DISTRIBUTED APPROXIMATE MESSAGE PASSING WITH SUMMATION PROPAGATION

Ryo Hayakawa[⋆] *Ayano Nakai*[⋆] *Kazunori Hayashi*[†]

[⋆]Graduate School of Informatics, Kyoto University
[†]Graduate School of Engineering, Osaka City University

ABSTRACT

In this paper, we propose a fully distributed approximate message passing (AMP) algorithm, which reconstructs an unknown vector from its linear measurements obtained at nodes in a network. The proposed algorithm is a distributed implementation of the centralized AMP algorithm, and consists of the local computation at each node and the global computation using communications between nodes. For the global computation, we propose a distributed algorithm named summation propagation to calculate a summation required in the AMP algorithm. The proposed distributed AMP algorithm does not require any central node such as a fusion center, and can be realized only with locally available information at each node. Simulation results show that the proposed algorithm can achieve the same estimation accuracy as that of the centralized AMP algorithm.

Index Terms— Approximate message passing, consensus propagation, sensor network, compressed sensing

1. INTRODUCTION

As a distributed framework for compressed sensing [1,2], distributed compressed sensing has attracted much attention with several applications such as wireless sensor networks, video coding, and image fusion [3–5]. For the distributed compressed sensing, distributed least absolute shrinkage and selection operator (D-LASSO) [6] and distributed alternating direction method of multipliers (D-ADMM) [7] have been proposed. In each iteration of these algorithms, however, the nodes in the network need to solve an optimization problem, which may have high computational cost when the problem size is large. To reduce the local computation at each node, distributed iterative hard thresholding (D-IHT) [8,9] has been proposed. Although each node performs simple calculations such as additions and multiplications, the sparsity level of the unknown vector is assumed to be known in D-IHT. To address the problem, a distributed algorithm has been proposed in [10, 11] based on approximate message passing (AMP) algorithm [12, 13], which has been gathering attention due to its high computational efficiency and analytical tractability [14–17]. Although the distributed algorithm can reconstruct an unknown sparse vector without the knowledge of the sparsity level, it requires a unique node, which plays a role as a fusion center communicating with all nodes in the network.

In this paper, we propose a novel fully distributed AMP algorithm, which does not require any fusion center. The proposed algorithm is a fully distributed implementation of the centralized AMP algorithm to obtain the estimate of an unknown sparse vector at each node without sharing its measurements. The distributed AMP algorithm can be divided into the local computation at each node and the global computation using communications between nodes. In the global computation, each node needs to calculate a summation of vectors at all nodes to obtain the same estimate as that of the centralized AMP algorithm. To compute the summation, we propose a distributed algorithm named summation propagation, which can be derived by modifying consensus propagation [18] for the average consensus problem. The distributed AMP algorithm with summation propagation can provide the same estimate of the unknown sparse vector as that of the centralized AMP algorithm without sharing the measurements at all nodes. Moreover, since the centralized AMP algorithm can be extended to more general scenarios [15, 19, 20], the proposed approach can be applied for other reconstruction problems such as discrete-valued vector reconstruction. Simulation results show that the proposed algorithm can achieve the same performance as that of the centralized AMP algorithm.

In the rest of the paper, we use the following notations. We denote the vector whose elements are all 0 by $\mathbf{0}$. For a vector $\boldsymbol{v} = [v_1 \cdots v_N]^{\mathrm{T}} \in \mathbb{R}^N$, the sample mean of the elements of $\boldsymbol{v}$ is given by $\langle \boldsymbol{v} \rangle = \frac{1}{N}\sum_{n=1}^{N} v_n$. We represent the sign function by $\mathrm{sgn}(\cdot)$. $\phi(z) = \frac{1}{\sqrt{2\pi}} \exp\left(-z^2/2\right)$ and $\Phi(z) = \int_{-\infty}^{z} \phi(z')\mathrm{d}z'$ are the probability density function and the cumulative distribution function of the standard Gaussian distribution, respectively.

