

MULTI-PROCESSOR APPROXIMATE MESSAGE PASSING USING LOSSY COMPRESSION

Puxiao Han,^v Junan Zhu,ⁿ Ruixin Niu,^v and Dror Baronⁿ

^v: Virginia Commonwealth University, Dept. Electrical and Computer Engineering
Richmond, VA 23284, U.S.A. Email: {hanp, rniu}@vcu.edu

ⁿ: North Carolina State University, Dept. Electrical and Computer Engineering
Raleigh, NC 27695, U.S.A. Email: {jzhu9, barondror}@ncsu.edu

ABSTRACT

In this paper, a communication-efficient multi-processor compressed sensing framework based on the approximate message passing algorithm is proposed. We perform lossy compression on the data being communicated between processors, resulting in a reduction in communication costs with a minor degradation in recovery quality. In the proposed framework, a new state evolution formulation takes the quantization error into account, and analytically determines the coding rate required in each iteration. Two approaches for allocating the coding rate, an online back-tracking heuristic and an optimal allocation scheme based on dynamic programming, provide significant reductions in communication costs.

Index Terms— lossy compression, multi-processor approximate message passing, rate distortion function.

1. INTRODUCTION

Compressed sensing (CS) [1, 2] has numerous applications in various areas of signal processing. Due to the curse of dimensionality, it can be demanding to perform CS on a single processor. Furthermore, clusters comprised of many processors have the potential to accelerate computation. Hence, multi-processor CS (MP-CS) has become of recent interest [3–5].

We consider MP-CS systems comprised of two parts: (i) local computation (LC) is performed at each processor, and (ii) global computation (GC) obtains an estimate of the unknown signal after processors exchange the results of LC. In our previous work [6], we developed an MP-CS framework based on the approximate message passing (AMP) algorithm [7], in which a GC approach performs AMP in an MP system, providing the same recovery result as centralized AMP. We chose AMP, because it is analytically tractable due to the state evolution (SE) [8, 9] formalism, and can be extended to Bayesian CS [9, 10], matrix completion [11], and non-negative principal component analysis (PCA) [12].

The work of P. Han and R. Niu was supported in part by the VCU Presidential Research Quest Fund. The work of J. Zhu and D. Baron was supported in part by the National Science Foundation under Grant CCF-1217749 and the U.S. Army Research Office under Contract W911NF-14-1-0314.

Compared with many results on distributed computation, optimization, and network topology [3, 13] in MP-CS, a modest subset of the literature considers the communication costs of the GC step [4, 5, 14, 15]. In this paper, we still consider multi-processor AMP (MP-AMP), and focus more on the communication costs. In contrast to our prior work [6], we are willing to accept a minor decrease in recovery quality while providing significant and often dramatic communications savings. Such results are especially well-suited to clusters where communication between servers is costly. To achieve the reduction in communication costs, we use lossy compression to reduce the inter-processor communication costs, and provide a modified SE formulation that accounts for quantization error. Two approaches for allocating the coding rate, an online back-tracking heuristic and an optimal allocation scheme based on dynamic programming, provide significant reductions in communication costs. Furthermore, we consider Bayesian AMP, which achieves better recovery accuracy than non-Bayesian AMP [7] by assuming that the unknown signal follows a known prior distribution.

In the following, bold capital and bold lower-case letters are used to denote matrices and vectors respectively, capital letters without bold typically refer to dimensionality or random variables, and $[\cdot]^T$ denotes vector or matrix transposition. The ℓ_2 norm of a vector is denoted by $\|\cdot\|$, $\mathcal{N}(\mu, \sigma^2)$ is a Gaussian distribution with mean μ and variance σ^2 , and $\mathcal{U}[a, b]$ is a continuous uniform distribution within $[a, b]$.

