

# Approximate Message Passing Algorithm for Complex Separable Compressed Imaging

Akira Hirabayashi\*, Jumpei Sugimoto†, and Kazushi Mimura‡

\* College of Information Science and Engineering, Ritsumeikan University, Kusatsu Shiga 525-8577 Japan.

E-mail: akirahrb@media.ritsumei.ac.jp

† Graduate School of Medicine, Yamaguchi University, Yamaguchi 755-8611 Japan.

‡ Graduate School of Information Sciences, Hiroshima City University, Hiroshima 731-3194 Japan.

E-mail: mimura@hiroshima-cu.ac.jp

**Abstract**—We propose the approximate message passing (AMP) algorithm for complex separable compressed imaging. The standard formulation of compressed sensing uses one-dimensional signals while images are usually reshaped into such vectors by raster scan, which requires a huge matrix. In separable cases like discrete Fourier transform (DFT), however, sensing processes can be formulated using two moderate size matrices which are multiplied to images from the both sides. We exploit this formulation in our AMP algorithm. Since we suppose DFT for the sensing process, in which measurements are complex, our formulation applies to cases in which both target signals and measurements are complex. We show that the proposed algorithm perfectly reconstructs a  $128 \times 128$  image, which could not be handled by the raster scan approach on the same computational environment. We also show that the compression rate of the proposed algorithm is mostly same as the so-called weak threshold.

## I. INTRODUCTION

Compressed sensing is a technique to recover a sparse vector with  $N$  elements from an undersampled set of random linear measurements using a sensing matrix [1] [2]. One popular class of reconstruction schemes is linear programming [3]. For instance, the interior point method [4] requires the computational cost of  $O(N^3)$  to solve an  $\ell_1$  norm minimization problem with linear constraints. To reduce the cost, various iterative algorithms have been proposed so far [3]. Above all, the approximate message passing (AMP) algorithm has been paid attention as a low cost one that can attain the same performance with the  $\ell_1$  minimization [5]. For a fixed number of iterations, the computational cost of AMP with a dense sensing matrix generally becomes  $O(N^2)$ . Certain sparse sensing matrices further reduce the cost to  $O(N)$ . Hence, methods to construct such matrices have been widely studied [6].

This paper also focuses on the AMP, which was originally proposed for one-dimensional sparse vectors. In image processing applications, however, target signals are given naturally in the two-dimensional form. They can be treated in one-dimensional form by the raster scan and then the corresponding sensing matrix, which is formed by the Kronecker product, gets the size of  $O(N^2)$ . This treatment is necessary for general sensing processes. However, typical sensing processes for images, such as the two-dimensional Fourier transform and the two-dimensional wavelet transform,

are separable, and we can exploit a concise formulation of  $O(N)$ , where two matrices are applied to images from both sides. For the sensing matrices formed by the Kronecker product, the restricted isometry constant, spark, incoherence, null space property, and empirical performance have been discussed [7], [8]. To the best of our knowledge, however, the AMP algorithm for the separable sensing formulation has not yet been proposed so far. In addition, the sensing formulation should be able to take complex values because the discrete Fourier transform is supposed, as is in [9] for one-dimensional case. In this paper, we propose an AMP algorithm to recover two-dimensional complex-valued signals with complex-valued separable sensing formulation. This approach can be regarded as a combination of the Kronecker compressed sensing [8] and the complex-valued one-dimensional AMP [9]. We show by simulations that the proposed algorithm effectively reduces the memory and computational costs and achieves the compression rate as same as the so-called weak threshold [5].

This paper is organized as follows. The next section briefly reviews AMP for the one-dimensional real-valued case. Section III proposes the complex-valued AMP for the separable imaging formulation. Section IV is devoted to simulations. Section V concludes the paper.