2. PRELIMINARIES

2.1. AMP Algorithm

AMP algorithm [12, 13] has been originally proposed for compressed sensing, where the goal is to reconstruct a sparse vector $\boldsymbol{x} = [x_1 \cdots x_N]^{\mathrm{T}} \in \mathbb{R}^N$ from its underdetermined

This work was supported in part by the Grants-in-Aid for Scientific Research no. 15K06064 and 15H02252 from MEXT, and the Grant-in-Aid for JSPS Research Fellow no. 17J07055 from JSPS.

978-1-5386-4658-8/18/$31.00 ©2018 IEEE

Algorithm 1 AMP Algorithm

```
Input: $\boldsymbol{y} \in \mathbb{R}^M, \boldsymbol{A} \in \mathbb{R}^{M \times N}$
Output: $\hat{\boldsymbol{x}} \in \mathbb{R}^N$
$\hat{\boldsymbol{x}}(1) = \boldsymbol{0}, \boldsymbol{s}(0) = \boldsymbol{0}, \boldsymbol{r}(0) = \boldsymbol{0}, \hat{\sigma}^2(0) = 0, \Delta = M/N$
for $t = 1$ to $T$ do
    $\boldsymbol{s}(t) = \boldsymbol{y} - \boldsymbol{A}\hat{\boldsymbol{x}}(t) + \frac{1}{\Delta}\boldsymbol{s}(t-1) \cdot \left\langle \eta' \left( \boldsymbol{r}(t-1); \hat{\sigma}^2(t-1) \right) \right\rangle$
    $\boldsymbol{r}(t) = \hat{\boldsymbol{x}}(t) + \frac{1}{M}\boldsymbol{A}^{\mathrm{T}}\boldsymbol{s}(t)$
    $\hat{\sigma}^2(t) = \frac{\|\boldsymbol{s}(t)\|_2^2}{MN}$
    $\hat{\boldsymbol{x}}(t+1) = \eta\left( \boldsymbol{r}(t); \hat{\sigma}^2(t) \right)$
end for
$\hat{\boldsymbol{x}} = \hat{\boldsymbol{x}}(T+1)$
```

linear measurements $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{v} \in \mathbb{R}^M$ ($M < N$). The elements of $\boldsymbol{x}$ are assumed to be independent and identically distributed (i.i.d.) random variables. We also assume that the measurement matrix $\boldsymbol{A} \in \mathbb{R}^{M \times N}$ is composed of i.i.d. random variables with zero mean and unit variance, while the variance is usually assumed to be $1/M$ in the literature. $\boldsymbol{v} \in \mathbb{R}^M$ is the additive noise vector whose elements are i.i.d. Gaussian variables with zero mean and variance σ_v^2.

Algorithm 1 shows the AMP algorithm, where $\hat{\boldsymbol{x}}(t)$ denotes the estimate of the unknown vector $\boldsymbol{x}$ at the tth iteration. One of possible candidates of the function $\eta(\cdot)$ is the soft thresholding function given by

$$\left[\eta\left(\boldsymbol{r}; \sigma^2\right)\right]_n = \operatorname{sgn}\left(r_n\right) \max\left(|r_n| - \tau\frac{\sigma}{\sqrt{\Delta}}, 0 \right), \tag{1}$$

where $\left[\eta\left(\boldsymbol{r}; \sigma^2\right)\right]_n$ and r_n denotes the nth element of $\eta\left(\boldsymbol{r}; \sigma^2\right)$ and $\boldsymbol{r}$, respectively. The nth element of $\eta'\left(\boldsymbol{r}; \sigma^2\right)$ is the partial derivative of $\eta\left(\boldsymbol{r}; \sigma^2\right)$ with respect to r_n. τ (≥ 0) is the parameter and $\hat{\sigma}^2(t)$ is the estimate of the mean-square-error (MSE) $\sigma^2(t) = \frac{1}{N}\|\boldsymbol{x} - \hat{\boldsymbol{x}}(t)\|_2^2$ of $\hat{\boldsymbol{x}}(t)$ [16].