2. THE CENTRALIZED AMP ALGORITHM

Approximate message passing (AMP) [7] is a statistical algorithm derived from the theory of probabilistic graphical models [16]. Given noisy measurements $\mathbf{y} = \mathbf{A}\mathbf{s}_0 + \mathbf{e}$ of the unknown signal $\mathbf{s}_0 \in \mathbb{R}^N$, where elements in \mathbf{s}_0 are independent and identically distributed (i.i.d.) realizations of a scalar random variable $S_0 \sim p_{S_0}$, $\mathbf{A} \in \mathbb{R}^{M \times N}$ is the sensing matrix with entries \sim i.i.d. $\mathcal{N}(0, 1/M)$, and $\mathbf{e} \in \mathbb{R}^M$ is additive measurement noise, which is i.i.d. $\mathcal{N}(0, \sigma_e^2)$, AMP iteratively recovers \mathbf{s}_0 , starting from an initial estimate $\mathbf{x}_0 = \mathbf{0}$ and residual $\mathbf{z}_0 = \mathbf{y}$:

$$\mathbf{f}_t = \mathbf{x}_t + \mathbf{A}^T \mathbf{z}_t, \quad (1)$$

$$\mathbf{x}_{t+1} = \eta_t(\mathbf{f}_t), \quad (2)$$

$$\mathbf{z}_{t+1} = \mathbf{y} - \mathbf{A}\mathbf{x}_{t+1} + (N/M)\overline{\eta'_t(\mathbf{f}_t)}\mathbf{z}_t, \quad (3)$$

where t is the iteration number, the bar above the vector in (3) denotes its empirical average, η_t is known as the denoising function or denoiser, and η'_t denotes its derivative.

According to Bayati and Montanari [9], as $N \rightarrow \infty$ and $M/N = \kappa > 0$, the elements of \mathbf{f}_t in (1) follow i.i.d. $F_t = S_0 + \sigma_t Z$, where $Z \sim \mathcal{N}(0, 1)$ and the sequence $\{\sigma_t^2\}$ satisfies

$$\begin{aligned} \sigma_{t+1}^2 &= \sigma_e^2 + (1/\kappa)\mathbb{E}[\eta_t(S_0 + \sigma_t Z) - S_0]^2 \\ &= \sigma_e^2 + (1/\kappa)\mathbb{E}\|\mathbf{x}_{t+1} - \mathbf{s}_0\|^2/N. \end{aligned} \quad (4)$$

Note that $\sigma_0^2 = \sigma_e^2 + (1/\kappa)\mathbb{E}[S_0]^2$; equation (4) is known as state evolution (SE), and the optimal denoiser for mean square error (MSE) is the conditional mean [9, 17]:

$$\eta_t(F_t) = \mathbb{E}[S_0 | S_0 + \sigma_t Z = F_t]. \quad (5)$$

In this paper, we assume that S_0 follows the Bernoulli Gaussian distribution:

$$p_{S_0}(s) = \epsilon \mathcal{N}(s; \mu_s, \sigma_s^2) + (1 - \epsilon)\delta(s), \quad (6)$$

where $\delta(s)$ denotes the Dirac delta function, and S_0 typically has mean $\mu_s = 0$. The denoiser is easily derived using (5).

As a measure of the measurement noise level and recovery accuracy, we define the signal-to-noise-ratio (SNR) as

$$\begin{aligned} \text{SNR} &= 10 \log_{10} (\mathbb{E}[\|\mathbf{A}\mathbf{s}_0\|^2] / \mathbb{E}[\|\mathbf{e}\|^2]) \\ &\approx 10 \log_{10} (\mathbb{E}[\|\mathbf{s}_0\|^2] / \mathbb{E}[\|\mathbf{e}\|^2]) = 10 \log_{10} (\rho / \sigma_e^2), \end{aligned}$$

where $\rho = \epsilon/\kappa$, and the signal-to-distortion-ratio (SDR) at iteration t as

$$\text{SDR}(t) = 10 \log_{10} (\mathbb{E}[\|\mathbf{s}_0\|^2] / \mathbb{E}[\|\mathbf{x}_t - \mathbf{s}_0\|^2]).$$

Using the SE equation in (4), we have

$$\text{SDR}(t) = 10 \log_{10} [\rho / (\sigma_t^2 - \sigma_e^2)].$$

Note that the Bernoulli Gaussian assumption in this paper is only for illustration, and our work is easily extended to other prior distributions p_{S_0} .

