IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received October 19, 2018, accepted December 11, 2018, date of publication December 18, 2018, date of current version January 11, 2019.

*Digital Object Identifier 10.1109/ACCESS.2018.2887261*

# Bilinear Adaptive Generalized Vector Approximate Message Passing

**XIANGMING MENG[1], AND JIANG ZHU[2], (Member, IEEE)**
[1]Huawei Technologies, Co., Ltd., Shanghai 201206, China
[2]Key Laboratory of Ocean Observation-Imaging Testbed of Zhejiang Province, Ocean College, Zhejiang University, Zhoushan 316021, China
Corresponding author: Jiang Zhu (jiangzhu16@zju.edu.cn)

This work was supported in part by the Zhejiang Province Science Foundation of China under Grant LQ18F010001.

**ABSTRACT** This paper considers the generalized bilinear recovery problem, which aims to jointly recover the vector $\mathbf{b}$ and the matrix $\mathbf{X}$ from componentwise nonlinear measurements $\mathbf{Y} \sim p(\mathbf{Y}|\mathbf{Z}) = \prod_{i,j} p(Y_{ij}|Z_{ij})$, where $\mathbf{Z} = \mathbf{A}(\mathbf{b})\mathbf{X}$, $\mathbf{A}(\cdot)$ is a known affine linear function of $\mathbf{b}$, and $p(Y_{ij}|Z_{ij})$ is a scalar conditional distribution that models the general output transform. A wide range of real-world applications, e.g., quantized compressed sensing with matrix uncertainty, blind self-calibration and dictionary learning from nonlinear measurements, one-bit matrix completion, and joint channel and data decoding, can be cast as the generalized bilinear recovery problem. To address this problem, we propose a novel algorithm called the Bilinear Adaptive Generalized Vector Approximate Message Passing (BAd-GVAMP), which extends the recently proposed Bilinear Adaptive Vector AMP algorithm to incorporate arbitrary distributions on the output transform. The numerical results on various applications demonstrate the effectiveness of the proposed BAd-GVAMP algorithm.

**INDEX TERMS** Generalized bilinear model, approximate message passing, expectation propagation, expectation maximization, dictionary learning, self-calibration, matrix factorization.

## I. INTRODUCTION

In this work, we consider the generalized bilinear recovery problem: jointly estimate the vector $\mathbf{b}$ and the matrix $\mathbf{X}$ from componentwise and probabilistic measurements $\mathbf{Y} \sim p(\mathbf{Y}|\mathbf{Z}) = \prod_{i,j} p(Y_{ij}|Z_{ij})$, where $\mathbf{Z} = \mathbf{A}(\mathbf{b})\mathbf{X}$, $\mathbf{A}(\cdot)$ is a known affine linear function of $\mathbf{b}$ (i.e., $\mathbf{A}(\mathbf{b}) = \mathbf{A}_0 + \sum_{i=1}^{G} b_i\mathbf{A}_i$ with known matrices $\mathbf{A}_i$.). This problem arises in a wide range of applications in the field of signal processing and computer science. For example, compressed sensing under matrix uncertainty [1]–[4], matrix completion [5]–[7], robust principle component analysis (RPCA) [8], dictionary learning [9], [10], joint channel and data decoding [11]–[13] can all be formulated as generalized bilinear recovery problem. Generally the scalar conditional distribution $p(Y_{ij}|Z_{ij})$ models arbitrary componentwise measurement process in a probabilistic manner. Specially, $p(Y_{ij}|Z_{ij}) = \mathcal{N}(Y_{ij}; Z_{ij}, \gamma_w^{-1})$ corresponds to the scenario of linear measurements, i.e., $\mathbf{Y} = \mathbf{A}(\mathbf{b})\mathbf{X}+\mathbf{W}$, where $\mathcal{N}(x; a, \gamma^{-1})$ denotes a Gaussian distribution with mean and variance being $a$ and $\gamma^{-1}$. In practice, however, the measurements are often obtained in a nonlinear way. For example, quantization is a common nonlinear measurement process in analog-to-digital converter (ADC) that maps the input signal from continuous space to discrete space, which has been widely used in (one-bit) compressed sensing [14], millimeter massive multiple input multiple output (MIMO) system [15], [16], etc. As a result, it is of high significance to study the generalized bilinear recovery problem.

There has been extensive research on this active field in the past few years, including the convex relaxation methods [17], [18], variational methods [19], approximate message passing (AMP) methods such as bilinear generalized AMP (BiGAMP) [9], [10] and parametric BiGAMP (PBiGAMP) [20], etc. It was shown that the AMP based methods are competitive in terms of phase transition and computation time [9], [10], [20], [21]. However, as the measurement matrix deviates from the i.i.d. Gaussian, the AMP may diverge [22], [23]. To improve convergence of AMP, vector approximate message passing (VAMP) [24] and orthogonal AMP (OAMP) [25] have been recently proposed, which achieve good convergence performance for any right-rotationally invariant measurement matrices and can be rigorously characterized by the scalar state evolution. For the generalized linear model (GLM), AMP is extended to generalized approximate message passing (GAMP) [26], [27]. Later, generalized VAMP [28] and generalized expectation consistent algorithm [29] are proposed to handle a class of right-rotationally invariant measurement matrices. In [30], a unified Bayesian inference framework was proposed for GLM which iteratively decouples the original GLM into a sequence of standard linear models (SLM). Such unified framework provides new perspectives on some of the existing GLM algorithms, e.g., GAMP and GVAMP, and more importantly, significantly facilitates the design of GLM inference algorithms. Due to the improved convergence of VAMP over AMP on general measurement matrices, many works have been done to extend VAMP to deal with the bilinear recovery problem [31], [32]. In [31], lifted VAMP is proposed for standard bilinear inference problem such as compressed sensing with matrix uncertainty and self-calibration. However, lift VAMP suffers from high computational complexity since the number of unknowns increases significantly, especially when the number of original variables is large. To overcome the computation issue, the bilinear adaptive VAMP (BAd-VAMP) has been proposed very recently in [33] which avoids lifting and instead builds on the adaptive AMP framework [34], [35]. Nevertheless, BAd-VAMP is only applicable to linear measurements which limits its usage in the generalized bilinear recovery problem.