## II. ONE-DIMENSIONAL SIGNAL RECOVERY

The real-valued one-dimensional compressed sensing problem is to recover an unknown sparse signal  $\mathbf{x} \in \mathbb{R}^N$  from a given sensing matrix  $A \in \mathbb{R}^{M \times N}$  ( $M < N$ ) and a given undersampled set of random linear measurements

$$\mathbf{y} = A\mathbf{x}_0 \in \mathbb{R}^M. \quad (1)$$

In general, the solution is indeterminate, since the number of the constraints is less than that of the unknown variables. We therefore call  $\alpha = M/N$  the compression rate. If we know that the signal  $\mathbf{x}_0$  is  $K$ -sparse, that is  $\mathbf{x}_0$  has  $K$  non-zero elements at most, then there are cases in which we can recover the vector by applying this prior knowledge. The ratio  $\rho := K/N$  is referred to as the signal density or the sparsity. As a simple way for sparse recovery, it can be considered to infer an estimate  $\mathbf{x} \in \mathbb{R}^N$  that gives a minimum number of non-zero elements, namely a minimum  $\ell_0$  norm, in the set of the vectors that satisfy the constraint  $\mathbf{y} = A\mathbf{x}$ . Since this method,

however, becomes a combinatorial optimization problem with respect to a support set of the unknown vector, it is impractical to perform.

Instead, a method that is relaxed to the following  $\ell_1$  norm minimization has been widely discussed.

$$\mathbf{x} = \operatorname{argmin}_{\mathbf{s} \in \mathbb{R}^N} \|\mathbf{s}\|_1 \text{ s.t. } \mathbf{y} = A\mathbf{s}. \quad (2)$$

The computational cost of this optimization problem is  $O(N^3)$  in the case where the interior point method is applied. To reduce the computational cost, various algorithms based on the iterative method have been proposed to solve this  $\ell_1$  optimization problem [3]. In particular, we now focus on AMP [5] that is summarized as follows.

Let  $A^\top$  denote transpose of  $A$  and  $\mathbb{I}(P)$  stand for the indicator function that takes one if the proposition  $P$  is true, or zero otherwise. The sign function, denoted by  $\operatorname{sgn}(x)$ , returns 1 if  $x > 0$ , 0 if  $x = 0$ , or -1 otherwise.

*Algorithm 1 (AMP, Ref.[5]):* Let  $\eta_t : \mathbb{R} \rightarrow \mathbb{R}$  be a soft threshold function that is applied componentwisely, as

$$\eta_t(x) := \left( x - \theta_t \operatorname{sgn}(x) \right) \mathbb{I}(|x| > \theta_t)$$

with  $\theta_t$  a threshold value proportional to an (empirical) mean squared error (MSE). Starting from  $\mathbf{x}^0 = \mathbf{0}$  and  $\mathbf{z}^0 = \mathbf{y}$ , the AMP algorithm proceeds iteratively according to

$$\mathbf{x}^{t+1} = \eta_t(A^\top \mathbf{z}^t + \mathbf{x}^t), \quad (3)$$

$$\mathbf{z}^t = \mathbf{y} - A\mathbf{x}^t + \frac{1}{\alpha} \mathbf{z}^{t-1} \langle \eta'_{t-1}(A^\top \mathbf{z}^{t-1} + \mathbf{x}^{t-1}) \rangle, \quad (4)$$

where  $\eta'_t(x)$  stands for a derivative of  $\eta(x)$  with respect to  $x$ , and  $\langle \cdot \rangle$  denotes an average of vector entries, i.e.,  $\langle \mathbf{a} \rangle := p^{-1} \sum_{i=1}^p a_i$  for  $p$ -dimensional vector  $\mathbf{a} = (a_i)$ .

When a given appropriate termination condition is satisfied, this algorithm stops and outputs the latest estimate  $\mathbf{x}^t$ . ■

The variable  $\mathbf{x}^t \in \mathbb{R}^N$  is the current estimate of  $\mathbf{x}_0$ , while  $\mathbf{z}^t \in \mathbb{R}^M$  is the current residual. For a fixed number of iterations and a fixed compression rate, the computational cost of AMP with a dense sensing matrix becomes  $O(N^2)$ . It is lesser than that of the interior point method. It is known that there is the sparsity-undersampling tradeoff [5].

### III. TWO-DIMENSIONAL SIGNAL RECOVERY

We consider the complex-valued two-dimensional compressed sensing problem that is to recover an unknown complex sparse image  $X_0 \in \mathbb{C}^{N_1 \times N_2}$  from a given complex sensing matrices  $A \in \mathbb{C}^{M_1 \times N_1}$ ,  $B \in \mathbb{C}^{N_2 \times M_2}$  and a given complex undersampled set of random linear measurements

$$Y = AX_0B \in \mathbb{C}^{M_1 \times M_2}, \quad (5)$$

where the sizes of the image and measurement are  $N := N_1N_2$  and  $M := M_1M_2$ , respectively. The compression rate is again  $\alpha := M/N$ . It is possible to employ one-dimensional compressed sensing schemes to this separable sensing formulation [10]. However, when the sizes of the sensing matrices  $A$  and  $B$  are proportional to that of the signal

$N$ , the size of the corresponding sensing matrix denoted by the Kronecker product  $A \otimes B$  becomes  $O(N^2)$  for a fixed compression rate. On the other hand, when we directly exploit (5), the total size of the sensing matrices is  $O(N)$ , and we can easily apply AMP to large-scale images.