3. MULTI-PROCESSOR AMP FRAMEWORK

3.1. Communication in Multi-Processor AMP

Consider a system with P processors and one fusion center. Each processor $p \in \{1, \dots, P\}$ takes M/P rows of \mathbf{A} , namely \mathbf{A}^p , and obtains $\mathbf{y}^p = \mathbf{A}^p \mathbf{s}_0 + \mathbf{e}^p$. The procedures in (1) — (3) can then be rewritten in a distributed manner:

Local Computation (LC) performed by each processor p :

$$\begin{aligned} \mathbf{z}_t^p &= \mathbf{y}^p - \mathbf{A}^p \mathbf{x}_t + (1/\kappa)\overline{\eta'_t(\mathbf{f}_{t-1})}\mathbf{z}_{t-1}^p, \\ \mathbf{f}_t^p &= \mathbf{x}_t/P + (\mathbf{A}^p)^T \mathbf{z}_t^p. \end{aligned}$$

Global Computation (GC) performed by the fusion center:

$$\mathbf{f}_t = \sum_{p=1}^P \mathbf{f}_t^p, \overline{\eta'_t(\mathbf{f}_t)}, \text{ and } \mathbf{x}_{t+1} = \eta_t(\mathbf{f}_t).$$

It can be seen that in the GC step of MP-AMP, each processor p sends \mathbf{f}_t^p to the fusion center, and the fusion center sums them to obtain \mathbf{f}_t and \mathbf{x}_{t+1} , and sends \mathbf{x}_{t+1} to each processor.¹ Our goal in this paper is to reduce these communication costs while barely impacting recovery performance.

Suppose that all the elements in \mathbf{f}_t^p are computed as 32-bit single-precision floating-point numbers. Because SE is robust to small perturbations [7, 8], we can compress \mathbf{f}_t^p lossily up to some reasonable distortion level, and send the compressed output to the fusion center. To ensure that this error is indeed a “small perturbation,” we require the error to be additive and, if possible, white and Gaussian, so that we can analyze the relationship between the error and coding rate.

3.2. Lossy Compression of \mathbf{f}_t^p

Before we propose specific lossy compression approaches, we describe an important property of MP-AMP. In addition to the well-known Gaussianity of the vector $\mathbf{f}_t - \mathbf{s}_0$ [9], numerical results show that elements of $\mathbf{f}_t^p - (1/P)\mathbf{s}_0$ are also i.i.d. Gaussian with mean 0 and variance σ_t^2/P . Furthermore, $\mathbf{f}_t^p - (1/P)\mathbf{s}_0$ and $\mathbf{f}_t^q - (1/P)\mathbf{s}_0$ are independent for different processors p and q . In light of this property, \mathbf{f}_t^p can be described as a scalar channel:

$$F_t^p = S_0/P + (\sigma_t/\sqrt{P})Z_p, \text{ where } Z_p \sim \mathcal{N}(0, 1).$$

For the Bernoulli Gaussian distribution (6),

$$F_t^p \sim \epsilon \mathcal{N}(\mu_s/P, (\sigma_s^2 + P\sigma_t^2)/P^2) + (1 - \epsilon)\mathcal{N}(0, \sigma_t^2/P).$$

Scalar Quantization: Next, we propose a uniform quantizer with entropy coding, also known as entropy coded scalar quantization (ECSQ) [18].

Let $\Psi(u)$ denote the characteristic function of F_t^p , it can be shown that

$$\begin{aligned} |\Psi(u)| &\leq \epsilon \exp[-0.5(\sigma_s^2 + P\sigma_t^2)u^2/P^2] \\ &\quad + (1 - \epsilon) \exp(-0.5\sigma_t^2 u^2/P) \leq \exp(-0.5\sigma_t^2 u^2/P) \end{aligned}$$

is nearly band-limited. Due to this property, it is possible to develop a uniform quantizer of $\mathbf{f}_t^p \sim$ i.i.d. F_t^p , where the quantization error \mathbf{v}_t^p is approximately statistically equivalent to a uniformly distributed noise $V_t^p \sim \mathcal{U}[-0.5\Delta_Q, 0.5\Delta_Q]$ uncorrelated to F_t^p . Actually, a quantization bin size $\Delta_Q \leq 2\sigma_t/\sqrt{P}$ will suffice for validation of $\mathbf{v}_t^p \sim$ i.i.d. V_t^p [19].