In this paper, we propose a new algorithm called the bilinear adaptive generalized vector AMP (BAd-GVAMP), which extends the BAd-VAMP [33] from linear measurements to nonlinear measurements. Specifically, a novel factor graph representation of the generalized bilinear problem is first proposed by incorporating the Dirac delta function. Then, similar to [30], by using the expectation propagation (EP) [36], we decouple the original generalized bilinear recovery problem into two modules: one module performs componentwise minimum mean square error (MMSE) estimate while the other performs BAd-VAMP with some slight modification of the message passing schedule. Furthermore, the messages exchanging between the two modules are derived to obtain the final BAd-GVAMP. Interestingly, BAd-GVAMP reduces to the BAd-VAMP under linear measurements. Numerical results are conducted for quantized compressed sensing with matrix uncertainty, self-calibration as well as structured dictionary learning from quantized measurements, which demonstrates the effectiveness of the proposed algorithm.

### A. NOTATION

Let $\mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ denote a Gaussian distribution of the random variable $\mathbf{x}$ with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$. Let $(\cdot)^{\mathrm{T}}$, $\|\cdot\|_F$, $\|\cdot\|$, and $\delta(\cdot)$ denote the transpose operator, the Frobenius norm, the $l_2$ norm, the probability density function (PDF) and the Dirac delta function, respectively. Let $<\mathbf{x}>$ denote the average $<\mathbf{x}>=\sum_{i=1}^{N} x_i/N$ for $\mathbf{x} \in \mathbb{R}^N$.

## II. PROBLEM SETUP

Consider the generalized bilinear recovery problem as follows: jointly estimate the matrix $\mathbf{X} \in \mathbb{R}^{N\times L}$ and the parameters $\boldsymbol{\Theta} \triangleq \{\boldsymbol{\theta}_X, \boldsymbol{\theta}_A, \boldsymbol{\theta}_Y\}$ from the componentwise probabilistic measurements $\mathbf{Y} \in \mathbb{R}^{M\times L}$, i.e.,

$$\mathbf{X} \sim p(\mathbf{X}; \boldsymbol{\theta}_X) = \prod_{i,j} p(x_{ij}; \boldsymbol{\theta}_X) = \prod_{l=1}^{L} p(\mathbf{x}_l; \boldsymbol{\theta}_X), \tag{1a}$$

$$\mathbf{Z} = \mathbf{A}(\boldsymbol{\theta}_A)\mathbf{X}, \tag{1b}$$

$$\mathbf{Y} \sim p(\mathbf{Y}|\mathbf{Z}; \boldsymbol{\theta}_Y) = \prod_{i,j} p(Y_{ij}|Z_{ij}; \boldsymbol{\theta}_Y) = \prod_{l=1}^{L} p(\mathbf{y}_l|\mathbf{z}_l; \boldsymbol{\theta}_Y), \tag{1c}$$

where $\mathbf{Y}$ denotes the nonlinear observations, $\mathbf{A}(\cdot) \in \mathbb{R}^{M\times N}$ is a known matrix-valued linear function parameterized by the unknown vector $\boldsymbol{\theta}_A$, $p(\mathbf{X}; \boldsymbol{\theta}_X)$ is the prior distribution of $\mathbf{X}$ parameterized by $\boldsymbol{\theta}_X$, $p(\mathbf{Y}|\mathbf{Z}; \boldsymbol{\theta}_Y)$ is the componentwise probabilistic output distribution conditioned on $\mathbf{Z}$ and parameterized by $\boldsymbol{\theta}_Y$. Given the above statistical model, the goal is to compute the maximum likelihood (ML) estimate of $\boldsymbol{\Theta}$ and the MMSE estimate of $\mathbf{X}$, i.e.,

$$\hat{\boldsymbol{\Theta}}_{\mathrm{ML}} = \underset{\boldsymbol{\Theta}}{\operatorname{argmax}}\, p_{\mathbf{Y}}(\mathbf{Y}; \boldsymbol{\Theta}), \tag{2}$$

$$\hat{\mathbf{X}}_{\mathrm{MMSE}} = \mathrm{E}[\mathbf{X}|\mathbf{Y}; \hat{\boldsymbol{\Theta}}_{\mathrm{ML}}], \tag{3}$$

where $p_{\mathbf{Y}}(\mathbf{Y}; \boldsymbol{\Theta}) = \int p(\mathbf{X}; \boldsymbol{\Theta})p(\mathbf{Y}|\mathbf{X}; \boldsymbol{\Theta})\mathrm{d}\mathbf{X}$ is the likelihood function of $\boldsymbol{\Theta}$ and the expectation is taken with respect to the posterior probability density distribution

$$p(\mathbf{X}|\mathbf{Y}; \hat{\boldsymbol{\Theta}}_{\mathrm{ML}}) = \frac{p(\mathbf{X}, \mathbf{Y}; \hat{\boldsymbol{\Theta}}_{\mathrm{ML}})}{p(\mathbf{Y}; \hat{\boldsymbol{\Theta}}_{\mathrm{ML}})}, \tag{4}$$

where $p(\mathbf{X}, \mathbf{Y}; \hat{\boldsymbol{\Theta}}_{\mathrm{ML}})$ is

$$p(\mathbf{X}, \mathbf{Y}; \hat{\boldsymbol{\Theta}}_{\mathrm{ML}}) = p(\mathbf{X}; \hat{\boldsymbol{\Theta}}_{\mathrm{ML}})p(\mathbf{Y}|\mathbf{X}; \hat{\boldsymbol{\Theta}}_{\mathrm{ML}}). \tag{5}$$

However, exact ML estimate of $\boldsymbol{\Theta}$ and exact MMSE estimate of $\mathbf{X}$ is intractable due to high-dimensional integration. As a result, approximate methods need to be designed in practice.