We now consider the following  $\ell_1$  optimization problem:

$$X = \operatorname{argmin}_{S \in \mathbb{C}^{N_1 \times N_2}} \|S\|_1 \text{ s.t. } Y = ASB, \quad (6)$$

where the matrix norm is defined by  $\|U\|_1 := \sum_{i=1}^p \sum_{j=1}^q |u_{ij}|$  for  $U \in \mathbb{C}^{p \times q}$ . According to the derivation of the original AMP [5] and the belief propagation [11], we have the following complex-valued two-dimensional AMP algorithm by means of the maximization of the posterior marginals. We use the notation  $x^R$  and  $x^I$  to represent the real and imaginary parts of a complex value, matrix, and function  $x$ . The notation  $A^*$  denotes complex conjugate transpose of  $A$ .

*Algorithm 2 (2D-CAMP):* Let  $\eta_t : \mathbb{C} \rightarrow \mathbb{C}$  be a soft threshold function that is applied componentwisely, as

$$\eta_t(x) = \left( x - \theta_t \frac{x}{|x|} \right) \mathbb{I}(|x| > \theta_t) \in \mathbb{C}, \quad (7)$$

where  $\theta_t \in \mathbb{R}$  is a threshold value proportional to an (empirical) MSE. Putting  $\eta_t(x) = \eta_t^R(x^R, x^I) + i\eta_t^I(x^R, x^I)$ , its derivatives, which are real-valued, are written as

$$\partial_R \eta_t^R(x) := \frac{\partial \eta_t^R(x^R, x^I)}{\partial x^R}, \quad \partial_I \eta_t^R(x) := \frac{\partial \eta_t^R(x^R, x^I)}{\partial x^I},$$

$$\partial_R \eta_t^I(x) := \frac{\partial \eta_t^I(x^R, x^I)}{\partial x^R}, \quad \text{and} \quad \partial_I \eta_t^I(x) := \frac{\partial \eta_t^I(x^R, x^I)}{\partial x^I}.$$

Starting from  $X^0 = O$  and  $Z^0 = Y$ , the AMP algorithm proceeds iteratively according to

$$X^{t+1} = \eta_t(A^* Z^t B^* + X^t), \quad (8)$$

$$\begin{aligned} Z^t &= Y - AX^t B \\ &+ \frac{1}{2\alpha} \langle H_{t-1}^{RR} + iH_{t-1}^{RI} \rangle Z_{t-1}^I \\ &- \frac{1}{2\alpha} \langle H_{t-1}^{IR} + iH_{t-1}^{II} \rangle Z_{t-1}^R, \end{aligned} \quad (9)$$

where

$$\begin{aligned} H_t^{RR} &:= \partial_R \eta_t^R(A^* Z^t B^* + X^t), \\ H_t^{RI} &:= \partial_I \eta_t^R(A^* Z^t B^* + X^t), \\ H_t^{IR} &:= \partial_R \eta_t^I(A^* Z^t B^* + X^t), \\ H_t^{II} &:= \partial_I \eta_t^I(A^* Z^t B^* + X^t). \end{aligned} \quad (10)$$

Here,  $\langle \cdot \rangle$  denotes an average of matrix entries, i.e.,  $\langle A \rangle := (pq)^{-1} \sum_{i=1}^p \sum_{j=1}^q a_{ij}$  for  $p \times q$  matrix  $A = (a_{ij})$ . When a given appropriate termination condition is satisfied, this algorithm stops and outputs the latest estimate  $X^t$ . ■

Derivation of the algorithm is given in Appendix A.

### IV. SIMULATIONS

To verify the performance of the proposed algorithm, we conducted computer simulations of two types. One is image reconstruction by the proposed algorithm, the other is empirical evaluations of perfect-reconstruction threshold.