The fusion center will receive the quantized data $\tilde{\mathbf{f}}_t^p \sim$ i.i.d. \tilde{F}_t^p , and calculate $\tilde{\mathbf{f}}_t = \sum_{p=1}^P \tilde{\mathbf{f}}_t^p \sim$ i.i.d. \tilde{F}_t , where

$$\tilde{F}_t = \sum_{p=1}^P \tilde{F}_t^p = F_t + V_t, \text{ and } V_t = \sum_{p=1}^P V_t^p. \quad (7)$$

Applying the central limit theorem, V_t approximately follows $\mathcal{N}(0, P\sigma_Q^2)$ for large P , where $\sigma_Q^2 = \Delta_Q^2/12$.

Entropy Coding and Optimum Bit Rate: Let p_i be the probability that F_t^p falls into the i -th quantization bin. The entropy of quantized F_t^p, \tilde{F}_t^p , is $H_Q = -\sum_i p_i \log_2(p_i)$ [20],

¹In order to calculate each \mathbf{z}_{t+1}^p , the fusion center also needs to send $\overline{\eta'_t(\mathbf{f}_t)}$ to all the processors. This is a scalar, and the corresponding communication cost is negligible compared with that of transmitting a vector.

that is, the sensors need H_Q bits on average to present each element in $\tilde{\mathbf{f}}_t^p$ to the fusion center, which is achievable through entropy coding [20].

In rate distortion (RD) theory [20], we are given a length- n random sequence $Y_n = \{Y_{n,i}\}_{i=1}^n \sim \text{i.i.d. } Y$, and our goal is to identify a reconstruction sequence $\hat{Y}_n = \{\hat{Y}_{n,i}\}_{i=1}^n$ that can be encoded at low rate while the distortion $d(Y_n, \hat{Y}_n) = \frac{1}{n} \sum_i d(Y_{n,i}, \hat{Y}_{n,i})$ (e.g., squared error distortion) between the input and reconstruction sequences is small. RD theory has characterized the fundamental best-possible trade-off between the distortion $D = d(Y_n, \hat{Y}_n)$ and coding rate $R(D)$, which is called the rate distortion function. The RD function $R(D)$ can be computed numerically (cf. Blahut [21] and Arimoto [22]). For the uniform quantizer that yields a quantization MSE of σ_Q^2 with a coding rate H_Q bits per element, the RD function will give a bit rate $R(D = \sigma_Q^2) < H_Q$, which is achievable through vector quantization [18].

New SE Equation: For both ECSQ and RD-based vector quantization that lead to a quantization MSE of σ_Q^2 , the fusion center will have $\tilde{F}_t = S_0 + \sqrt{\sigma_t^2 + P\sigma_Q^2} \tilde{Z}$, where $\tilde{Z} \sim \mathcal{N}(0, 1)$. The new denoiser and SE equation become

$$\eta_t^Q(\tilde{F}_t) = \mathbb{E} \left[S_0 \mid S_0 + \sqrt{\sigma_t^2 + P\sigma_Q^2} \tilde{Z} = \tilde{F}_t \right] \text{ and} \\ \sigma_{t+1}^2 = \sigma_e^2 + (1/\kappa) \mathbb{E} \left[\eta_t^Q \left(S_0 + \sqrt{\sigma_t^2 + P\sigma_Q^2} \tilde{Z} \right) - S_0 \right]^2. \quad (8)$$

Currently, we only consider compression of \mathbf{f}_t^p . When broadcast from the fusion center to the P processors is allowed in the network topology, the communication cost of sending \mathbf{x}_t – even uncompressed – is smaller than that of communicating the P vectors \mathbf{f}_t^p . We are considering the case where broadcast is not allowed in our ongoing work.