The AMP algorithm can be applied not only for the sparse vector reconstruction but also for the reconstruction of a discrete-valued vector as $\boldsymbol{x} \in \{b_1, \ldots, b_L\}^N$ [19,20]. For example, by using the function

$$\left[\eta\left(\boldsymbol{r}; \sigma^2\right)\right]_n = \frac{\sum_{\ell=1}^{L} p_\ell b_\ell \phi\left(\frac{\sqrt{\Delta}}{\sigma}\left(r_n - b_\ell\right) \right)}{\sum_{\ell'=1}^{L} p_{\ell'} \phi\left(\frac{\sqrt{\Delta}}{\sigma}\left(r_n - b_{\ell'}\right) \right)} \tag{2}$$

instead of the soft thresholding function (1), we can reconstruct the discrete-valued vector from its underdetermined linear measurements, where $p_\ell = \Pr(x_n = b_\ell)$ ($\ell = 1, \ldots, L$) and $\sum_{\ell=1}^{L} p_\ell = 1$. Generalized AMP (GAMP) [15] algorithm has also been proposed for more general scenario of the vector reconstruction problem.

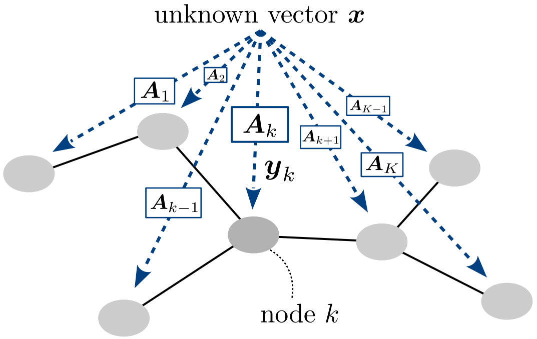

Fig. 1. System model

2.2. Consensus Propagation

Consensus propagation [18] is a distributed protocol to achieve the average consensus on an undirected graph G composed of K nodes. Specifically, node k ($k = 1, \ldots, K$) has an initial value $c_k \in \mathbb{R}$, and the goal is that each node obtains the average $\mu = \frac{1}{K}\sum_{k=1}^{K} c_k$ by the local computations and communications. Starting from the initialization $\nu_{k \to j}^{(0)} = 0$ and $\iota_{k \to j}^{(0)} = 0$ for each node $j \in \mathcal{N}_k$, node k updates these variables as

$$\nu_{k \to j}^{(t')} = \frac{c_k + \sum_{i \in \mathcal{N}_k \setminus j} \iota_{i \to k}^{(t'-1)} \nu_{i \to k}^{(t'-1)}}{1 + \sum_{i \in \mathcal{N}_k \setminus j} \iota_{i \to k}^{(t'-1)}}, \tag{3}$$

$$\iota_{k \to j}^{(t')} = 1 + \sum_{i \in \mathcal{N}_k \setminus j} \iota_{i \to k}^{(t'-1)}, \tag{4}$$

where $\mathcal{N}_k$ denotes the set of neighbor nodes of node k. After T' iterations, the estimate of the average μ at node k is obtained as

$$\hat{\mu}_k = \frac{c_k + \sum_{i \in \mathcal{N}_k} \iota_{i \to k}^{(T')} \nu_{i \to k}^{(T')}}{1 + \sum_{i \in \mathcal{N}_k} \iota_{i \to k}^{(T')}}. \tag{5}$$

Note that if the graph G is a tree and T' is greater than or equal to the diameter of the graph, the estimates $\hat{\mu}_k$ ($k = 1, \ldots, K$) are exactly equal to μ.

