *et al.* [19] studied problem (3) in the context of optimization, where they proposed a greedy coordinate-block descent algorithm and a parallel implementation of the fast iterative shrinkage-thresholding algorithm (FISTA) [20].

This paper relies on approximate message passing (AMP) [21–24], an iterative framework that solves linear inverse problems. We overview the recent progress in under-

978-1-5090-4780-2/17/$31.00 ©2017 IEEE

standing the distributed AMP algorithm applied to either row-wise or column-wise partitioned linear inverse problems.

The rest of the paper is organized as follows. After reviewing the AMP literature in Section II, Section III discusses the row-partitioned version, and the column-partitioned version appears in Section IV. We conclude the paper in Section V.

II. Approximate Message Passing

To solve large-scale MP linear inverse problems partitioned either row-wise or column-wise, we use approximate message passing (AMP) [21–24], an iterative framework that solves linear inverse problems by successively decoupling [25–27] matrix channel problems into scalar channel denoising problems with additive white Gaussian noise (AWGN). AMP has received considerable attention because of its fast convergence, computational efficiency, and state evolution (SE) formalism [21, 23, 28], which offers a precise characterization of the AWGN denoising problem in each iteration. In the Bayesian setting, AMP often achieves the minimum mean squared error (MMSE) [24, 29] in the limit of large linear systems. Various extensions to AMP have been considered since AMP was initially introduced. Below, we summarize recent developments in AMP theory and application.

Generalizations of AMP: Recently, a number of authors have studied the incorporation of various non-separable denoisers within AMP [3, 30–34], generalization of the measurement matrix prior [35–39], and relaxation of assumptions on the probabilistic observation model [33, 38, 40]. AMP-based methods have also been applied to solve the bilinear inference problem [41–43], with matrix factorization applications.

Applications: The AMP framework and its many extensions have found applications in capacity-achieving sparse superposition codes [34], compressive imaging [30, 31, 44], hyperspectral image reconstruction [3] and hyperspectral unmixing [45], universal compressed sensing reconstruction [32], MIMO detection [4], and matrix factorization applications [41–43].

Multi-processor AMP: Recently, Zhu *et al.* [14, 15] studied the application of lossy compression in row-wise partitioned MP-AMP, such that the cost of running the reconstruction algorithm is minimized. Ma *et al.* [16] proposed a distributed version of AMP to solve column-wise partitioned linear inverse problems, with a rigorous study of state evolution.

Centralized AMP: Our model for the linear system (1) includes an independent and identically distributed (i.i.d.) Gaussian measurement matrix $\mathbf{A}$, i.e., $A_{i,j} \sim \mathcal{N}(0, \frac{1}{M})$.[1] The signal entries follow an i.i.d. distribution. The noise entries obey $w_i \sim \mathcal{N}(0, \sigma_W^2)$, where σ_W^2 is the noise variance.

Starting from $\mathbf{x}_0 = \mathbf{0}$ and $\mathbf{z}_0 = \mathbf{0}$, the AMP framework [21] proceeds iteratively according to

$$\mathbf{x}_{t+1} = \eta_t(\mathbf{A}^T\mathbf{z}_t + \mathbf{x}_t), \tag{4}$$

$$\mathbf{z}_t = \mathbf{y} - \mathbf{A}\mathbf{x}_t + \frac{1}{\kappa}\mathbf{z}_{t-1}\langle d\eta_{t-1}(\mathbf{A}^T\mathbf{z}_{t-1} + \mathbf{x}_{t-1})\rangle, \tag{5}$$

[1]When the matrix $\mathbf{A}$ is not i.i.d. Gaussian, the use of damping or other variants of AMP algorithms such as Swept AMP [35] and VAMP [39] is necessary in order for the algorithm to converge. This paper only considers an i.i.d. Gaussian matrix $\mathbf{A}$ in order to present some theoretical results; the theoretic understanding of using AMP in general matrices is less mature.

where $\eta_t(\cdot)$ is a denoising function, $d\eta_t(\cdot) = \frac{d\eta_t(\cdot)}{d\{\cdot\}}$ is shorthand for the derivative of $\eta_t(\cdot)$, and $\langle \mathbf{u} \rangle = \frac{1}{N}\sum_{i=1}^{N} u_i$ for some vector $\mathbf{u} \in \mathbb{R}^N$. The subscript t represents the iteration index, T denotes transpose, and $\kappa = \frac{M}{N}$ is the measurement rate. Owing to the decoupling effect [25–27], in each AMP iteration [22, 23], the vector $\mathbf{f}_t = \mathbf{A}^T\mathbf{z}_t + \mathbf{x}_t$ in (4) is statistically equivalent to the input signal $\mathbf{x}$ corrupted by AWGN $\mathbf{e}_t$ generated by a source $E \sim \mathcal{N}(0, \sigma_t^2)$,

$$\mathbf{f}_t = \mathbf{x} + \mathbf{e}_t. \tag{6}$$

In large systems ($N \to \infty, \frac{M}{N} \to \kappa$), a useful property of AMP [22, 23] is that the noise variance σ_t^2 of the equivalent scalar channel (6) evolves following SE:

$$\sigma_{t+1}^2 = \sigma_W^2 + \frac{1}{\kappa}\text{MSE}(\eta_t, \sigma_t^2), \tag{7}$$

where the mean squared error (MSE) is $\text{MSE}(\eta_t, \sigma_t^2) = \mathbb{E}_{X,E}\left[(\eta_t(X+E) - X)^2\right]$, $\mathbb{E}_{X,W}(\cdot)$ is expectation with respect to X and E, and $X \sim f_X$ is the source that generates $\mathbf{x}$. Formal statements for SE appear in prior work [22, 23, 28].