3.3. Online Back-tracking (BT-MP-AMP)

Let $\sigma_{t,C}^2$ and $\sigma_{t,D}^2$ denote the σ_t^2 obtained by centralized AMP (4) and MP-AMP (8), respectively. In order to reduce communication while maintaining high fidelity, we first constrain $\sigma_{t,D}^2$ so that it will not deviate much from $\sigma_{t,C}^2$, and then determine the minimum coding rate required in each iteration. This can be done through an online back-tracking algorithm, which we name BT-MP-AMP and present below.

In each iteration t , before quantizing \mathbf{f}_t^p , we first compute $\sigma_{t+1,C}^2$ for the next iteration. Then we find the maximum quantization MSE σ_Q^2 allowed so that the ratio $\sigma_{t+1,D}^2 / \sigma_{t+1,C}^2$ does not exceed some constant, provided that the required bit rate does not exceed some threshold. Based on the obtained σ_Q^2 we construct the corresponding quantizer.

Note that the SE in (8) is only an approximation, and we do not know the true value of $\sigma_{t,D}^2$ in the current iteration. To better predict $\sigma_{t+1,D}^2$, we use $\hat{\sigma}_{t,D}^2 = \|\mathbf{z}_t^p\|^2 / M$, which is a good estimator for $\sigma_{t,D}^2$ [8, 9], to compute $\sigma_{t+1,D}^2$. To obtain $\hat{\sigma}_{t,D}^2$, each processor p sends the scalar $\|\mathbf{z}_t^p\|^2$ to the fusion center, which then sends the scalar $\hat{\sigma}_{t,D}^2 = \sum_{p=1}^P \|\mathbf{z}_t^p\|^2 / M$

to all the processors. The corresponding communication cost is also negligible compared with that of communicating \mathbf{f}_t^p .

3.4. Dynamic Programming (DP-MP-AMP)

While back-tracking is a useful heuristic, it is possible for a given coding budget R per element, total number of AMP iterations T , and initial noise level σ_0^2 in the scalar channel to compute the coding rate allocations among the AMP iterations that minimize the final MSE, $\sigma_{T,D}^2$.

To do so, note that we can evaluate $\sigma_{t,C}^2$ offline and hence obtain the number of iterations required to reach the steady state, which would be a reasonable choice for T . Second, recalling the new SE equation in (8), $\sigma_{t,D}^2$ depends on $\sigma_{t-1,D}^2$ and σ_Q^2 , which is a function of R_t , the coding rate allocated in the t -th iteration. Therefore, we can rewrite $\sigma_{t,D}^2$ as follows:

$$\sigma_{t,D}^2 = f_1(\sigma_{t-1,D}^2, R_t) = f_2(\sigma_{t-2,D}^2, R_{t-1}, R_t) \\ = \dots = f_t(\sigma_0^2, R_1, \dots, R_{t-1}, R_t), \quad (9)$$

that is, given σ_0^2 , $\sigma_{T,D}^2$ is only a function of R_t for $t \in \{1, 2, \dots, T\}$. Denoting $\mathcal{F}_T(R) = \{R_1, \dots, R_T \geq 0 : \sum_{t=1}^T R_t = R\}$, minimizing $\sigma_{T,D}^2$ for a given R can be formulated as the following optimization problem:

$$\min_{\mathcal{F}_T(R)} \sigma_{T,D}^2 = \min_{\mathcal{F}_T(R)} f_T(\sigma_0^2, R_1, \dots, R_T). \quad (10)$$

Since $\sigma_{t,D}^2$ is increasing with $\sigma_{t-1,D}^2$, it is easy to verify the following recursive relationship:

$$\min_{\mathcal{F}_T(R)} \sigma_{T,D}^2 = \min_{0 \leq R_T \leq R} f_1 \left(\min_{\mathcal{F}_{T-1}(R-R_T)} \sigma_{T-1,D}^2, R_T \right) = \dots,$$

which makes the problem solvable through dynamic programming (DP).