3. SYSTEM MODEL

Figure 1 shows the system model considered in this paper. We consider an undirected graph G composed of K nodes, which can communicate only with their neighbor nodes. In this paper, we assume that the graph G is a tree. For general graphs including loops, we can extract a spanning tree from the original graph [21, 22] and consider the message passing on the tree. Each node k ($k = 1, \ldots, K$) observes an unknown vector $\boldsymbol{x}$ as $\boldsymbol{y}_k = \boldsymbol{A}_k\boldsymbol{x} + \boldsymbol{v}_k \in \mathbb{R}^{M_k}$, where $\boldsymbol{A}_k \in \mathbb{R}^{M_k \times N}$ and $\boldsymbol{v}_k \in \mathbb{R}^{M_k}$ are the measurement matrix and the additive noise vector at node k, respectively. All measurements $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_K$ can be combined as

$$\begin{bmatrix} \boldsymbol{y}_1 \\ \vdots \\ \boldsymbol{y}_K \end{bmatrix} = \begin{bmatrix} \boldsymbol{A}_1 \\ \vdots \\ \boldsymbol{A}_K \end{bmatrix} \boldsymbol{x} + \begin{bmatrix} \boldsymbol{v}_1 \\ \vdots \\ \boldsymbol{v}_K \end{bmatrix}. \tag{6}$$

Algorithm 2 Summation Propagation

```
Input: $c_k \in \mathbb{R}$ $(k = 1, \dots, K)$
Output: $\hat{C}_k \in \mathbb{R}$ $(k = 1, \dots, K)$
$\xi_{k \to j}^{(0)} = 0$ ($k = 1, \dots, K$ and $j \in \mathcal{N}_k$)
for $t' = 1$ to $T'$ do
    $\xi_{k \to j}^{(t')} = c_k + \sum_{i \in \mathcal{N}_k \setminus j} \xi_{i \to k}^{(t'-1)}$
    ($k = 1, \dots, K$ and $j \in \mathcal{N}_k$)
end for
$\hat{C}_k = c_k + \sum_{i \in \mathcal{N}_k} \xi_{i \to k}^{(T')}$ $(k = 1, \dots, K)$
```

4. PROPOSED DISTRIBUTED AMP ALGORITHM

Our goal is to reconstruct the unknown vector $\boldsymbol{x}$ in a distributed manner without sharing the measurements $\boldsymbol{y}_k$ and the measurement matrices $\boldsymbol{A}_k$ at all nodes. The update equations of the centralized AMP algorithm for (6) can be written as

$$\begin{aligned} \boldsymbol{s}_k(t) &= \boldsymbol{y}_k - \boldsymbol{A}_k \hat{\boldsymbol{x}}(t) \\ &\quad + \frac{1}{\Delta} \boldsymbol{s}_k(t-1) \left\langle \eta' \left(\boldsymbol{r}(t-1); \hat{\sigma}^2(t-1) \right) \right\rangle \\ &\quad (k = 1, \dots, K), \end{aligned} \tag{7}$$

$$\boldsymbol{r}(t) = \sum_{k=1}^{K} \left(\frac{1}{K} \hat{\boldsymbol{x}}(t) + \frac{1}{M} \boldsymbol{A}_k^{\mathrm{T}} \boldsymbol{s}_k(t) \right), \tag{8}$$

$$\hat{\sigma}^2(t) = \sum_{k=1}^{K} \frac{\| \boldsymbol{s}_k(t) \|_2^2}{MN}, \tag{9}$$

$$\hat{\boldsymbol{x}}(t+1) = \eta \left(\boldsymbol{r}(t); \hat{\sigma}^2(t) \right), \tag{10}$$

where $M = \sum_{k=1}^{K} M_k$. Since the summation in (8) and (9) involves all $\boldsymbol{A}_k$ and $\boldsymbol{s}_k(t)$, they cannot be computed locally at each node, while, once $\boldsymbol{r}(t)$ and $\hat{\sigma}^2(t)$ are obtained, the nodes can locally compute (7) and (10).