## III. BILIEAR ADAPTIVE GENERALIZED VAMP

In this section, we propose an efficient algorithm to approximate the ML estimate of $\boldsymbol{\Theta}$ and MMSE estimate of $\mathbf{X}$. The resultant BAd-GVAMP algorithm is an extension of BAd-VAMP from linear measurements to nonlinear measurements. To begin with, we first present a novel factor graph representation of the statistical model. By introducing a hidden variable $\mathbf{Z}$ and a Dirac delta function $\delta(\cdot)$, the joint distribution in (5) can be equivalently factored as

$$p(\mathbf{X}, \mathbf{Y}; \boldsymbol{\Theta}) = p(\mathbf{X}; \boldsymbol{\theta}_X)p(\mathbf{Y}|\mathbf{Z}; \boldsymbol{\theta}_Y)\delta(\mathbf{Z} - \mathbf{A}(\boldsymbol{\theta}_A)\mathbf{X}). \tag{6}$$

The corresponding factor graph of (6) is shown in Fig. 1 (a). The circles and squares denote the variable and factor node, respectively. Such alternative factor graph representation plays a key role in the design of our approximate estimation algorithm. Now we will derive the BAd-GVAMP

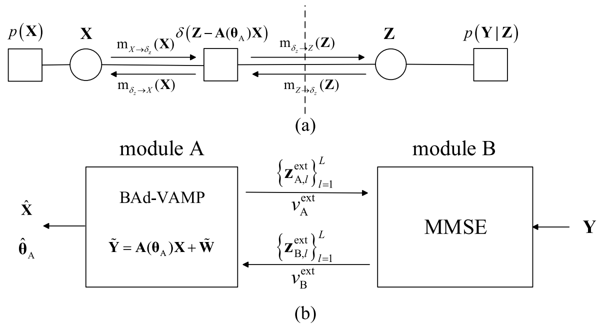



**FIGURE 1. The factor graph and inference module of the BAd-GVAMP algorithm.**

algorithm based on the presented factor graph in Fig. 1 (a) and the EP [36]. As one kind of approximate inference methods, EP approximates the target distribution p with an exponential family distribution (usually Gaussian) set $\boldsymbol{\Phi}$ which minimizes the Kullback-Leibler (KL) divergence $\mathrm{KL}(p||q)$, i.e., $q = \mathrm{Proj}(p) = \underset{q\in\boldsymbol{\Phi}}{\mathrm{argmin}}\, \mathrm{KL}(p||q)$. For Gaussian distribution set $\boldsymbol{\Phi}$, EP amounts to moment matching, i.e., the first and second moments of distribution $q$ matches those of the target distribution. For more details of EP and its relation to AMP methods, please refer to [24], [36], and [38]–[41].

To address the generalized bilinear recovery problem, specifically, we choose the projection set $\boldsymbol{\Phi}$ to be Gaussian with scalar covariance matrix, i.e., diagonal matrix whose diagonal elements are equal.[1] Then, using EP on the factor graph in Fig. 1, we decouple the original generalized bilinear recovery problem into two modules: the componentwise MMSE module and the BAd-VAMP module. The two modules interact with each other iteratively with extrinsic messages exchanging between them in a similar manner as [30]. The detailed derivation of BAd-GVAMP is presented as follows.

### A. COMPONENTWISE MMSE MODULE

Suppose that in the $t$-th iteration, the message $m^{t-1}_{\delta_z\to\mathbf{Z}}(\mathbf{Z}) = \{m^{t-1}_{\delta_z\to\mathbf{z}_l}(\mathbf{z}_l)\}_{l=1}^L$ from factor node $\delta(\mathbf{Z}-\mathbf{A}(\boldsymbol{\theta}_A)\mathbf{X})$ to variable node $\mathbf{Z}$ follows Gaussian distribution, i.e.,

$$m^{t-1}_{\delta_z\to\mathbf{z}_l}(\mathbf{z}_l) = \mathcal{N}(\mathbf{z}_l;\mathbf{z}^{\mathrm{ext}}_{\mathrm{A},l}(t-1), v^{\mathrm{ext}}_{\mathrm{A}}(t-1)\mathbf{I}_M), \tag{7}$$

where $\delta_z$ refers to the factor node $\delta(\mathbf{Z}-\mathbf{A}(\boldsymbol{\theta}_A)\mathbf{X})$. According to EP, the message $m^t_{\mathbf{z}_l\to\delta_z}(\mathbf{z}_l)$ from variable node $\mathbf{Z}$ to the factor node $\delta(\mathbf{Z}-\mathbf{A}(\boldsymbol{\theta}_A)\mathbf{X})$ can be calculated as

$$m^t_{\mathbf{z}_l\to\delta_z}(\mathbf{z}_l) \propto \frac{\mathrm{Proj}[p(\mathbf{y}_l|\mathbf{z}_l;\boldsymbol{\theta}_Y^{t-1})m^{t-1}_{\delta_z\to\mathbf{z}_l}(\mathbf{z}_l)]}{m^{t-1}_{\delta_z\to\mathbf{z}_l}(\mathbf{z}_l)} \tag{8}$$

$$\triangleq \frac{\mathrm{Proj}[q^t_{\mathrm{B}}(\mathbf{z}_l)]}{m^{t-1}_{\delta_z\to\mathbf{z}_l}(\mathbf{z}_l)}, \tag{9}$$

where $\propto$ denotes identity up to a normalizing constant. First, we perform componentwise MMSE and obtain the posterior means and variances of $\mathbf{z}_l$ as

$$\mathbf{z}^{\mathrm{post}}_{\mathrm{B},l}(t) = \mathrm{E}[\mathbf{z}_l|q^t_{\mathrm{B}}(\mathbf{z}_l)], \tag{10}$$

$$v^{\mathrm{post}}_{\mathrm{B},l}(t) = <\mathrm{Var}[\mathbf{z}_l|q^t_{\mathrm{B}}(\mathbf{z}_l)]>, \tag{11}$$

where $\mathrm{E}[\cdot|q^t_{\mathrm{B}}(\mathbf{z}_l)]$ and $\mathrm{Var}[\cdot|q^t_{\mathrm{B}}(\mathbf{z}_l)]$ are the mean and variance operations taken (componentwise) with respect to the distribution $\propto q^t_{\mathrm{B}}(\mathbf{z}_l)$ (9) and $<\mathbf{x}>$ denotes the average $<\mathbf{x}>=\sum_{i=1}^N x_i/N$ for $\mathbf{x}\in\mathbb{R}^N$. Then the posterior variances $v^{\mathrm{post}}_{\mathrm{B},l}(t)$ are averaged over $l$ which yields