The SE in (7) can also be expressed in the following recursion,

$$\begin{aligned} \tau_t^2 &= \sigma_W^2 + \sigma_t^2, \\ \sigma_{t+1}^2 &= \kappa^{-1}\mathbb{E}\left[(\eta_t(X + \tau_t Z) - X)^2\right], \end{aligned} \tag{8}$$

where Z is a standard normal random variable (RV) that is independent of X, and $\sigma_0^2 = \kappa^{-1}\mathbb{E}[X^2]$.

This paper considers the Bayesian setting, which assumes knowledge of the true prior for the signal $\mathbf{x}$. Therefore, the MMSE-achieving denoiser is conditional expectation, $\eta_t(\cdot) = \mathbb{E}[\mathbf{x}|\mathbf{f}_t]$, which is easily obtained. Other denoisers such as soft thresholding [21–23] yield MSE's that are greater than that of the Bayesian denoiser. When the true prior for $\mathbf{x}$ is unavailable, parameter estimation techniques can be used [32, 46, 47].

III. Row-wise MP-AMP

A. Lossless R-MP-AMP

Han *et al.* [48] proposed AMP for row-wise partitioned MP linear inverse problems (R-MP-AMP) for a network with P processor nodes and a fusion center. Each processor node stores rows of the matrix $\mathbf{A}$ as in (2), carries out the decoupling step of AMP, and generates part of the pseudo data $\mathbf{f}_t^p$. The fusion center merges the pseudo data sent by all processor nodes, $\mathbf{f}_t = \sum_{p=1}^{P}\mathbf{f}_t^p$, denoises $\mathbf{f}_t$, and sends back the denoised $\mathbf{f}_t$ to each processor node. The detailed steps are summarized in Algorithm 1. Mathematically, Algorithm 1 is equivalent to the centralized AMP in (4)-(5). Therefore, the SE in (7) tracks the evolution of Algorithm 1. Note that $\mathbf{a}^p$ denotes the row partition of the matrix $\mathbf{A}$ at processor p.

B. Lossy R-MP-AMP

In lossless R-MP-AMP (Algorithm 1), the processor nodes and fusion center send real-valued vectors of length N to each other, i.e., $\mathbf{f}_t^p$ and $\mathbf{x}_{t+1}$, at floating point precision. However, in some applications it is costly to send uncompressed real numbers at full precision. To reduce the communication load of inter-node messages, we use lossy compression [49, 50].

Algorithm 1 R-MP-AMP (lossless)

Inputs to Processor p**:** $\mathbf{y}$, $\mathbf{a}^p$, $\widehat{t}$
Initialization: $\mathbf{x}_0 = \mathbf{0},\ \mathbf{z}_0^p = \mathbf{0}, \forall p$
for $t = 1 : \widehat{t}$ **do**
At Processor p:
$\mathbf{z}_t^p = \mathbf{y}^p - \mathbf{a}^p\mathbf{x}_t + \frac{1}{\kappa}\mathbf{z}_{t-1}^p g_{t-1}, \quad \mathbf{f}_t^p = \frac{1}{P}\mathbf{x}_t + (\mathbf{a}^p)^T\mathbf{z}_t^p$
At fusion center:
$\mathbf{f}_t = \sum_{p=1}^P \mathbf{f}_t^p, \quad g_t = \langle d\eta_t(\mathbf{f}_t)\rangle, \quad \mathbf{x}_{t+1} = \eta_t(\mathbf{f}_t)$
Output from fusion center: $\mathbf{x}_{\widehat{t}}$

Applying lossy compression to the messages sent from each processor node to the fusion center, we obtain the lossy R-MP-AMP [13, 15] steps as described in Algorithm 2, where $Q(\cdot)$ denotes quantization.

Algorithm 2 R-MP-AMP (lossy)

Inputs to Processor p**:** $\mathbf{y}$, $\mathbf{a}^p$, $\widehat{t}$
Initialization: $\mathbf{x}_0 = \mathbf{0},\ \mathbf{z}_0^p = \mathbf{0}, \forall p$
for $t = 1 : \widehat{t}$ **do**
At Processor p:
$\mathbf{z}_t^p = \mathbf{y}^p - \mathbf{a}^p\mathbf{x}_t + \frac{1}{\kappa}\mathbf{z}_{t-1}^p g_{t-1}, \quad \mathbf{f}_t^p = \frac{1}{P}\mathbf{x}_t + (\mathbf{a}^p)^T\mathbf{z}_t^p$
At fusion center:
$\mathbf{f}_{Q,t} = \sum_{p=1}^P Q(\mathbf{f}_t^p), \quad g_t = \langle d\eta_t(\mathbf{f}_{Q,t})\rangle,$
$\mathbf{x}_{t+1} = \eta_t(\mathbf{f}_{Q,t})$
Output from fusion center: $\mathbf{x}_{\widehat{t}}$

The reader might notice that the fusion center also needs to transmit the denoised signal vector $\mathbf{x}_t$ and a scalar g_{t-1} to the distributed nodes. The transmission of the scalar g_{t-1} is negligible relative to the transmission of $\mathbf{x}_t$, and the fusion center may broadcast $\mathbf{x}_t$ so that naive compression of $\mathbf{x}_t$, such as compression with a fixed quantizer, is sufficient. Hence, we will not discuss possible lossy compression of the messages transmitted by the fusion center.