To obtain $\boldsymbol{r}(t)$ and $\hat{\sigma}^2(t)$ in a distributed manner, we propose summation propagation in Algorithm 2 by modifying consensus propagation in Section 2.2. The numerator and the denominator of (5) can be regarded as the estimates of the summation $\sum_{k=1}^{K} c_k$ and the number of nodes K, respectively. We thus derive update equations to obtain the numerator $c_k + \sum_{i \in \mathcal{N}_k} \xi_{i \to k}^{(T')}$, where we define $\xi_{i \to k}^{(t')} := \iota_{i \to k}^{(t')} \nu_{i \to k}^{(t')}$. By multiplying the both sides of (3) and (4), the update equation of $\xi_{\cdot \to \cdot}^{(t')}$ can be obtained as

$$\xi_{k \to j}^{(t')} = c_k + \sum_{i \in \mathcal{N}_k \setminus j} \xi_{i \to k}^{(t'-1)}. \tag{11}$$

After T' iterations of (11), the estimate of the summation is given by $\hat{C}_k := c_k + \sum_{i \in \mathcal{N}_k} \xi_{i \to k}^{(T')}$. For the graph G with the tree structure, $\hat{C}_k = \sum_{k'=1}^{K} c_{k'}$ holds for any k if T' is greater than or equal to the diameter of G.

In Algorithm 3, we summarize the proposed distributed AMP algorithm with summation propagation. Each node locally calculates (7) and (10), and globally compute (8) and (9) via summation propagation. Although Algorithm 3 uses the number of nodes K and the number of all measurements M as inputs, they can also be obtained with summation propagation in advance because $K = \sum_{k=1}^{K} 1$ and $M = \sum_{k=1}^{K} M_k$. Note that the proposed algorithm does not require any fusion center unlike the AMP-based algorithms in [10, 11]. Moreover, we can extend the proposed approach for the discrete-valued vector reconstruction using (2) and the GAMP algorithm.

Algorithm 3 Distributed AMP Algorithm with Summation Propagation

```
Input: $\boldsymbol{y}_k \in \mathbb{R}^{M_k}, \boldsymbol{A}_k \in \mathbb{R}^{M_k \times N}$ $(k = 1, \dots, K)$, $K$,
       $M = \sum_{k=1}^{K} M_k$
Output: $\hat{\boldsymbol{x}}_k \in \mathbb{R}^N$ $(k = 1, \dots, K)$
$\hat{\boldsymbol{x}}_k(1) = \boldsymbol{0}, \boldsymbol{s}_k(0) = \boldsymbol{0}, \boldsymbol{r}_k(0) = \boldsymbol{0}, \tilde{\sigma}_k^2(0) = 0$ $(k = 1, \dots, K)$, $\Delta = M/N$
for $t = 1$ to $T$ do
    local computation $(k = 1, \dots, K)$:
    $\boldsymbol{s}_k(t) = \boldsymbol{y}_k - \boldsymbol{A}_k \hat{\boldsymbol{x}}_k(t) + \frac{1}{\Delta} \boldsymbol{s}_k(t-1)$
        $\cdot \left\langle \eta' \left( \boldsymbol{r}_k(t-1); \tilde{\sigma}_k^2(t-1) \right) \right\rangle$
    $\boldsymbol{r}_k(t) = \frac{1}{K} \hat{\boldsymbol{x}}_k(t) + \frac{1}{M} \boldsymbol{A}_k^{\mathrm{T}} \boldsymbol{s}_k(t)$
    $\hat{\sigma}_k^2(t) = \frac{\| \boldsymbol{s}_k(t) \|_2^2}{MN}$
    global computation via summation propagation
    ($k = 1, \dots, K$ and $j \in \mathcal{N}_k$):
    $\boldsymbol{\theta}_{k \to j}^{(0)} = \boldsymbol{0}$, $\gamma_{k \to j}^{(0)} = 0$
    for $t' = 1$ to $T'$ do
        $\boldsymbol{\theta}_{k \to j}^{(t')} = \boldsymbol{r}_k(t) + \sum_{i \in \mathcal{N}_k \setminus j} \boldsymbol{\theta}_{i \to k}^{(t'-1)}$
        $\gamma_{k \to j}^{(t')} = \hat{\sigma}_k^2(t) + \sum_{i \in \mathcal{N}_k \setminus j} \gamma_{i \to k}^{(t'-1)}$
    end for
    $\tilde{\boldsymbol{r}}_k(t) = \boldsymbol{r}_k(t) + \sum_{i \in \mathcal{N}_k} \boldsymbol{\theta}_{i \to k}^{(T')}$
    $\tilde{\sigma}_k^2(t) = \hat{\sigma}_k^2(t) + \sum_{i \in \mathcal{N}_k} \gamma_{i \to k}^{(T')}$
    local computation $(k = 1, \dots, K)$:
    $\hat{\boldsymbol{x}}_k(t+1) = \eta \left( \tilde{\boldsymbol{r}}_k(t); \tilde{\sigma}_k^2(t) \right)$
end for
$\hat{\boldsymbol{x}}_k = \hat{\boldsymbol{x}}_k(T+1)$ $(k = 1, \dots, K)$
```