$$v^{\mathrm{post}}_{\mathrm{B}}(t) = \sum_{l=1}^L v^{\mathrm{post}}_{\mathrm{B},l}(t)/L, \tag{12}$$

so that $\mathrm{Proj}[q^t_{\mathrm{B}}(\mathbf{z}_l)]$ is approximated as

$$\mathrm{Proj}[q^t_{\mathrm{B}}(\mathbf{z}_l)] \approx \mathcal{N}(\mathbf{z}_l;\mathbf{z}^{\mathrm{post}}_{\mathrm{B},l}(t), v^{\mathrm{post}}_{\mathrm{B}}(t)\mathbf{I}_M) \triangleq \tilde{q}^t_{\mathrm{B}}(\mathbf{z}_l). \tag{13}$$

As a result, the message $m^t_{\mathbf{z}_l\to\delta_z}(\mathbf{z}_l)$ from the variable node $\mathbf{Z}$ to the factor node $\delta(\mathbf{Z}-\mathbf{A}(\boldsymbol{\theta}_A)\mathbf{X})$ can be calculated (componentwise) as

$$m^t_{\mathbf{z}_l\to\delta_z}(\mathbf{z}_l) \propto \frac{\mathcal{N}(\mathbf{z}_l;\mathbf{z}^{\mathrm{post}}_{\mathrm{B},l}(t), v^{\mathrm{post}}_{\mathrm{B}}(t)\mathbf{I}_M)}{\mathcal{N}(\mathbf{z}_l;\mathbf{z}^{\mathrm{ext}}_{\mathrm{A},l}(t), v^{\mathrm{ext}}_{\mathrm{A}}(t)\mathbf{I}_M)}, \tag{14a}$$

$$\propto \mathcal{N}(\mathbf{z}_l;\mathbf{z}^{\mathrm{ext}}_{\mathrm{B},l}(t), v^{\mathrm{ext}}_{\mathrm{B}}(t)\mathbf{I}_M), \tag{14b}$$

where the extrinsic means $\mathbf{z}^{\mathrm{ext}}_{\mathrm{B},l}(t)$ and variances $v^{\mathrm{ext}}_{\mathrm{B}}(t)$ are

$$v^{\mathrm{ext}}_{\mathrm{B}}(t) = \left(\frac{1}{v^{\mathrm{post}}_{\mathrm{B}}(t)} - \frac{1}{v^{\mathrm{ext}}_{\mathrm{A}}(t)}\right)^{-1}, \tag{15}$$

$$\mathbf{z}^{\mathrm{ext}}_{\mathrm{B},l}(t) = v^{\mathrm{ext}}_{\mathrm{B}}(t)\left(\frac{\mathbf{z}^{\mathrm{post}}_{\mathrm{B},l}(t)}{v^{\mathrm{post}}_{\mathrm{B}}(t)} - \frac{\mathbf{z}^{\mathrm{ext}}_{\mathrm{A},l}(t)}{v^{\mathrm{ext}}_{\mathrm{A}}(t)}\right). \tag{16}$$

To learn the unknown parameter $\boldsymbol{\theta}_Y$, EM can be adopted [37], i.e.,

$$\boldsymbol{\theta}_Y(t) = \underset{\boldsymbol{\theta}_Y}{\mathrm{argmax}}\, \mathrm{E}\left[\log p(\mathbf{Y}|\mathbf{Z};\boldsymbol{\theta}_Y)m^{t-1}_{\delta_z\to\mathbf{Z}}(\mathbf{Z})|\tilde{q}^t_{\mathrm{B}}(\mathbf{z}_l)\right]$$

$$= \underset{\boldsymbol{\theta}_Y}{\mathrm{argmax}} \sum_{l=1}^L \mathrm{E}\left[\log p(\mathbf{y}_l|\mathbf{z}_l;\boldsymbol{\theta}_Y)|\tilde{q}^t_{\mathrm{B}}(\mathbf{z}_l)\right], \tag{17}$$

where $\tilde{q}^t_{\mathrm{B}}(\mathbf{z}_l)$ is given by (13).

### B. BAD-VAMP MODULE

As shown in (14), the message $m^t_{\mathbf{z}_l\to\delta_z}(\mathbf{z}_l)$ from the variable node $\mathbf{Z}$ to the factor node $\delta(\mathbf{Z}-\mathbf{A}(\boldsymbol{\theta}_A)\mathbf{X})$ follows Gaussian distribution $\mathcal{N}(\mathbf{z}_l;\mathbf{z}^{\mathrm{ext}}_{\mathrm{B},l}(t), v^{\mathrm{ext}}_{\mathrm{B}}(t)\mathbf{I}_M)$. Referring to the definition of the $\delta(\cdot)$ for the factor node, we obtain a pseudo linear observation equation as

$$\tilde{\mathbf{y}}_l(t) = \mathbf{A}(\boldsymbol{\theta}_A)\mathbf{x}_l + \tilde{\mathbf{w}}_l, \quad l=1,\cdots,L, \tag{18}$$

where $\tilde{\mathbf{y}}_l(t) \triangleq \mathbf{z}^{\mathrm{ext}}_{\mathrm{B},l}(t)$, $\tilde{\mathbf{w}}_l \sim \mathcal{N}(\tilde{\mathbf{w}}_l;\mathbf{0};\tilde{\gamma}_w^{-1}(t)\mathbf{I}_M)$ and $\tilde{\gamma}_w(t) \triangleq 1/v^{\mathrm{ext}}_{\mathrm{B}}(t)$. The factor graph corresponding to (18) is shown Fig. 2, where the dash square is used to indicate pseudo observations. As a result, the BAd-VAMP algorithm [33] for the standard bilinear recovery problem can be applied.

[1] Note that general diagonal matrix can also be used.