Assume that we quantize $\mathbf{f}_t^p, \forall p$, and use C bits on average to encode the quantized vector $Q(\mathbf{f}_t^p) \in \mathcal{X}^N \subset \mathbb{R}^N$, where $\mathcal{X}$ is a set of representation levels. The per-symbol *coding rate* is $R = \frac{C}{N}$. We incur an *expected distortion*

$$D_t^p = \mathbb{E}\left[\frac{1}{N}\sum_{i=1}^N (Q(f_{t,i}^p) - f_{t,i}^p)^2\right]$$

at iteration t in each processor node,[2] where $Q(f_{t,i}^p)$ and $f_{t,i}^p$ are the i-th entries of the vectors $Q(\mathbf{f}_t^p)$ and $\mathbf{f}_t^p$, respectively, and expectation is over $\mathbf{f}_t^p$. When the size of the problem grows, i.e., $N \to \infty$, the rate-distortion (RD) function, denoted by $R(D)$, offers the information theoretic limit on the coding rate R for communicating a long sequence up to distortion D [49–51]. A pivotal conclusion from RD theory is that coding rates can be greatly reduced even if D is small. The function $R(D)$ can be computed in various ways [52–54] and can be achieved by an RD-optimal quantization scheme in the limit of large N. Other quantization schemes will require larger coding rates to achieve the same expected distortion D.

Assume that appropriate vector quantization (VQ) schemes [51, 55, 56] that achieve $R(D)$ are applied within each MP-AMP iteration. The signal *at the fusion center* before denoising can then be modeled as

$$\mathbf{f}_{Q,t} = \sum_{p=1}^P Q(\mathbf{f}_t^p) = \mathbf{x} + \mathbf{e}_t + \mathbf{n}_t, \tag{9}$$

where $\mathbf{e}_t$ is the equivalent scalar channel noise (6) and $\mathbf{n}_t$ is the overall quantization error. For large block sizes, we expect the VQ quantization error $\mathbf{n}_t$ to resemble additive white Gaussian noise with variance PD_t that is independent of $\mathbf{x}+\mathbf{e}_t$ at high rates, or at all rates using dithering [57].

State evolution for lossy R-MP-AMP: Han *et al.* suggest that SE for lossy R-MP-AMP [13] follows

$$\sigma_{t+1}^2 = \sigma_W^2 + \frac{1}{\kappa}\text{MSE}(\eta_t, \sigma_t^2 + PD_t), \tag{10}$$

where σ_t^2 can be estimated by $\widehat{\sigma}_t^2 = \frac{1}{M}\|\mathbf{z}_t\|_2^2$ with $\|\cdot\|_p$ denoting the ℓ_p norm [22, 23], and σ_{t+1}^2 is the variance of $\mathbf{e}_{t+1}$. The rigorous justification of (10) by extending the framework put forth by Bayati and Montanari [23] and Rush and Venkataramanan [58] is left for future work. Instead, we argue that lossy SE (10) asymptotically tracks the evolution of σ_t^2 in lossy MP-AMP in the limit of low normalized distortion $\frac{PD_t}{\sigma_t^2} \to 0$. Our argument is comprised of three parts: (*i*) $\mathbf{e}_t$ and $\mathbf{n}_t$ (9) are approximately independent in the limit of $\frac{PD_t}{\sigma_t^2} \to 0$, (*ii*) $\mathbf{e}_t + \mathbf{n}_t$ is approximately independent of $\mathbf{x}$ in the limit of $\frac{PD_t}{\sigma_t^2} \to 0$, and (*iii*) lossy SE (10) holds if (*i*) and (*ii*) hold. The first part ($\mathbf{e}_t$ and $\mathbf{n}_t$ are independent) ensures that we can track the variance of $\mathbf{e}_t + \mathbf{n}_t$ with $\sigma_t^2 + PD_t$. The second part ($\mathbf{e}_t + \mathbf{n}_t$ is independent of $\mathbf{x}$) ensures that lossy MP-AMP follows lossy SE (10) as it falls under the general framework discussed in Bayati and Montanari [23] and Rush and Venkataramanan [58]. Hence, the third part of our argument holds. The numerical justification of these three parts appears in Zhu *et al.* [15, 59].

Optimal coding rates: Denote the coding rate used to transmit $Q(\mathbf{f}_t^p)$ at iteration t by R_t. The sequence of $R_t,\ t = 1, ..., \widehat{t}$, where $\widehat{t}$ is the total number of MP-AMP iterations, is called the *coding rate sequence*, and is denoted by the vector $\mathbf{R} = [R_1, ..., R_{\widehat{t}}]$. Given the coding rate sequence $\mathbf{R}$, the distortion D_t can be evaluated with $R(D)$, and the scalar channel noise variance σ_t^2 can be evaluated with (10). Hence, the MSE for $\mathbf{R}$ can be predicted. The coding rate sequence $\mathbf{R}$ can be optimized using dynamic programming (DP) [15, 60]. That said, our recent theoretical analysis of lossy R-MP-AMP has revealed that the coding rate is linear in the limit of EMSE$\to 0$, where EMSE denotes excess MSE (EMSE=MSE-MMSE). This result is summarized in the following theorem.