5. SIMULATION RESULTS

In this section, we evaluate the performance of the distributed AMP algorithm via computer simulations. In the simulations, we have generated a tree graph composed of $K = 50$ nodes. The diameter of the graph is 6.

5.1. Sparse Vector Reconstruction

We firstly show the performance of the distributed AMP algorithm for the sparse vector reconstruction. The probability density function of the element x_n of $\boldsymbol{x}$ is $p(x_n) = q\delta(x_n) + (1-q)\phi(x_n)$, where $\delta(\cdot)$ denotes the delta function. The pa-

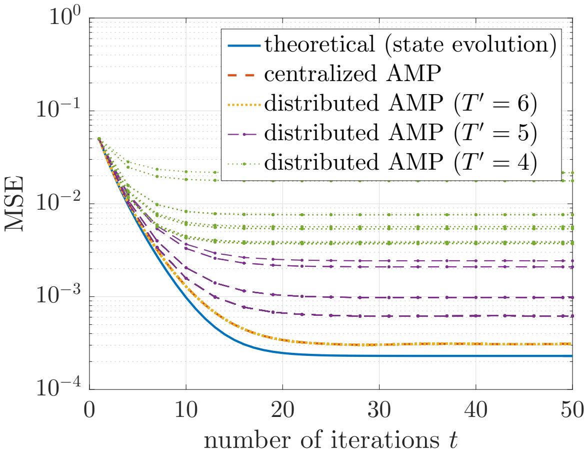

Fig. 2. MSE for sparse vector reconstruction

rameter $q \in [0, 1]$, which determines the sparsity level of $\boldsymbol{x}$, is assumed to be unknown. In the algorithm, we use the soft thresholding function (1) and the parameter $\tau = \hat{\tau}$ given by

$$\hat{\tau} = \arg\max_{\tau \geq 0} \frac{\Delta + 2\{\tau\phi(\tau) - (1 + \tau^2)\Phi(-\tau)\}}{(1 + \tau^2) + 2\{\tau\phi(\tau) - (1 + \tau^2)\Phi(-\tau)\}} \quad (12)$$

as in [12].

Figure 2 shows the average MSE obtained by the computer simulations. We set $N = 1000$, $M_k = 6$ ($k = 1, \ldots, K$), $q = 0.95$ and $\sigma_v^2 = 0.1$. We plot the 50 curves corresponding to each node as "distributed AMP". For comparison, we also plot the performance of the centralized AMP algorithm given in Algorithm 1 as "centralized AMP". "theoretical (state evolution)" denotes the asymptotic performance of the AMP algorithm in the large system limit ($M, N \to \infty$ with fixed $M/N = \Delta$), which is theoretically obtained via state evolution [12, 14]. When $T' = 6$, all nodes achieve the same MSE as that of the centralized AMP algorithm and their performance are close to the theoretical prediction from the state evolution. When $T' = 5$ and $T' = 4$, however, the performance degrades and the MSE curves of the nodes are not identical because the consensus in the global computation cannot be achieved.

In Fig. 3, we show the average MSE versus q when $N = 1000$, $M_k = 6$ ($k = 1, \ldots, K$), $\sigma_v^2 = 0.1$, and $T = 50$. As in Fig. 2, we can see that the distributed AMP algorithm achieves the same performance as that of the centralized AMP algorithm when $T' = 6$.