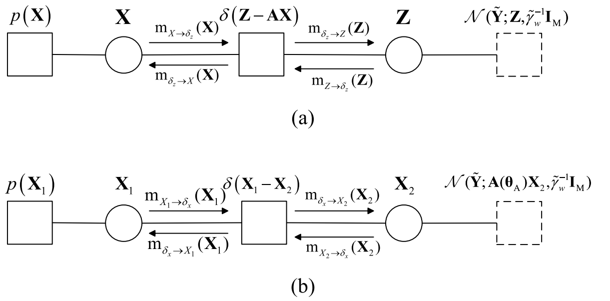



**FIGURE 2.** Two equivalent factor graphs for the pseudo linear observation model (18). Note that Fig. 2 (b) is the factor graph proposed by [24].

For completeness and ease of reference, we present the derivation of BAd-VAMP in [33] based on the factor graph shown in Fig. 2 (b), in which replicas of $\mathbf{X}$ are introduced, i.e., $\mathbf{X}_1 = \mathbf{X}_2 = \mathbf{X}$. In the following, let $\mathbf{x}_{1,l}$ and $\mathbf{x}_{2,l}$ denote the $l$ th column of $\mathbf{X}_1$ and $\mathbf{X}_2$, respectively. Assume that the message $\{m^{t-1}_{\delta_x\to\mathbf{x}_{2,l}}(\mathbf{x}_{2,l})\}_{l=1}^{L}$ transmitted from the factor node $\delta(\mathbf{X}_1-\mathbf{X}_2)$ to the variable node $\mathbf{X}_2$ is

$$m^{t-1}_{\delta_x\to\mathbf{x}_{2,l}}(\mathbf{x}_{2,l}) = \mathcal{N}(\mathbf{x}_{2,l};\mathbf{r}_{2,l}(t-1);\gamma^{-1}_{2,l}(t-1)\mathbf{I}_N), \tag{19}$$

where $\delta_x$ refers to the factor node $\delta(\mathbf{X}_1-\mathbf{X}_2)$. Note that $m^{t-1}_{\delta_x\to\mathbf{x}_{2,l}}(\mathbf{x}_{2,l})$ can be viewed as the prior of $\mathbf{x}_{2,l}$. Combining the pseudo observation equation (18) with $\boldsymbol{\theta}_A(t-1)$, the linear MMSE (LMMSE) estimate of $\mathbf{x}_{2,l}$ is performed and the posterior distribution of $\mathbf{x}_{2,l}$ is obtained as

$$q_2^t(\mathbf{X}_2) = \prod_{l=1}^{L}\mathcal{N}(\mathbf{x}_{2,l};\hat{\mathbf{x}}_{2,l}(t),\boldsymbol{\Xi}_{\mathbf{x}_{2,l}}(t)) \triangleq \prod_{l=1}^{L} q_2^t(\mathbf{x}_{2,l}), \tag{20}$$

where the posterior mean $\hat{\mathbf{x}}_{2,l}(t)$ and covariance matrix $\boldsymbol{\Xi}_{\mathbf{x}_{2,l}}(t)$ are

$$\begin{aligned}&\hat{\mathbf{x}}_{2,l}(t)\\ &= (\tilde{\gamma}_w(t)\mathbf{A}^{\mathrm{T}}(\boldsymbol{\theta}_A(t-1))\mathbf{A}(\boldsymbol{\theta}_A(t-1))+\gamma_{2,l}(t-1)\mathbf{I}_N)^{-1}\\ &\quad\times(\tilde{\gamma}_w(t)\mathbf{A}^{\mathrm{T}}(\boldsymbol{\theta}_A(t-1))\tilde{\mathbf{y}}_l(t)+\gamma_{2,l}(t-1)\mathbf{r}_{2,l}(t-1)),\end{aligned} \tag{21}$$

$$\begin{aligned}&\boldsymbol{\Xi}_{\mathbf{x}_{2,l}}(t)\\ &= \times(\tilde{\gamma}_w(t)\mathbf{A}^{\mathrm{T}}(\boldsymbol{\theta}_A(t-1))\mathbf{A}(\boldsymbol{\theta}_A(t-1))+\gamma_{2,l}(t-1)\mathbf{I}_N)^{-1}.\end{aligned} \tag{22}$$

In addition, the EM algorithm is incorporated to learn $\boldsymbol{\theta}_A$ and update the pseudo noise precision $\tilde{\gamma}_w$, i.e.,

$$\boldsymbol{\theta}_A(t) = \underset{\boldsymbol{\theta}_A}{\mathrm{argmax}}\ \mathrm{E}[\log p_{\tilde{\mathbf{Y}}(t)|\mathbf{X}_2}(\tilde{\mathbf{Y}}(t)|\mathbf{X}_2;\boldsymbol{\theta}_A,\tilde{\gamma}_w(t))|q_2^t(\mathbf{X}_2)], \tag{23a}$$

$$\begin{aligned}\tilde{\gamma}_w(t) = \underset{\tilde{\gamma}_w}{\mathrm{argmax}}\ &\mathrm{E}[\log p_{\tilde{\mathbf{Y}}(t)|\mathbf{X}_2}(\tilde{\mathbf{Y}}(t)|\mathbf{X}_2;\\ &\boldsymbol{\theta}_A(t-1),\tilde{\gamma}_w)|q_2^t(\mathbf{X}_2)].\end{aligned} \tag{23b}$$

Specifically, for the affine-linear model $\mathbf{A}(\boldsymbol{\theta}_A) = \mathbf{A}_0 + \sum_{i=1}^{G}\theta_{A,i}\mathbf{A}_i$, the detailed expression of estimating $\boldsymbol{\theta}_A$ and $\tilde{\gamma}_w$ are given by [33]

$$\boldsymbol{\theta}_A(t) = (\mathbf{H}(t))^{-1}\boldsymbol{\beta}^t, \tag{24a}$$

$$\begin{aligned}1/\tilde{\gamma}_w(t) = \frac{1}{ML}(&\|\tilde{\mathbf{Y}}(t)-\mathbf{A}(\boldsymbol{\theta}_A(t-1))\hat{\mathbf{X}}_2(t)\|_{\mathrm{F}}^2\\ &+\mathrm{tr}\{\mathbf{A}(\boldsymbol{\theta}_A(t-1))\boldsymbol{\Xi}_{\mathbf{X}}(t)\mathbf{A}^{\mathrm{T}}(\boldsymbol{\theta}_A(t-1))\}),\end{aligned} \tag{24b}$$