Theorem 1 (Linearity of the coding rate sequence [15]): Supposing that lossy SE (10) holds, we have

$$\lim_{t\to\infty} \frac{D_{t+1}^*}{D_t^*} = \theta,$$

[2]Because we assume that $\mathbf{A}$ and $\mathbf{z}$ are both i.i.d., the expected distortions are the same over all P nodes, and can be denoted by D_t for simplicity. Note also that $D_t = \mathbb{E}[(Q(f_{t,i}^p) - f_{t,i}^p)^2]$ due to $\mathbf{x}$ being i.i.d.

where $\theta = \frac{N}{M}\text{MSE}'(\sigma_\infty^2)$ and D_t^* denotes the optimal distortion at iteration t. Further, define the additive growth rate at iteration t as $R_{t+1} - R_t$. The additive growth rate for the optimal coding rate sequence $\mathbf{R}^*$ satisfies

$$\lim_{t\to\infty}\left(R_{t+1}^* - R_t^*\right) = \frac{1}{2}\log_2\left(\frac{1}{\theta}\right).$$

Comparison of DP results to Theorem 1: We run DP (discussed in Zhu *et al.* [15]) to find an optimal coding rate sequence $\mathbf{R}^*$ to reconstruct a *Bernoulli-Gaussian* (BG) signal, whose entries follow

$$x_j \sim \rho\mathcal{N}(0,1) + (1-\rho)\delta(x_j), \tag{11}$$

where $\delta(\cdot)$ is the Dirac delta function and ρ is called the *sparsity rate* of the signal. The detailed setting is: sparsity rate $\rho = 0.2$, $P = 100$ nodes, measurement rate $\kappa = 1$, noise variance $\sigma_W^2 = 0.01$, and normalized cost ratio of computation to communication $b = 0.782$ (a formal definition of b appears in [15]). The goal is to achieve a desired EMSE of 0.005 dB, i.e., $10\log_{10}\left(1 + \frac{\text{EMSE}}{\text{MMSE}}\right) = 0.005$. We use uniform ECSQ [49, 51] with optimal block entropy coding [51] at each processor node and the corresponding relation between the rate R_t and distortion D_t of ECSQ in the DP scheme. We know that ECSQ achieves a coding rate within an additive constant of the RD function $R(D)$ at high rates [51]. Therefore, the additive growth rate of the optimal coding rate sequence obtained for ECSQ will be the same as the additive growth rate if the RD relation is modeled by $R(D)$ [49–51].

The resulting optimal coding rate sequence is plotted in Fig. 1. The additive growth rate of the last six iterations is $\frac{1}{6}(R_{12}^* - R_6^*) = 0.742$, while the asymptotic additive growth rate according to Theorem 1 is $\frac{1}{2}\log_2\left(\frac{1}{\theta}\right) \approx 0.751$. Note that the discrepancy of 0.009 between the additive growth rate from the simulation and the asymptotic additive growth rate is within the numerical precision of our DP scheme. In conclusion, our numerical result matches the theoretical prediction of Theorem 1.

Algorithm 3 C-MP-AMP (lossless)

Inputs to Processor p: $\mathbf{y}$, $\mathbf{A}^p$, $\{\widehat{t}_s\}_{s=0,\ldots,\widehat{s}}$ (maximum number of inner iterations at each outer iteration)
Initialization: $\mathbf{x}_{0,\widehat{t}_0}^p = \mathbf{0}$, $\mathbf{z}_{0,\widehat{t}_0-1}^p = \mathbf{0}$, $\mathbf{r}_{0,\widehat{t}_0}^p = \mathbf{0}$, $\forall p$
for $s = 1:\widehat{s}$ **do** (loop over outer iterations)
At fusion center: $\mathbf{g}_s = \sum_{u=1}^P \mathbf{r}_{s-1,\widehat{t}_{s-1}}^u$
At Processor p:
$\mathbf{x}_{s,0}^p = \mathbf{x}_{s-1,\widehat{t}_{s-1}}^p$, $\mathbf{r}_{s,0}^p = \mathbf{r}_{s-1,\widehat{t}_{s-1}}^p$
for $t = 0:\widehat{t}_s - 1$ **do** (loop over inner iterations)
$\mathbf{z}_{s,t}^p = \mathbf{y} - \mathbf{g}_s - \left(\mathbf{r}_{s,t}^p - \mathbf{r}_{s,0}^p\right)$
$\mathbf{x}_{s,t+1}^p = \eta_{s,t}(\mathbf{x}_{s,t}^p + (\mathbf{A}^p)^T\mathbf{z}_{s,t}^p)$
$\mathbf{r}_{s,t+1}^p = \mathbf{A}^p\mathbf{x}_{s,t+1}^p - \frac{\mathbf{z}_{s,t}^p}{M}\sum_{i=1}^{N_p}\eta'_{s,t}([\mathbf{x}_{s,t}^p + (\mathbf{A}^p)^T\mathbf{z}_{s,t}^p]_i)$
Output from processor p: $\mathbf{x}_{\widehat{s},\widehat{t}_{\widehat{s}}}^p$

IV. Column-wise MP-AMP

In our proposed column-wise multiprocessor AMP (C-MP-AMP) algorithm [16], the fusion center collects vectors that represent the estimates of the portion of the measurement vector $\mathbf{y}$ contributed by the data from individual processors. The sum of these vectors is computed in the fusion center and transmitted to all processors. Each processor performs standard AMP iterations with a new equivalent measurement vector, which is computed using the vector received from the fusion center. The pseudocode for C-MP-AMP is presented in Algorithm 3.