### A. Image Reconstruction

We first show an example of image recovery. Fig. 1(a) shows the  $128 \times 128$ -pixel Shepp-Logan Phantom image, denoted by  $T$ , in which Haar wavelet coefficients less than 0.1 were discarded. The signal density  $\rho$  is 0.104. The coefficient matrix is denoted by  $X_0$ . To improve randomness, we employed random permutation matrix  $R_1$  and  $R_2$ , of which each column and row always have a single one. They are random, but known, as well as the random DFT below. Hence, the use of these matrices does not degrade robustness of the proposed algorithm in any sense, but well improves the randomness of the support of  $X_0$ . In this way,  $T$  is expressed by  $T = H^T R_1^T X_0 R_2^T H$ , where  $H$  is the one-dimensional Haar wavelet transform. The observation process is the random DFT, which is formulated by one-dimensional DFT matrices  $F_1$  and  $F_2$ , whose rows or columns are deleted randomly, applied from both side of  $T$ . Finally, we have Eq. (5), where  $A = F_1 H^T R_1^T$  and  $B = R_2^T H F_2^*$ . In this case, the target signal  $X_0$  is real and the proximity operator reduces to the real-valued version, as

$$\eta_t(x) = \left( x^R - \theta_t \text{sgn}(x^R) \right) \mathbb{I}(|x^R| > \theta_t) \in \mathbb{R}. \quad (11)$$

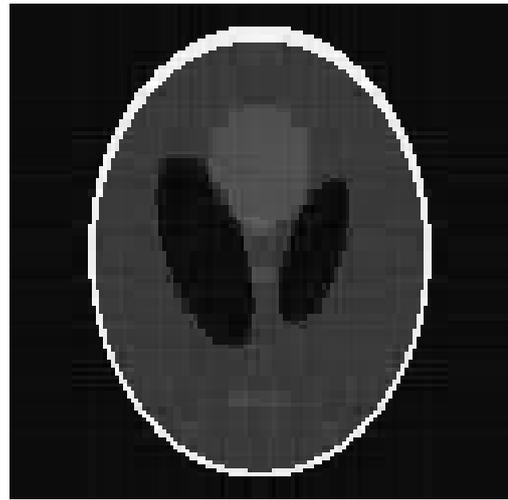
Therefore, the derivatives  $\partial_I \eta_t^R(x)$ ,  $\partial_R \eta_t^I(x)$  and  $\partial_I \eta_t^I(x)$  vanish from Eq. (9).

We observed  $112 \times 112$  measurements by Eq. (5). The compression rate is  $\alpha = 112^2/128^2 \approx 0.766$ . Given the complex measurement  $Y$  and the matrices  $A$  and  $B$ , we reconstructed  $X_0$  by the proposed algorithm and then the image  $T$  by  $H^T R_1^T X_0 R_2^T H$ . We implemented the algorithm by MATLAB on a windows 7 computer with 2.99GB memory. The results is shown in Fig. 1(b). Its mean square error (MSE) was  $6.89 \times 10^{-7}$ . The computational time was 42.07 seconds, which will be accelerated if the program is implemented by C language. Note that the same simulation using one-dimensional AMP with raster scan was not completed because of lack of memory. These results show the effectiveness of the proposed algorithm.

### B. Evaluation of Perfect-Reconstruction Threshold

In the previous section, we showed that the proposed algorithm successfully reconstructed sparse matrix of density  $\rho = 0.104$  from the measurements of compression rate  $\alpha \approx 0.766$ . In this section, we show how much measurements can be eliminated depending on the signal density  $\rho$  when the signal is reconstructed by the proposed algorithm.

The target signal  $X_0$  in Eq. (5) is randomly generated hundred times as follows. First, an independently, identically distributed gaussian random image of the corresponding size ( $32 \times 32 = 1024$  or  $48 \times 48 = 2304$ ) is generated. The image is dense and therefore a random mask is applied so that the resultant image becomes sparse of the supposed signal density. Each  $X_0$  is observed by Eq. (5), where  $A$  and  $B$  are the random DFT matrices generated for each  $X_0$ . By changing the number of measurements, we repeated signal recovery hundred times and recorded the compression rate at which all hundred signals are perfectly reconstructed. This means



(a) Original image.



(b) Reconstructed image.