5.2. Binary Vector Reconstruction

Next, we evaluate the performance for the reconstruction of binary vector $\boldsymbol{x} \in \{0, 1\}^N$. Sparse event detection [23] in wireless sensor networks can be reduced to such a binary vector reconstruction problem. We assume that the probabilities $p_1 = \Pr(x_n = 0)$ and $p_2 = \Pr(x_n = 1)$ are known at each node, and use the function $\eta(\cdot)$ given by (2) in the algorithm.

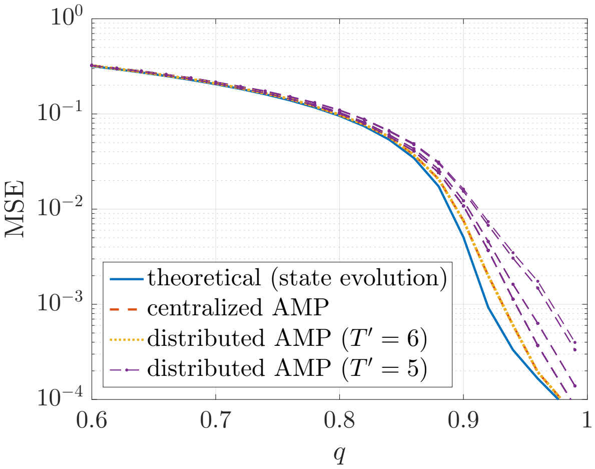

Fig. 3. MSE versus q for sparse vector reconstruction

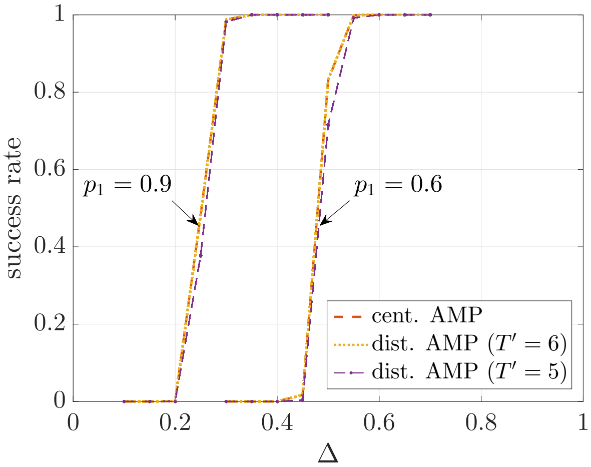

Fig. 4. Success rate for discrete-valued vector reconstruction

Figure 4 shows the success rate of the proposed algorithm for $p_1 = 0.9$ and $p_1 = 0.6$. In the simulation, we set $N = 1000$, $M_1 = \cdots = M_K = \Delta N/K, \sigma_v^2 = 1$, and $T = 50$. We can see that the performance of the proposed algorithm with $T' = 6$ corresponds to that of the centralized AMP algorithm. Moreover, the distributed AMP algorithm achieves the comparable performance even when $T' = 5$.

6. CONCLUSION

In this paper, we have proposed the fully distributed AMP algorithm using the idea of consensus propagation. The proposed algorithm does not require any fusion center to reconstruct the unknown vector. Simulation results show that the proposed algorithm can achieve the same performance as that of the centralized AMP algorithm for both the sparse vector reconstruction and the binary vector reconstruction.

Future work includes the extension of distributed AMP for general graphs and the reduction of the number of communications between nodes.

7. REFERENCES

[1] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.

[2] K. Hayashi, M. Nagahara, and T. Tanaka, "A user's guide to compressed sensing for communications systems," *IEICE Trans. Commun.*, vol. E96-B, no. 3, pp. 685–712, Mar. 2013.

[3] M. F. Duarte, S. Sarvotham, D. Baron, M. B. Wakin, and R. G. Baraniuk, "Distributed compressed sensing of jointly sparse signals," in *Proc. Asilomar Conf. Signals, Syst., Comput.*, Oct. 2005.