State evolution: Similar to AMP, the dynamics of the C-MP-AMP algorithm can be characterized by an SE formula. Let $(\sigma_{0,\widehat{t}}^p)^2 = \kappa_p^{-1}\mathbb{E}[X^2]$, where $\kappa_p = M/N_p$, $\forall p = 1,\ldots,P$. For outer iterations $1 \le s \le \widehat{s}$ and inner iterations $0 \le t \le \widehat{t}_s$, we define the sequences $\{(\sigma_{s,t}^p)^2\}$ and $\{(\tau_{s,t}^p)^2\}$ as

$$(\sigma_{s,0}^p)^2 = (\sigma_{s-1,\widehat{t}}^p)^2, \tag{12}$$

$$(\tau_{s,t}^p)^2 = \sigma_W^2 + \sum_{u=1}^P(\sigma_{s,0}^u)^2 + \left((\sigma_{s,t}^p)^2 - (\sigma_{s,0}^p)^2\right), \tag{13}$$

$$(\sigma_{s,t+1}^p)^2 = \kappa_p^{-1}\mathbb{E}\left[\left(\eta_{s,t}(X + \tau_{s,t}^p Z) - X\right)^2\right], \tag{14}$$

where Z is standard normal and independent of X. With these definitions, we have the following theorem for C-MP-AMP.

Theorem 2 ([16]): Under the assumptions listed in [58, Section 1.1], for $p = 1,\ldots,P$, let $M/N_p \to \kappa_p \in (0,\infty)$ be a constant. Define $N = \sum_{p=1}^P N_p$. Then for any PL(2) function[3] $\phi : \mathbb{R}^2 \to \mathbb{R}$, we have

$$\lim_{N\to\infty}\frac{1}{N_p}\sum_{i=1}^{N_p}\phi([x_{s,t+1}^p]_i, x_i^p) \overset{\text{a.s.}}{=} \mathbb{E}\left[\phi(\eta_{s,t}(X + \tau_{s,t}^p Z), X)\right], \forall p,$$

where $\mathbf{x}_{s,t+1}^p$ is generated by the C-MP-AMP algorithm, $\tau_{s,t}^p$ is defined in (12–14), x_i^p is the i^{th} element in x^p, x^p is the true signal in the p^{th} processor, $X \sim p_X$, and Z is a standard normal RV that is independent of X.

Remark 1: C-MP-AMP converges to a fixed point that is no worse than that of AMP. This statement can be demonstrated as follows. When C-MP-AMP converges, the quantities in (12–14) do not keep changing, hence we can drop all the iteration indices for fixed point analysis. Notice that the last term on the right hand side (RHS) of (13) vanishes, which leaves the RHS independent of p. Denote $(\tau_{s,t}^p)^2$ by τ^2 for all s,t,p, and plug (14) into (13), then

$$\tau^2 = \sigma_W^2 + \sum_{p=1}^P \kappa_p^{-1}\mathbb{E}\left[(\eta(X+\tau Z) - X)^2\right]$$
$$\overset{(a)}{=} \sigma_W^2 + \kappa^{-1}\mathbb{E}\left[(\eta(X+\tau Z) - X)^2\right],$$

which is identical to the fixed point equation obtained from (8), where (a) holds because $\sum_{p=1}^P \kappa_p^{-1} = \sum_{p=1}^P \frac{N_p}{M} = \frac{N}{M}$. Because AMP always converges to the worst fixed point of (8) [24], the average asymptotic performance of C-MP-AMP is at least as good as AMP.

Remark 2: The asymptotic dynamics of C-MP-AMP can be identical to AMP with a specific communication schedule. This

[3] A function $f : \mathbb{R}^m \to \mathbb{R}$ is *pseudo-Lipschitz* of order-2, denoted PL(2), if there exists a constant $L > 0$ such that for all $x, y \in \mathbb{R}^m$, $|\phi(x) - \phi(y)| \le L(1 + \|x\| + \|y\|)\|x - y\|$, where $\|\cdot\|$ denotes the Euclidean norm.

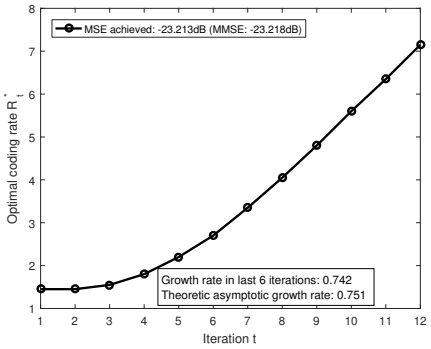

Fig. 1. Low-EMSE growth rate of optimal coding rate sequence per DP vs. asymptotic growth rate $\frac{1}{2}\log_2\left(\frac{1}{\theta}\right)$. (BG signal (11), $\rho = 0.2,\ \kappa = 1,\ P = 100, \sigma_W^2 = 0.01,\ b = 0.782$.)