To implement DP, we need to discretize $\mathcal{F}_T(R)$ into $\{R_1, \dots, R_T \in \Omega : \sum_{t=1}^T R_t = R\}$, where $\Omega = \{R^{(1)}, \dots, R^{(S)}\}$ with $R^{(s)} = R(s-1)/(S-1)$, $\forall s \in \{1, \dots, S\}$. In this paper, we set the bit rate resolution $\Delta R = R/(S-1) = 0.1$ bits per element. Then, we create an $S \times T$ array Σ , with the element in the s -th row ($s \in \{1, \dots, S\}$) and t -th column ($t \in \{1, \dots, T\}$) denoted as $\sigma_D^2(s, t)$, storing the optimal value of $\sigma_{t,D}^2$ when a total of $R^{(s)}$ bits per element are used in the first t iterations. By definition of $\sigma_D^2(s, t)$, we have

$$\sigma_D^2(s, t) = \min_{r \in \{1, 2, \dots, s\}} f_1 \left(\sigma_D^2(r, t-1), R^{(s-r+1)} \right), \quad (11)$$

and the first column of elements in Σ is obtained by:

$$\sigma_D^2(s, 1) = f_1 \left(\sigma_0^2, R^{(s)} \right), \quad \forall s \in \{1, 2, \dots, S\}. \quad (12)$$

After obtaining Σ , the optimal value of $\sigma_{T,D}^2$, by definition, is $\sigma_D^2(S, T)$. Meanwhile, to obtain the optimal bit allocation strategy, we need another $S \times T$ array \mathbf{R} to store the optimal bit rate $R_{DP}(s, t)$ that is allocated at iteration t when a total of $R^{(s)}$ bits per element are used in the first t iterations. Similar to BT-MP-AMP, we name the proposed MP-AMP approach combined with DP as DP-MP-AMP.