[4] D. Baron, M. F. Duarte, M. B. Wakin, S. Sarvotham, and R. G. Baraniuk, "Distributed compressive sensing," 2009. [Online]. Available: http://arxiv.org/abs/0901.3403

[5] H. Yin, J. Li, Y. Chai, and S. X. Yang, "A survey on distributed compressed sensing: theory and applications," *Frontiers of Computer Science*, vol. 8, no. 6, pp. 893–904, Dec. 2014.

[6] J. A. Bazerque and G. B. Giannakis, "Distributed spectrum sensing for cognitive radio networks by exploiting sparsity," *IEEE Trans. Signal Process.*, vol. 58, no. 3, pp. 1847–1862, Mar. 2010.

[7] J. F. C. Mota, J. M. F. Xavier, P. M. Q. Aguiar, and M. Püschel, "Distributed basis pursuit," *IEEE Trans. Signal Process.*, vol. 60, no. 4, pp. 1942–1956, Apr. 2012.

[8] S. Patterson, Y. C. Eldar, and I. Keidar, "Distributed sparse signal recovery for sensor networks," in *Proc. IEEE ICASSP*, May 2013.

[9] P. Han, R. Niu, and Y. C. Eldar, "Modified distributed iterative hard thresholding," in *Proc. IEEE ICASSP*, Apr. 2015.

[10] P. Han, R. Niu, M. Ren, and Y. C. Eldar, "Distributed approximate message passing for sparse signal recovery," in *Proc. IEEE GlobalSIP*, Dec. 2014.

[11] J. Zhu, R. Pilgrim, and D. Baron, "An overview of multiprocessor approximate message passing," in *Proc. IEEE CISS*, Mar. 2017.

[12] D. L. Donoho, A. Maleki, and A. Montanari, "Message-passing algorithms for compressed sensing," *Proc. Nat. Acad. Sci.*, vol. 106, no. 45, pp. 18 914–18 919, Nov. 2009.

[13] ——, "Message passing algorithms for compressed sensing: I. motivation and construction," in *Proc. IEEE Inf. Theory Workshop*, Jan. 2010.

[14] 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.

[15] S. Rangan, "Generalized approximate message passing for estimation with random linear mixing," in *Proc. IEEE ISIT*, Jul./Aug. 2011.

[16] D. L. Donoho, A. Maleki, and A. Montanari, "The noise-sensitivity phase transition in compressed sensing," *IEEE Trans. Inf. Theory*, vol. 57, no. 10, pp. 6920–6941, Oct. 2011.

[17] D. L. Donoho, A. Javanmard, and A. Montanari, "Information-theoretically optimal compressed sensing via spatial coupling and approximate message passing," *IEEE Trans. Inf. Theory*, vol. 59, no. 11, pp. 7434–7464, Nov. 2013.

[18] C. C. Moallemi and B. V. Roy, "Consensus propagation," *IEEE Trans. Inf. Theory*, vol. 52, no. 11, pp. 4753–4766, Nov. 2006.

[19] R. Hayakawa and K. Hayashi, "Discreteness-aware AMP for reconstruction of symmetrically distributed discrete variables," in *Proc. IEEE SPAWC*, Jul. 2017.

[20] ——, "Binary vector reconstruction via discreteness-aware approximate message passing," in *Proc. APSIPA ASC*, Dec. 2017.

[21] M. Bui, F. Butelle, and C. Lavault, "A distributed algorithm for constructing a minimum diameter spanning tree," *J. Parallel and Distributed Computing*, vol. 64, no. 5, pp. 571–577, May 2004.

[22] J. Herzen, C. Westphal, and P. Thiran, "Scalable routing easy as PIE: A practical isometric embedding protocol," in *Proc. IEEE ICNP*, Oct. 2011.

[23] J. Meng, H. Li, and Z. Han, "Sparse event detection in wireless sensor networks using compressive sensing," in *Proc. IEEE CISS*, Mar. 2009.