Fig. 1. Example of image recovery of  $128 \times 128$  pixel.

that the square error of the reconstructed image to the original one is less than  $N \times 10^{-8}$ . The result is shown in Fig. 2. The red and blue lines show the results for  $32 \times 32 = 1024$  and  $48 \times 48 = 2304$  dimensional cases, respectively. The horizontal and vertical axes indicate  $\rho$  and  $\alpha$ , respectively. The dashed line denotes the weak threshold with the real-valued one-dimensional  $\ell_1$  recovery [5]. We can see that the compression rate is mostly same or even less than the weak threshold when  $0.2 < \rho < 0.8$ .

It is worthy to note that in the raster scan formulation, any combination of DFT coefficients can be deleted from the full set of the coefficients, while in the formulation of Eq. (5), the deletion is done by ‘row by row’ or ‘column by column’. This implies that our formulation has less randomness than in the raster scan formulation. It is interesting that, in spite of this, the compression rate gets lower than the weak threshold in the aforementioned range of  $\rho$ .

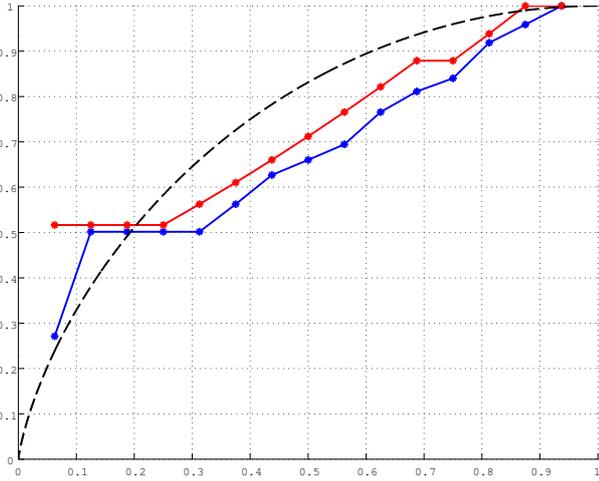


Fig. 2. Empirically evaluated perfect-reconstruction threshold of the proposed algorithm (2D-CAMP). The red and blue lines show the results for  $32 \times 32 = 1024$  and  $48 \times 48 = 2304$  dimensional images, respectively. The horizontal and vertical axes indicate  $\rho$  and  $\alpha$ , respectively. The dashed line denotes the weak threshold.

Let us close this section by stating the following comments about the complex target and complex measurement setup. Since both the real and imaginary parts take non-zero value at a same time in the case, it becomes easier to infer a support set of a signal. Therefore, the empirical performance can get lower than the weak threshold of the real-valued one-dimensional  $\ell_1$  recovery [5], nevertheless the situation is similar to the Kronecker compressed sensing.

## V. CONCLUSION

We proposed the AMP algorithm for the complex-valued two-dimensional compressed sensing. Our algorithm contributes to the large-scale two-dimensional sparse recovery. Some similar problems such as the dictionary learning, three-dimensional sparse signal recovery may be able to be treated by the similar framework addressed here. To discuss these problems is our future work.

## ACKNOWLEDGMENT

This work was supported by the Research Grant (A) No. 20 from Tateishi Science and Technology Foundation and partially supported by JSPS KAKENHI Grants No. 22500136, 25289114, and 25330264.

## APPENDIX

### A. Derivation of Algorithm 2

We here show an outline of the derivation of Algorithm 2. The optimization problem (6) is equivalent to the maximization of the joint distribution:

$$\mu_\beta(S) := \frac{1}{Z(\beta; Y)} e^{-2\beta \|S\|_1} \delta(Y - ASB), \quad (12)$$

with a parameter  $\beta > 0$ . An estimate can be, therefore, obtained by solving the maximization of the marginal distribution  $\mu_{\beta, ij}(S_{ij}) := \int dS_{\setminus ij} \mu_\beta(S)$  in the  $\beta \rightarrow \infty$  limit, namely

$$X_{ij} = \operatorname{argmax}_{S_{ij}} \lim_{\beta \rightarrow \infty} \mu_{\beta, ij}(S_{ij}). \quad (13)$$

The marginal distribution  $\mu_{\beta, ij}(S_{ij})$  can be approximately evaluated using the message passing algorithm [11] as

$$\mu_{\beta, ij}(S_{ij}) \propto e^{-\beta |S_{ij}|} \prod_{(ab)} \hat{\nu}_{(ab) \rightarrow (ij)}^t(S_{ij}), \quad (14)$$

where the distribution  $\hat{\nu}_{(ab) \rightarrow (ij)}^t(S_{ij})$  is a solution to the following simultaneous functional equations with respect to the distribution  $\hat{\nu}_{(ab) \rightarrow (ij)}^t(S_{ij})$  and  $\nu_{(ij) \rightarrow (ab)}^{t+1}(S_{ij})$ , which are called the messages.