where

$$[\mathbf{H}(t)]_{ij} = \mathrm{tr}\{\mathbf{A}_j^{\mathrm{T}}\mathbf{A}_i(\boldsymbol{\Xi}_{\mathbf{X}}(t)+\hat{\mathbf{X}}_2(t)\hat{\mathbf{X}}_2^{\mathrm{T}}(t))\}, \tag{25a}$$

$$\begin{aligned}[\boldsymbol{\beta}^t]_i = &\,\mathrm{tr}\{\tilde{\mathbf{Y}}^{\mathrm{T}}(t)\mathbf{A}_i\hat{\mathbf{X}}_2(t)\}\\ &-\mathrm{tr}\{\mathbf{A}_0^{\mathrm{T}}\mathbf{A}_i(\boldsymbol{\Xi}_{\mathbf{X}}(t)+\hat{\mathbf{X}}_2(t)\hat{\mathbf{X}}_2^{\mathrm{T}}(t))\},\end{aligned} \tag{25b}$$

$\boldsymbol{\Xi}_{\mathbf{X}}(t) = \sum_{l=1}^{L}\boldsymbol{\Xi}_{\mathbf{x}_{2,l}}(t)$ and $\boldsymbol{\Xi}_{\mathbf{x}_{2,l}}(t)$ is given by (22).

The message $m^t_{\mathbf{x}_{2,l}\to\delta_x}(\mathbf{x}_{2,l})$ from the variable node $\mathbf{X}_2$ to the delta node $\delta(\mathbf{X}_1-\mathbf{X}_2)$ is calculated as

$$m^t_{\mathbf{x}_{2,l}\to\delta_x}(\mathbf{x}_{2,l}) \propto \frac{\mathrm{Proj}[q_2^t(\mathbf{x}_{2,l})]}{m^{t-1}_{\delta_x\to\mathbf{x}_{2,l}}(\mathbf{x}_{2,l})}, \tag{26}$$

where $q_2^t(\mathbf{x}_{2,l})$ is defined in (20). Projecting the posterior distribution $q_2^t(\mathbf{x}_{2,l})$ to the Gaussian distribution with scalar covariance matrix yields

$$\mathrm{Proj}[q_2^t(\mathbf{x}_{2,l})] \propto \mathcal{N}(\mathbf{x}_{2,l};\hat{\mathbf{x}}_{2,l}(t),\eta^{-1}_{2,l}(t)\mathbf{I}_N), \tag{27}$$

where

$$\eta^{-1}_{2,l}(t) = \mathrm{tr}(\boldsymbol{\Xi}_{\mathbf{x}_{2,l}}(t))/N. \tag{28}$$

Substituting (27) in (26), we obtain

$$m^t_{\mathbf{x}_{2,l}\to\delta_x}(\mathbf{x}_{2,l}) \propto \mathcal{N}(\mathbf{x}_{2,l};\mathbf{r}_{1,l}(t),\gamma^{-1}_{1,l}(t)\mathbf{I}_N), \tag{29}$$

where

$$\gamma_{1,l}(t) = \eta_{2,l}(t)-\gamma_{2,l}(t-1), \tag{30}$$

$$\mathbf{r}_{1,l}(t) = (\eta_{2,l}(t)\hat{\mathbf{x}}_{2,l}(t)-\gamma_{2,l}(t-1)\mathbf{r}_{2,l}(t-1))/\gamma_{1,l}(t). \tag{31}$$

According to the definition of the factor node $\delta(\mathbf{X}_1-\mathbf{X}_2)$, the message $m^t_{\delta_x\to\mathbf{x}_{1,l}}(\mathbf{x}_{1,l})$ satisfies

$$m^t_{\delta_x\to\mathbf{x}_{1,l}}(\mathbf{x}_{1,l}) = m^t_{\mathbf{x}_{2,l}\to\delta_x}(\mathbf{x}_{2,l})|_{\mathbf{x}_{2,l}=\mathbf{x}_{1,l}} \tag{32a}$$

$$= \mathcal{N}(\mathbf{x}_{1,l};\mathbf{r}_{1,l}(t),\gamma^{-1}_{1,l}(t)\mathbf{I}_N). \tag{32b}$$

Combining the prior $p(\mathbf{X}_1;\boldsymbol{\theta}_X)$ with $\boldsymbol{\theta}_X(t-1)$, the posterior mean and variances of $\mathbf{X}_1$ are calculated as

$$\eta^{-1}_{1,l}(t) = <\mathrm{Var}[\mathbf{x}_{1,l}|q^t_{1,l}(\mathbf{x}_{1,l})]>, \tag{33}$$

$$\hat{\mathbf{x}}_{1,l}(t) = \mathrm{E}[\mathbf{x}_{1,l}|q^t_{1,l}(\mathbf{x}_{1,l})], \tag{34}$$

where

$$q^t_{1,l}(\mathbf{x}_{1,l}) \propto p(\mathbf{x}_{1,l})\mathcal{N}(\mathbf{x}_{1,l};\mathbf{r}_{1,l}(t),\gamma^{-1}_{1,l}(t)\mathbf{I}_N). \tag{35}$$

To learn the unknown parameters $\boldsymbol{\theta}_X(t)$ and $\gamma_{1,l}(t)$, EM algorithm is applied in the inner iterations [33], i.e.,

$$\boldsymbol{\theta}_X(t) = \underset{\boldsymbol{\theta}_X}{\mathrm{argmax}}\ \mathrm{E}[\log p_{\mathbf{X}}(\mathbf{X}_1;\boldsymbol{\theta}_X)|q_1^t(\mathbf{X}_1)], \tag{36}$$

and

$$\gamma_{1,l}(t) = \underset{\gamma_{1,l}}{\operatorname{argmax}} \, \mathrm{E}[\log p(\mathbf{r}_{1,l}(t)|\mathbf{x}_l;\gamma_{1,l})|q_1^t(\mathbf{X}_1)] \quad (37)$$

$$= \left\{\frac{1}{N}\|\hat{\mathbf{x}}_{1,l}(t)-\mathbf{r}_{1,l}(t)\|^2 + \frac{1}{\eta_{1,l}(t)}\right\}^{-1}. \quad (38)$$