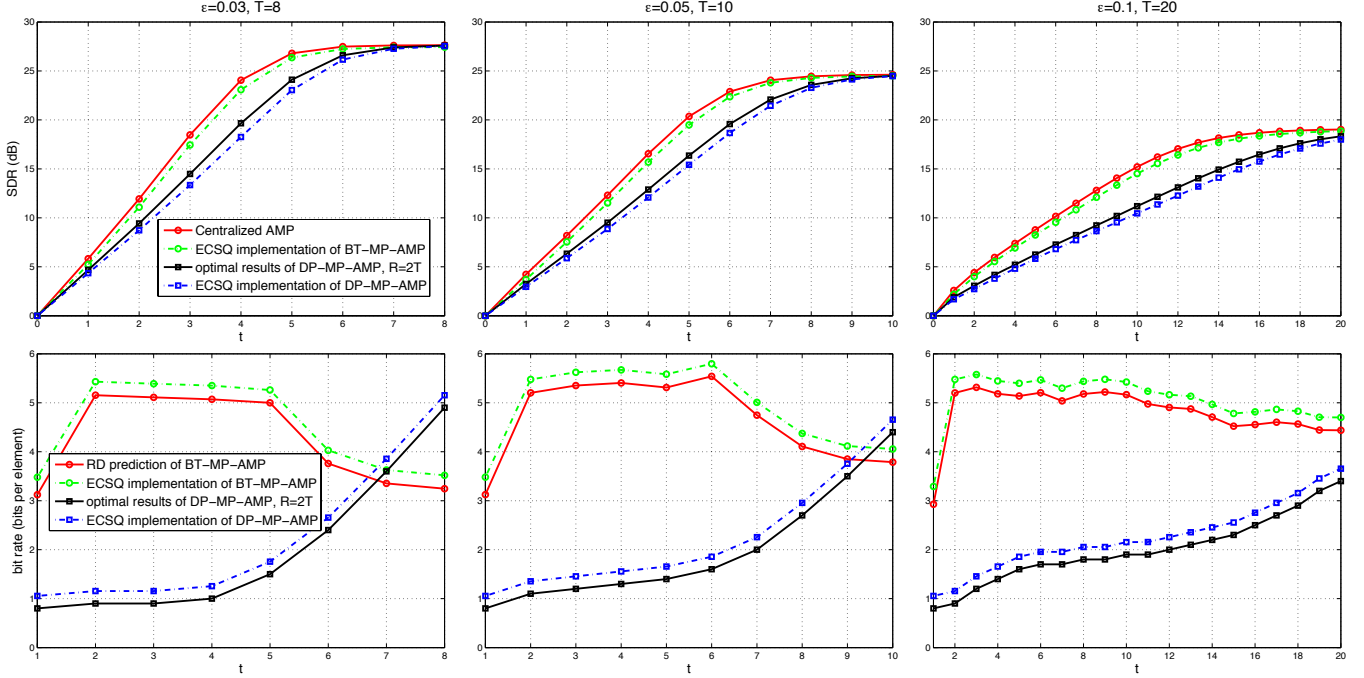


Fig. 1. SDR and bit rates as functions of iteration number t . ($N=10,000$, $M=3,000$, $\kappa=0.3$, $\mu_s=0$, $\sigma_s=1$, $\text{SNR}=20$ dB.)

Table 1. Total bits per element of MP-AMP

ϵ	0.03	0.05	0.10
T	8	10	20
BT-MP-AMP (RD prediction)	33.82	46.43	96.16
BT-MP-AMP (ECSQ simulation)	36.09	49.19	101.50
DP-MP-AMP (RD prediction)	16	20	40
DP-MP-AMP (ECSQ simulation)	18.04	22.55	45.10

4. NUMERICAL RESULTS

We evaluate BT-MP-AMP and DP-MP-AMP in an MP system with $P = 30$ processors at $\text{SNR} = 20$ dB, where we set $N = 10,000$, $M = 3,000$, i.e., $\kappa = 0.3$, and generate Bernoulli-Gaussian sequences \mathbf{s}_0 with $\epsilon \in \{0.03, 0.05, 0.1\}$, $\mu_s = 0$, and $\sigma_s = 1$.

We first evaluate the SE equation (4) of centralized AMP for the three sparsity levels. As shown in Fig. 1, they reach the steady state after $T = 8, 10$, and 20 iterations respectively. Then, we run BT-MP-AMP and DP-MP-AMP, where for the latter the total rates are $R = 2T$ bits per element and the RD-function models the relation between R_t and σ_Q^2 .

According to RD theory, in the high rate limit, we should expect a gap of roughly 0.255 bits per element between the entropy and RD function for a given distortion level [18]. Therefore, in an implementation of DP-MP-AMP where we apply ECSQ, we add 0.255 bits per element to the results in each iteration obtained by DP. Note that the two solid curves in the top three panels are obtained through offline calculation and optimization, and the two dash-dotted curves are obtained through AMP simulations.

As shown in Fig. 1, BT-MP-AMP uses fewer than 6 bits per element in each iteration, more than 80% communication savings compared with 32-bit single-precision floating-point transmission, while achieving almost the same SDR's as in centralized AMP. On the other hand, there are clear gaps between the SDR's of DP-MP-AMP and centralized AMP during the first few iterations, but they vanish quickly as t approaches T , in return for over 50% communication reduction beyond that provided by BT-MP-AMP, as shown in Table 1.

Note also that the ECSQ implementation of DP-MP-AMP has lower SDR's than that predicted by DP results based on the RD function at the beginning. This is because the 0.255-bits gap only holds in the high rate limit. However, due to the robustness of SE to disturbances, and the increasingly high rates as t approaches T , the ECSQ implementation matches the predicted DP results at the last iteration.

5. CONCLUSION

In this paper, we proposed a multi-processor approximate message passing framework with lossy compression. We used a uniform quantizer with entropy coding to reduce communication costs, and reformulated the state evolution formalism while accounting for quantization noise. Combining the quantizers and modified state evolution equation, an online back-tracking approach and another method based on dynamic programming determine the coding rate in each iteration by controlling the induced error. The numerical results suggest that our approaches can maintain a high signal-to-distortion-ratio despite a significant and often dramatic reduction in inter-processor communication costs.