$$\nu_{(ij) \rightarrow (ab)}^{t+1}(S_{ij}) \propto e^{-2\beta |S_{ij}|} \prod_{(cd) \neq (ab)} \hat{\nu}_{(cd) \rightarrow (ij)}^t(S_{ij}) \quad (15)$$

$$\begin{aligned} \hat{\nu}_{(ab) \rightarrow (ij)}^t(S_{ij}) &\propto \int_{\mathbb{C}^{N-1}} \prod_{(k\ell) \neq (ij)} \nu_{(ij) \rightarrow (ab)}^{t+1}(S_{ij}) \\ &\times \delta(Y_{ab} - (ASB)_{ab}) dS_{\setminus (ij)}. \end{aligned} \quad (16)$$

To proceed evaluation of the message, we assume that the distribution  $\nu_{(ij) \rightarrow (ab)}^{t+1}(S_{ij})$  is the complex normal distribution with mean  $x_{(ij) \rightarrow (ab)}^t$  and variance  $\tau_{(ij) \rightarrow (ab)}^t / \beta$ .

Under this assumption,  $\hat{\nu}_{(ab) \rightarrow (ij)}^t(S_{ij})$  becomes

$$\hat{\nu}_{(ab) \rightarrow (ij)}^t(S_{ij}) = \exp\left(-\frac{\beta}{\hat{\tau}_{(ab) \rightarrow (ij)}^t} |z_{(ab) \rightarrow (ij)}^t - A_{ai} S_{ij} B_{jb}|^2\right), \quad (17)$$

where

$$z_{(ab) \rightarrow (ij)}^t := Y_{ab} - \sum_{(k\ell) \neq (ij)} A_{ak} x_{(k\ell) \rightarrow (ab)}^t B_{\ell b}, \quad (18)$$

$$\hat{\tau}_{(ab) \rightarrow (ij)}^t := \sum_{(k\ell) \neq (ij)} \tau_{(k\ell) \rightarrow (ab)}^t |A_{ak}|^2 |B_{\ell b}|^2. \quad (19)$$

Hence, the distribution  $\hat{\nu}_{(ab) \rightarrow (ij)}^t(S_{ij})$  can be regarded as  $\text{CN}(z_{(ab) \rightarrow (ij)}^t / (A_{ai} B_{jb}), \hat{\tau}_{(ab) \rightarrow (ij)}^t / (\beta |A_{ai}|^2 |B_{jb}|^2))$ . Substituting (17) into (15) and approximating  $\tau_{(ab) \rightarrow (ij)}^t \approx \hat{\tau}^t$  with

$$\hat{\tau}^t := \sum_{(ij)} \tau_{(ij) \rightarrow (ab)}^t |A_{ai}|^2 |B_{jb}|^2, \quad (20)$$

we then have

$$\nu_{(ij) \rightarrow (ab)}^{t+1}(S_{ij}) = f_\beta\left(S_{ij}; \sum_{(cd) \neq (ab)} \overline{A_{ci}} z_{(cd) \rightarrow (ij)}^t \overline{B_{jd}}, \hat{\tau}^t\right), \quad (21)$$

where  $\bar{x}$  denotes the complex conjugate of a scalar complex value  $x \in \mathbb{C}$ . We defined the function (distribution)  $f_\beta$  as

$$f_\beta(s; x, b) := \frac{1}{z_\beta(x, b)} \exp\left(-2\beta |s| - \frac{\beta}{b} |s - x|^2\right),$$

where  $z_\beta(x, b)$  denotes a normalization constant.