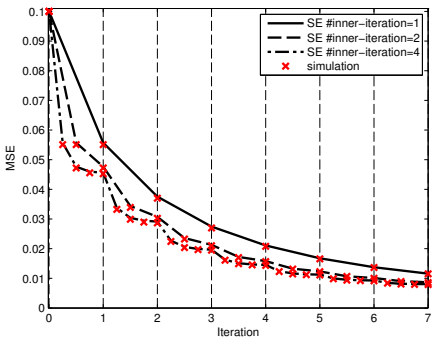

Fig. 2. Verification of SE for C-MP-AMP with various communication schedules. (P=3, N=30000, M=9000, SNR=15dB.)

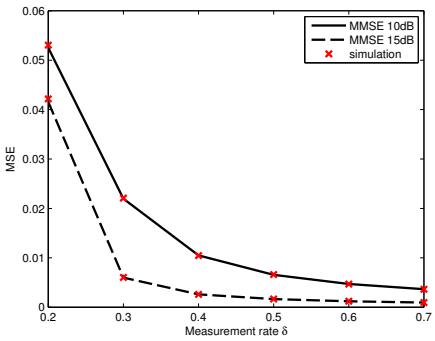

Fig. 3. Verification that C-MP-AMP achieves the MMSE at various measurement rates $\kappa = M/N$ and SNR levels. (P=3, N=30000.)

can be achieved by letting $\widehat{t}_s = 1, \forall s$. In this case, the quantity $(\tau_{s,t}^p)$ is involved only for $t = 0$. Because the last term in (13) is 0 when $t = 0$, the computation of $(\tau_{s,0}^p)^2$ is independent of p. Therefore, $\tau_{s,0}^p$ are again equal for all p. Dropping the processor index for $(\tau_{s,t}^p)^2$, the recursion in (12–14) can be simplified as

$$(\tau_{s,0})^2 = \sigma_W^2 + \sum_{p=1}^{P} \kappa_p^{-1}\mathbb{E}\left[(\eta_{s,0}(X+\tau_{s,0}Z)-X)^2\right]$$
$$= \sigma_W^2 + \kappa^{-1}\mathbb{E}\left[(\eta_{s-1,0}(X+\tau_{s-1,0}Z)-X)^2\right],$$

where the iteration evolves over s, which is identical to (8) evolving over t.

Numerical results for SE: We provide numerical results for C-MP-AMP for the Gaussian matrix setting, where SE is justified rigorously. We simulate i.i.d. Bernoulli-Gaussian signals (11) with $\rho = 0.1$. The measurement noise vector $\mathbf{w}$ has i.i.d. Gaussian $\mathcal{N}(0, \sigma_W^2)$ entries, where σ_W^2 depends on the signal to noise ratio (SNR) as SNR $:= 10\log_{10}\left((N\mathbb{E}[X^2])/(M\sigma_W^2)\right)$. The estimation function $\eta_{s,t}$ is defined as $\eta_{s,t}(u) = \mathbb{E}[X|X + \tau_{s,t}^p Z = u]$, where Z is a standard normal RV independent of X, and $\tau_{s,t}^p$ is estimated by $\|\mathbf{z}_{s,t}^p\|/\sqrt{M}$, which is implied by SE. All numerical results are averaged over 50 trials.

Let us show that the MSE of C-MP-AMP is accurately predicted by SE when the matrix $\mathbf{A}$ has i.i.d. Gaussian entries with $A_{i,j} \sim \mathcal{N}(0, 1/M)$. It can be seen from Fig. 2 that the MSE achieved by C-MP-AMP from simulations (red crosses) matches the MSE predicted by SE (black curves) at every outer iteration s and inner iteration t for various choices of numbers of inner iterations (the number of red crosses within a grid).

As discussed in Remark 1, the average estimation error of C-MP-AMP is no worse than that of AMP, which implies that C-MP-AMP can achieve the MMSE of large random linear systems [26] when AMP achieves it.[4] This is verified in Fig. 3.

V. Discussion

This overview paper discussed multi-processor (MP) approximate message passing (AMP) for solving linear inverse problems, where the focus was on two variants for partitioning the measurement matrix. In row-MP-AMP, each processor uses entire rows of the measurement matrix, and decouples statistical information from those rows to scalar channels. The multiple scalar channels, each corresponding to one processor, are merged at a fusion center. We showed how lossy compression can reduce communication requirements in this row-wise variant. In column-MP-AMP, each node is responsible for some entries of the signal. While we have yet to consider lossy compression in column-MP-AMP, it offers privacy advantages, because entire rows need not be stored. Ongoing work can consider lossy compression of inter-processor messages in column-MP-AMP, as well as rigorous state evolution analyses.

[4] AMP can achieve the MMSE in the limit of large linear systems when the model parameters (κ, SNR, and sparsity of $\mathbf{x}$) are within a region [24].