6. REFERENCES

- [1] D. L. Donoho, "Compressed sensing," *IEEE Trans. Info. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.
- [2] E. J. Candes, J. K. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Comm. Pure Appl. Math.*, vol. 59, no. 8, pp. 1207–1223, Mar. 2006.
- [3] J. Mota, J. Xavier, P. Aguiar, and M. Puschel, "Distributed basis pursuit," *IEEE Trans. Sig. Proc.*, vol. 60, no. 4, pp. 1942–1956, Apr. 2012.
- [4] S. Patterson, Y. C. Eldar, and I. Keidar, "Distributed sparse signal recovery for sensor networks," in *IEEE Int. Conf. on Acoust., Speech, and Sig. Proc. (ICASSP)*, 2013, pp. 4494–4498.
- [5] S. Patterson, Y. C. Eldar, and I. Keidar, "Distributed compressed sensing for static and time-varying networks," *IEEE Trans. Sig. Proc.*, vol. 62, no. 19, pp. 4931–4946, Oct. 2014.
- [6] P. Han, R. Niu, M. Ren, and Y. C. Eldar, "Distributed approximate message passing for sparse signal recovery," in *IEEE Global Conf. Sig. Info. Proc. (GlobalSIP)*, 2014, pp. 497–501.
- [7] D. L. Donoho, A. Maleki, and A. Montanari, "Message passing algorithms for compressed sensing," in *Proc. Natl. Acad. Sci.*, Madrid, Spain, Sep. 2009, vol. 106, pp. 18914–18919.
- [8] D. L. Donoho, A. Maleki, and A. Montanari, "The Noise-Sensitivity Phase Transition in Compressed Sensing," *IEEE Trans. Info. Theory*, vol. 57, pp. 6920–6941, Oct. 2011.
- [9] M. Bayati and A. Montanari, "The Dynamics of Message Passing on Dense Graphs, with Applications to Compressed Sensing," *IEEE Trans. Info. Theory*, vol. 57, pp. 764–785, Feb. 2011.
- [10] J. Tan, Y. Ma, and D. Baron, "Compressive imaging via approximate message passing with image denoising," *IEEE Trans. Sig. Proc.*, vol. 63, no. 8, pp. 2085–2092, Apr. 2015.
- [11] J. T. Parker, P. Schniter, and V. Cevher, "Bilinear generalized approximate message passing," *arXiv preprint arXiv:1310.2632*, 2013.
- [12] A. Montanari and E. Richard, "Non-negative principal component analysis: Message passing algorithms and sharp asymptotics," *arXiv preprint arXiv:1406.4775*, 2014.
- [13] J. F. Mota, J. M. Xavier, P. M. Aguiar, and M. Puschel, "D-ADMM: A communication-efficient distributed algorithm for separable optimization," *IEEE Trans. Sig. Proc.*, vol. 61, no. 10, pp. 2718–2723, May 2013.
- [14] P. Han, R. Niu, and Y. C. Eldar, "Modified distributed iterative hard thresholding," in *IEEE Int. Conf. on Acoust., Speech, and Sig. Proc. (ICASSP)*, 2015, pp. 3766–3770.
- [15] P. Han, R. Niu, and Y. C. Eldar, "Communication-efficient distributed IHT," in *Proc. SPARS*, 2015.
- [16] D. Koller and N. Friedman, *Probabilistic graphical models: principles and techniques*, MIT Press, 2009.
- [17] J. Tan, Y. Ma, and D. Baron, "Compressive imaging via approximate message passing with wavelet-based image denoising," in *IEEE Global Conf. Sig. Info. Proc. (GlobalSIP)*, 2014, pp. 424–428.
- [18] A. Gersho and R. M. Gray, *Vector quantization and signal compression*, vol. 159, Springer Science & Business Media, 2012.
- [19] B. Widrow and I. Kollár, *Quantization Noise: Roundoff Error in Digital Computation, Signal Processing, Control, and Communications*, Cambridge University Press, 2008.
- [20] T. M. Cover and J. A. Thomas, *Elements of information theory*, John Wiley & Sons, 2012.
- [21] R. E. Blahut, "Computation of channel capacity and rate-distortion functions," *IEEE Trans. Info. Theory*, vol. 18, no. 4, pp. 460–473, Jul. 1972.
- [22] S. Arimoto, "An algorithm for computing the capacity of arbitrary discrete memoryless channels," *IEEE Trans. Info. Theory*, vol. 18, no. 1, pp. 14–20, Jan. 1972.