Let  $Z$  be a random variable which follows  $Z \sim f_\beta(z; x, b)$ . We define the following two functions:

$$F_\beta(x, b) := \mathbb{E}_{f_\beta(z; x, b)}[Z], \quad (22)$$

$$G_\beta(x, b) := \mathbb{V}_{f_\beta(z; x, b)}[Z], \quad (23)$$

where  $\mathbb{E}_\mu$  and  $\mathbb{V}_\mu$  denote the expectation and variance operators with respect to a random variable that follows  $\mu$ , respectively. Since we put the mean and variance of  $\nu_{(ij) \rightarrow (ab)}^{t+1}(S_{ij})$  as  $x_{(ij) \rightarrow (ab)}^t$  and  $\tau_{(ij) \rightarrow (ab)}^t/\beta$ , these are represented by

$$x_{(ij) \rightarrow (ab)}^t = F_\beta \left( \sum_{(cd) \neq (ab)} \overline{A_{ci} z_{(cd) \rightarrow (ij)}^t} \overline{B_{jd}}, \hat{\tau}^t \right),$$

$$\tau_{(ij) \rightarrow (ab)}^t / \beta = G_\beta \left( \sum_{(cd) \neq (ab)} \overline{A_{ci} z_{(cd) \rightarrow (ij)}^t} \overline{B_{jd}}, \hat{\tau}^t \right),$$

respectively.

In the  $\beta \rightarrow \infty$  limit,  $F_\beta(x, b)$  and  $G_\beta(x, b)$  are dominated by a value  $z$  which maximizes the function  $f_\beta(z; x, b)$  for given parameters  $(x, b)$ . Such a value is given as

$$\eta(x, b) := \operatorname{argmax}_{s \in \mathbb{C}} \lim_{\beta \rightarrow \infty} f_\beta(s; x, b)$$

$$= \begin{cases} x - b \frac{x}{|x|} & \text{if } |x| > b \\ 0 & \text{if } |x| \leq b, \end{cases}$$

which is called the proximity operator. We therefore have

$$\lim_{\beta \rightarrow \infty} F_\beta(x, b) = \eta(x, b) \quad (24)$$

$$\lim_{\beta \rightarrow \infty} \beta G_\beta(x, b) = \begin{cases} b & \text{if } |x| > b \\ 0 & \text{if } |x| \leq b \end{cases} =: \hat{\eta}(x, b) \quad (25)$$

in the  $\beta \rightarrow \infty$  limit. We can summarize a result until here as follows:

$$x_{(ij) \rightarrow (ab)}^t = \eta \left( \sum_{(cd) \neq (ab)} \overline{A_{ci} z_{(cd) \rightarrow (ij)}^t} \overline{B_{jd}}, \hat{\tau}^t \right),$$

$$z_{(ab) \rightarrow (ij)}^t := Y_{ab} - \sum_{(kl) \neq (ij)} A_{ak} x_{(kl) \rightarrow (ab)}^t B_{lb},$$

$$\hat{\tau}^t = \frac{\hat{\tau}}{N_1 N_2 \delta} \sum_{(ij)} \hat{\eta} \left( \sum_{(cd) \neq (ab)} \overline{A_{ci} z_{(cd) \rightarrow (ij)}^t} \overline{B_{jd}}, \hat{\tau}^t \right).$$

We next consider the following generic algorithm that is derived by omitting details of the proximity operator.

$$x_{(ij) \rightarrow (ab)}^t = \eta_t \left( \sum_{(cd) \neq (ab)} \overline{A_{ci} z_{(cd) \rightarrow (ij)}^t} \overline{B_{jd}} \right), \quad (26)$$

$$z_{(ab) \rightarrow (ij)}^t := Y_{ab} - \sum_{(kl) \neq (ij)} A_{ak} x_{(kl) \rightarrow (ab)}^t B_{lb}, \quad (27)$$

and assume that

$$x_{(ij) \rightarrow (ab)}^t = x_{ij}^t + \delta x_{(ij) \rightarrow (ab)}^t + O(1/N), \quad (28)$$

$$z_{(ab) \rightarrow (ij)}^t = z_{ab}^t + \delta z_{(ab) \rightarrow (ij)}^t + O(1/N), \quad (29)$$

$$\delta x_{(ij) \rightarrow (ab)}^t = O(1/\sqrt{N}), \quad (30)$$

$$\delta z_{(ab) \rightarrow (ij)}^t = O(1/\sqrt{N}). \quad (31)$$

Substituting (28)–(31) into (26), we then have

$$x_{ij}^t + \delta x_{(ij) \rightarrow (ab)}^t = \eta_t \left( \sum_{(cd)} \overline{A_{ci} z_{(cd) \rightarrow (ij)}^t} \overline{B_{jd}} - \overline{A_{ai} z_{ab}^t} \overline{B_{jb}} \right). \quad (32)$$