Acknowledgments

The authors were supported by the National Science Foundation (NSF) under grant ECCS-1611112. Subsets of this overview paper appeared in our earlier works, including in Han *et al.* [13], Zhu *et al.* [14, 15], and Ma *et al.* [16]. Finally, we thank Yanting Ma for numerous helpful discussions.

References

[1] D. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.
[2] 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.
[3] J. Tan, Y. Ma, H. Rueda, D. Baron, and G. Arce, "Compressive hyperspectral imaging via approximate message passing," *IEEE J. Sel. Topics Signal Process.*, vol. 10, no. 2, pp. 389–401, Mar. 2016.
[4] C. Jeon, R. Ghods, A. Maleki, and C. Studer, "Optimality of large MIMO detection via approximate message passing," in *Proc. IEEE Int. Symp. Inf. Theory*, Hong Kong, Hong Kong, June 2015, pp. 1227–1231.
[5] T. Hastie, R. Tibshirani, and J. H. Friedman, *The Elements of Statistical Learning*, Springer, Aug. 2001.
[6] H. Arguello and G. Arce, "Code aperture optimization for spectrally agile compressive imaging," *J. Opt. Soc. Am.*, vol. 28, no. 11, pp. 2400–2413, Nov. 2011.
[7] H. Arguello, H. Rueda, Y. Wu, D. W. Prather, and G. R. Arce, "Higher-order computational model for coded aperture spectral imaging," *Appl. Optics*, vol. 52, no. 10, pp. D12–D21, Mar. 2013.
[8] J. Mota, J. Xavier, and P. Aguiar, "Distributed basis pursuit," *IEEE Trans. Signal Process.*, vol. 60, no. 4, pp. 1942–1956, Apr. 2012.
[9] S. Patterson, Y. C. Eldar, and I. Keidar, "Distributed compressed sensing for static and time-varying networks," *IEEE Trans. Signal Process.*, vol. 62, no. 19, pp. 4931–4946, Oct. 2014.

[10] P. Han, R. Niu, M. Ren, and Y. C. Eldar, "Distributed approximate message passing for sparse signal recovery," in *Proc. IEEE Global Conf. Signal Inf. Process. (GlobalSIP)*, Atlanta, GA, Dec. 2014, pp. 497–501.

[11] C. Ravazzi, S. M. Fosson, and E. Magli, "Distributed iterative thresholding for ℓ_0/ℓ_1 -regularized linear inverse problems," *IEEE Trans. Inf. Theory*, vol. 61, no. 4, pp. 2081–2100, Apr. 2015.

[12] P. Han, R. Niu, and Y. C. Eldar, "Communication-efficient distributed IHT," in *Proc. Signal Process. with Adaptive Sparse Structured Representations Workshop (SPARS)*, Cambridge, United Kingdom, July 2015.

[13] P. Han, J. Zhu, R. Niu, and D. Baron, "Multi-processor approximate message passing using lossy compression," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Shanghai, China, Mar. 2016, pp. 6240–6244.

[14] J. Zhu, A. Beirami, and D. Baron, "Performance trade-offs in multi-processor approximate message passing," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Barcelona, Spain, July 2016, pp. 680–684.

[15] J. Zhu, D. Baron, and A. Beirami, "Optimal trade-offs in multi-processor approximate message passing," *Arxiv preprint arXiv:1601.03790*, Nov. 2016.

[16] Y. Ma, Y. M. Lu, and D. Baron, "Multiprocessor approximate message passing with column-wise partitioning," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Process. (ICASSP)*, New Orleans, LA, Mar. 2017, Accepted for publication.

[17] Y. Zhou, U. Porwal, C. Zhang, H. Ngo, L. Nguyen, C. Ré, and V. Govindaraju, "Parallel feature selection inspired by group testing," in *Neural Inf. Process. Syst. (NIPS)*, Dec. 2014, pp. 3554–3562.

[18] X. Wang, D. Dunson, and C. Leng, "DECOrrelated feature space partitioning for distributed sparse regression," *Arxiv preprint arXiv:1602.02575*, Feb. 2016.

[19] Z. Peng, M. Yan, and W. Yin, "Parallel and distributed sparse optimization," in *Proc. IEEE 47th Asilomar Conf. Signals, Syst., and Comput.*, Nov. 2013, pp. 659–646.

[20] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM J. Imag. Sci.*, vol. 2, no. 1, pp. 183–202, Mar. 2009.

[21] 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.

[22] A. Montanari, "Graphical models concepts in compressed sensing," *Compressed Sensing: Theory and Applications*, pp. 394–438, 2012.

[23] 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.

[24] F. Krzakala, M. Mézard, F. Sausset, Y. Sun, and L. Zdeborová, "Probabilistic reconstruction in compressed sensing: Algorithms, phase diagrams, and threshold achieving matrices," *J. Stat. Mech. – Theory E.*, vol. 2012, no. 08, pp. P08009, Aug. 2012.

[25] T. Tanaka, "A statistical-mechanics approach to large-system analysis of CDMA multiuser detectors," *IEEE Trans. Inf. Theory*, vol. 48, no. 11, pp. 2888–2910, Nov. 2002.

[26] D. Guo and S. Verdú, "Randomly spread CDMA: Asymptotics via statistical physics," *IEEE Trans. Inf. Theory*, vol. 51, no. 6, pp. 1983–2010, June 2005.