Now the message $m^t_{\mathbf{x}_{1,l}\to\delta_x}(\mathbf{x}_{1,l})$ from the variable node $\mathbf{X}_1$ to the factor node $\delta(\mathbf{X}_1-\mathbf{X}_2)$ is calculated as

$$m^t_{\mathbf{x}_{1,l}\to\delta_x}(\mathbf{x}_{1,l}) \propto \frac{\mathrm{Proj}[q^t_{1,l}(\mathbf{x}_{1,l})]}{m^t_{\delta_x\to\mathbf{x}_{1,l}}(\mathbf{x}_{1,l})} \quad (39a)$$

$$\propto \mathcal{N}(\mathbf{x}_{1,l};\mathbf{r}_{2,l}(t),\gamma_{2,l}^{-1}(t)\mathbf{I}_N), \quad (39b)$$

where $q^t_{1,l}(\mathbf{x}_{1,l})$ is (35), $\mathbf{r}_{2,l}(t)$ and $\gamma_{2,l}^{-1}(t)$ are given by

$$\gamma_{2,l}(t) = \eta_{1,l}(t)-\gamma_{1,l}(t), \quad (40)$$

$$\mathbf{r}_{2,l}(t) = (\eta_{1,l}(t)\mathbf{x}_{1,l}(t)-\gamma_{1,l}(t)\mathbf{r}_{1,l}(t))/\gamma_{2,l}(t). \quad (41)$$

According to the definition of the factor node $\delta(\mathbf{X}_1-\mathbf{X}_2)$, the message $m^t_{\delta_x\to\mathbf{x}_{2,l}}(\mathbf{x}_{2,l})$ from the factor node $\delta(\mathbf{X}_1-\mathbf{X}_2)$ to the variable node $\mathbf{X}_2$ is $m^t_{\delta_x\to\mathbf{x}_{2,l}}(\mathbf{x}_{2,l}) = \mathcal{N}(\mathbf{x}_{2,l};\mathbf{r}_{2,l}(t),\gamma_{2,l}^{-1}\mathbf{I}_N)$, which closes the BAd-VAMP algorithm.

### C. MESSAGES FROM BAD-VAMP MODULE TO MMSE MODULE

After performing BAd-VAMP for one or more iterations, we now focus on how to calculate the extrinsic message $m^t_{\delta_z\to\mathbf{z}_l}(\mathbf{z}_l)$ from the BAd-VAMP module to the component-wise MMSE module. Referring to the original factor graph shown in Fig. 1 (a), according to EP, the extrinsic message $m^t_{\delta_z\to\mathbf{z}_l}(\mathbf{z}_l)$ can be calculated as

$$m^t_{\delta_z\to\mathbf{z}_l}(\mathbf{z}_l) \propto \frac{\mathrm{Proj}\left[\int_{\mathbf{x}_l}\delta(\mathbf{z}_l-\mathbf{A}\mathbf{x}_l)m^t_{\mathbf{x}_l\to\delta_z}(\mathbf{x}_l)\mathrm{d}\mathbf{x}_l m^t_{\mathbf{z}_l\to\delta_z}(\mathbf{z}_l)\right]}{m^t_{\mathbf{z}_l\to\delta_z}(\mathbf{z}_l)} \quad (42a)$$

$$\triangleq \frac{\mathrm{Proj}[q^t_{\mathrm{A}}(\mathbf{z}_l)]}{m^t_{\mathbf{z}_l\to\delta_z}(\mathbf{z}_l)}. \quad (42b)$$

In BAd-VAMP, as shown in the above subsection B, we have already obtained the message $m^t_{\delta_x\to\mathbf{x}_{2,l}}(\mathbf{x}_{2,l})$ from the factor node $\delta(\mathbf{X}_1-\mathbf{X}_2)$ to the variable node $\mathbf{X}_2$. It can be seen from Fig. 2 that the message $m^t_{\mathbf{x}_l\to\delta_z}(\mathbf{x}_l)$ is the same as $m^t_{\delta_x\to\mathbf{x}_{2,l}}(\mathbf{x}_{2,l})$ so that $m^t_{\mathbf{x}_l\to\delta_z}(\mathbf{x}_l) = \mathcal{N}(\mathbf{x}_l;\mathbf{r}_{2,l}(t),\gamma_{2,l}^{-1}(t)\mathbf{I}_N)$. After some algebra, the posterior distribution $q^t_{\mathrm{A}}(\mathbf{z}_l)$ of $\mathbf{z}_l$ can be calculated to be Gaussian, i.e., $q^t_{\mathrm{A}}(\mathbf{z}_l) = \mathcal{N}(\mathbf{z}_l;\mathbf{z}^{\mathrm{post}}_{\mathrm{A},l}(t),\boldsymbol{\Xi}_{\mathbf{z}_l}(t))$, with the covariance matrix and mean vector being

$$\boldsymbol{\Xi}_{\mathbf{z}_l}(t) = \mathbf{A}(\boldsymbol{\theta}_A(t)) \left(\gamma_{2,l}(t)\mathbf{I}_N+\tilde{\gamma}_w(t)\mathbf{A}^{\mathrm{T}}(\boldsymbol{\theta}_A(t))\mathbf{A}(\boldsymbol{\theta}_A(t))\right)^{-1}\mathbf{A}^{\mathrm{T}}(\boldsymbol{\theta}_A(t)), \quad (43)$$

$$\mathbf{z}^{\mathrm{post}}_{\mathrm{A},l}(t) = \mathbf{A}(\boldsymbol{\theta}_A(t))\left(\gamma_{2,l}(t)\mathbf{I}_N+\tilde{\gamma}_w(t)\mathbf{A}^{\mathrm{T}}(\boldsymbol{\theta}_A(t))\mathbf{A}(\boldsymbol{\theta}_A(t))\right)^{-1} (\gamma_{2,l}(t)\mathbf{r}_{2,l}(t)+\tilde{\gamma}_w(t)\mathbf{A}^{\mathrm{T}}(\boldsymbol{\theta}_A(t))\tilde{\mathbf{y}}_l(t)). \quad (44)$$