Since the proximity operator is non-holomorphic, the term  $\delta x_{(ij) \rightarrow (ab)}^t$  in (32) can be evaluated by expansion of the proximity function as two “two-variable functions”, namely  $\eta_t(x) = \eta_t^R(x^R, x^I) + i\eta_t^I(x^R, x^I)$ , with respect to a small term  $\overline{A_{ai} z_{ab}^t} \overline{B_{jb}}$ . Using a similar way  $\delta z_{(ab) \rightarrow (ij)}^t$  can be also obtained. We therefore obtain

$$x_{ij}^t = \eta_t \left( \sum_{(ab)} \overline{A_{ai} z_{(ab) \rightarrow (ij)}^t} \overline{B_{jb}} + x_{ij}^t \right), \quad (33)$$

$$z_{ab}^t = Y_{ab} - \sum_{(kl)} A_{ak} x_{(kl)}^t B_{lb}$$

$$+ \sum_{(kl)} A_{ak} \mathcal{R}(\overline{A_{ak} z_{ab}^{t-1}} \overline{B_{lb}}) B_{lb} \partial_R \eta_t^R(C_{kl}^{ab})$$

$$+ \sum_{(kl)} A_{ak} \mathcal{I}(\overline{A_{ak} z_{ab}^{t-1}} \overline{B_{lb}}) B_{lb} \partial_I \eta_t^R(C_{kl}^{ab})$$

$$+ i \sum_{(kl)} A_{ak} \mathcal{R}(\overline{A_{ak} z_{ab}^{t-1}} \overline{B_{lb}}) B_{lb} \partial_R \eta_t^I(C_{kl}^{ab})$$

$$+ i \sum_{(kl)} A_{ak} \mathcal{I}(\overline{A_{ak} z_{ab}^{t-1}} \overline{B_{lb}}) B_{lb} \partial_I \eta_t^I(C_{kl}^{ab}), \quad (34)$$

where  $C_{kl}^{ab} := \sum_{(cd) \neq (ab)} \overline{A_{ck} z_{cd}^{t-1}} \overline{B_{ld}} + x_{kl}^{t-1}$ ,  $\mathcal{R}(x)$  and  $\mathcal{I}(x)$  denote the real and imaginary parts of  $x$ . We assume that the Onsager term, which is a summation from the third to the sixth terms in the right-hand side of (34), is concentrated around its expectation value. Evaluating the expectation value of the Onsager term, we arrive at Algorithm 2.

## REFERENCES

- [1] D. L. Donoho, “Compressed Sensing,” *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.
- [2] E. J. Candes and T. Tao, “Decoding by linear programming,” *IEEE Trans. Inf. Theory*, vol. 51, no. 12, pp. 4203–4215, Dec. 2005.
- [3] Y. C. Eldar and G. Kutyniok, “Compressed Sensing: Theory and Applications,” Cambridge University Press, 2011.
- [4] S. Boyd and L. Vandenberghe, “Convex Optimization,” Cambridge University Press, 2004.
- [5] D. L. Donoho, A. Maleki and A. Montanari, “Message-passing algorithms for compressed sensing,” *Proc. the National Academy of Sciences*, vol. 106, no. 45, pp. 18914–18919, Nov. 2009.
- [6] M. Angelini, F. Ricci-Tersenghi, Y. Kabashima, “Compressed Sensing with Sparse, Structured Matrices,” *Proc. the 50th Annual Allerton Conference on Communication, Control, and Computing*, Oct., 2012.
- [7] S. Jökar, “Sparse Recovery and Kronecker Products,” *Proc. of the 44th Annual Conference on Information Sciences and Systems (CISS2010)*, 17–19, Mar., 2010.
- [8] M. F. Duarte and R. G. Baraniuk, “Kronecker Compressive Sensing,” *IEEE Trans. on Image Processing*, vol. 21, no. 2, 494–504, Feb., 2012.
- [9] A. Maleki, L. Anitori, Z. Yang, and R. Baraniuk, “Asymptotic analysis of complex LASSO via complex approximate message passing (CAMP),” to be published in *IEEE Trans. Inf. Theory*.
- [10] A. K. Jain, “Fundamentals of Digital Image Processing,” Prentice Hall, 1988.
- [11] J. Pearl, “Fusion, propagation, and structuring in belief networks,” *Artificial intelligence*, vol. 29, no. 3, pp. 241–288, Sep. 1986.