[27] D. Guo and C. C. Wang, "Multiuser detection of sparsely spread CDMA," *IEEE J. Sel. Areas Commun.*, vol. 26, no. 3, pp. 421–431, Apr. 2008.

[28] C. Rush and R. Venkataramanan, "Finite-sample analysis of approximate message passing," *Proc. Int. Symp. Inf. Theory (ISIT)*, June 2016.

[29] J. Zhu and D. Baron, "Performance regions in compressed sensing from noisy measurements," in *Proc. IEEE Conf. Inf. Sci. Syst. (CISS)*, Baltimore, MD, Mar. 2013.

[30] 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.

[31] C. A. Metzler, A. Maleki, and R. G. Baraniuk, "From denoising to compressed sensing," *IEEE Trans. Inf. Theory*, vol. 62, no. 9, pp. 5117–5144, Sept. 2016.

[32] Y. Ma, J. Zhu, and D. Baron, "Approximate message passing algorithm with universal denoising and Gaussian mixture learning," *IEEE Trans. Signal Process.*, vol. 65, no. 21, pp. 5611–5622, Nov. 2016.

[33] A. Javanmard and A. Montanari, "State evolution for general approximate message passing algorithms, with applications to spatial coupling," *Arxiv preprint arXiv:1211.5164*, Dec. 2012.

[34] 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.

[35] 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.

[36] 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. Acoust., Speech, Signal Process. (ICASSP)*, Apr. 2015, pp. 2021–2025.

[37] S. Rangan, A. K. Fletcher, Philip 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.

[38] B. Çakmak, O. Winther, and B. H. Fleury, "S-AMP: Approximate Message Passing for General Matrix Ensembles," in *Proc. IEEE Int. Symp. Inf. Theory*, Hong Kong, Hong Kong, June 2015, pp. 2807–2811.

[39] S. Rangan, P. Schniter, and A. K. Fletcher, "Vector approximate message passing," *Arxiv preprint arXiv:1610.03082v1*, Oct. 2016.

[40] S. Rangan, "Generalized approximate message passing for estimation with random linear mixing," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, St. Petersburg, Russia, July 2011, pp. 2168–2172.

[41] J. Parker, P. Schniter, and V. Cevher, "Bilinear generalized approximate message passing—Part I: Derivation," *IEEE Trans. Signal Process.*, vol. 62, no. 22, pp. 5839–5853, Nov. 2014.

[42] J. Parker, P. Schniter, and V. Cevher, "Bilinear generalized approximate message passing—Part II: Applications," *IEEE Trans. Signal Process.*, vol. 62, no. 22, pp. 5854–5867, Nov. 2014.

[43] Y. Kabashima, F. Krzakala, M. Mézard, A. Sakata, and L. Zdeborová, "Phase transitions and sample complexity in Bayes-optimal matrix factorization," *IEEE Trans. Signal Process.*, vol. 62, no. 7, pp. 4228–4265, July 2017.

[44] S. Som and P. Schniter, "Compressive imaging using approximate message passing and a Markov-tree prior," *IEEE Trans. Signal Process.*, vol. 60, no. 7, pp. 3439–3448, July 2012.

[45] J. Vila, P. Schniter, and J. Meola, "Hyperspectral unmixing via turbo bilinear generalized approximate message passing," *IEEE Trans. Comput. Imag.*, vol. 1, no. 3, pp. 143–158, Sept. 2015.

[46] 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.

[47] 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.

[48] P. Han, R. Niu, and Y. C. Eldar, "Modified distributed iterative hard thresholding," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Brisbane, Australia, Apr. 2015, pp. 3766–3770.

[49] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, New York, NY, USA: Wiley-Interscience, 2006.

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

[51] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*, Kluwer, 1993.

[52] S. Arimoto, "An algorithm for computing the capacity of an arbitrary discrete memoryless channel," *IEEE Trans. Inf. Theory*, vol. 18, no. 1, pp. 14–20, Jan. 1972.

[53] R. E. Blahut, "Computation of channel capacity and rate-distortion functions," *IEEE Trans. Inf. Theory*, vol. 18, no. 4, pp. 460–473, July 1972.

[54] K. Rose, "A mapping approach to rate-distortion computation and analysis," *IEEE Trans. Inf. Theory*, vol. 40, no. 6, pp. 1939–1952, Nov. 1994.

[55] Y. Linde, A. Buzo, and R. M. Gray, "An algorithm for vector quantizer design," *IEEE Trans. Commun.*, vol. 28, no. 1, pp. 84–95, Jan. 1980.

[56] R. M. Gray, "Vector quantization," *IEEE ASSP Mag.*, vol. 1, no. 2, pp. 4–29, Apr. 1984.

[57] R. Zamir and M. Feder, "On lattice quantization noise," *IEEE Trans. Inf. Theory*, vol. 42, no. 4, pp. 1152–1159, July 1996.

[58] C. Rush and R. Venkataramanan, "Finite-sample analysis of approximate message passing," *Arxiv preprint arXiv:1606.01800*, June 2016.

[59] J. Zhu, *Statistical Physics and Information Theory Perspectives on Linear Inverse Problems*, Ph.D. thesis, North Carolina State University, Raleigh, NC, Jan. 2017.

[60] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, The MIT Press, Cambridge, MA, third edition, 2009.