Then, the posterior distribution $q^t_{\mathrm{A}}(\mathbf{z}_l)$ of $\mathbf{z}_l$ is further projected to Gaussian distribution with scalar covariance matrix, yielding

$$\mathrm{Proj}[q^t_{\mathrm{A}}(\mathbf{z}_l)] = \mathcal{N}(\mathbf{z}_l;\mathbf{z}^{\mathrm{post}}_{\mathrm{A},l}(t),v^{\mathrm{post}}_{\mathrm{A},l}(t)\mathbf{I}_M), \quad (45)$$

where

$$v^{\mathrm{post}}_{\mathrm{A},l}(t) = \mathrm{tr}(\boldsymbol{\Xi}_{\mathbf{z}_l}(t))/M, \quad (46)$$

Moreover, the posterior variances $\{v^{\mathrm{post}}_{\mathrm{A},l}(t)\}_{l=1}^L$ are averaged over the index $l$, which leads to

$$v^{\mathrm{post}}_{\mathrm{A}}(t) = \sum_{l=1}^{L} v^{\mathrm{post}}_{\mathrm{A},l}(t)/L, \quad (47)$$

by which $\mathrm{Proj}[q^t_{\mathrm{A}}(\mathbf{z}_l)]$ is approximated as $\mathrm{Proj}[q^t_{\mathrm{A}}(\mathbf{z}_l)] \approx \mathcal{N}(\mathbf{z}_l;\mathbf{z}^{\mathrm{post}}_{\mathrm{A},l}(t),v^{\mathrm{post}}_{\mathrm{A}}(t)\mathbf{I}_M)$. As a result, the message $m^{t+1}_{\delta_z\to\mathbf{z}_l}(\mathbf{z}_l)$ in (42) becomes

$$m^t_{\delta\to\mathbf{z}_l}(\mathbf{z}_l) \propto \mathcal{N}(\mathbf{z}_l;\mathbf{z}^{\mathrm{ext}}_{\mathrm{A},l}(t),v^{\mathrm{ext}}_{\mathrm{A}}(t)\mathbf{I}_M), \quad (48)$$

where

$$v^{\mathrm{ext}}_{\mathrm{A}}(t) = \left(\frac{1}{v^{\mathrm{post}}_{\mathrm{A}}(t)}-\frac{1}{v^{\mathrm{ext}}_{\mathrm{B}}(t)}\right)^{-1}, \quad (49)$$

$$\mathbf{z}^{\mathrm{ext}}_{\mathrm{A},l}(t) = v^{\mathrm{ext}}_{\mathrm{A}}(t)\left(\frac{\mathbf{z}^{\mathrm{post}}_{\mathrm{A},l}(t)}{v^{\mathrm{post}}_{\mathrm{A}}(t)}-\frac{\mathbf{z}^{\mathrm{ext}}_{\mathrm{B},l}(t)}{v^{\mathrm{ext}}_{\mathrm{B}}(t)}\right), \quad (50)$$

which closes the loop of the whole algorithm.

To sum up, the BAd-GVAMP algorithm can be summarized as Algorithm 1.

### D. RELATION OF BAD-GVAMP TO BAD-VAMP

The obtained BAd-GVAMP algorithm is an extension of BAd-VAMP from linear measurements to nonlinear measurements. Intuitively, as shown in Fig 1 (b), BAd-GVAMP iteratively reduces the original generalized bilinear recovery problem to a sequence of standard bilinear recovery problems. In each iteration of BAd-GVAMP, a pseudo linear measurement model is obtained and one iteration of BAd-VAMP is performed.[2] Note that the message passing schedule of the BAd-VAMP module within BAd-GVAMP is different from the original BAd-VAMP in [33]: in [33] variable de-noising is performed first and then LMMSE, while in the BAd-VAMP module of the proposed BAd-GVAMP, LMMSE is performed first and then variable de-noising. It is worth noting that in the special case of linear measurements, i.e., when $p(\mathbf{Y}|\mathbf{Z})$ is Gaussian, i.e., $p(\mathbf{y}_l|\mathbf{z}_l) = \mathcal{N}(\mathbf{y}_l;\mathbf{z}_l,\gamma_w^{-1}\mathbf{I}_M)$, the BAd-GVAMP reduces to BAd-VAMP precisely since in such case the extrinsic means $v^{\mathrm{ext}}_{\mathrm{B}}(t)$ and variances $\mathbf{z}^{\mathrm{ext}}_{\mathrm{B},l}(t)$ from the MMSE module always satisfy $v^{\mathrm{ext}}_{\mathrm{B}}(t) = \gamma_w^{-1}, \forall t, \mathbf{z}^{\mathrm{ext}}_{\mathrm{B},l}(t) = \mathbf{y}_l, \forall t,\ l$. Thus BAd-GVAMP is consistent with BAd-VAMP under Gaussian output transform.

[2] It is also possible to perform multiple iterations of the BAd-VAMP in a whole single iteration of BAd-GVAMP.

**Algorithm 1** Bilinear Adaptive Generalized VAMP (BAd-GVAMP)

**Initialization**: $\mathbf{z}_{\mathrm{A},l}^{\mathrm{ext}}(0)$, $v_{\mathrm{A}}^{\mathrm{ext}}(0)$, $\mathbf{r}_{2,l}(0)$, $\gamma_{2,l}(0)$ $\boldsymbol{\theta}_X(0)$, $\boldsymbol{\theta}_A(0)$ and $\boldsymbol{\theta}_Y(0)$.
**for** $t = 1, \cdots, T_{\mathrm{outer}}$ **do**
  Compute the posterior mean and variance of $\mathbf{Z}$ as $\mathbf{Z}_{\mathrm{B}}^{\mathrm{post}}(t)$ (10), $v_{\mathrm{B}}^{\mathrm{post}}(t)$ (12).
  Compute the extrinsic mean and variance of $\mathbf{z}$ as $v_{\mathrm{B}}^{\mathrm{ext}}(t)$ (15), $\mathbf{z}_{\mathrm{B},l}^{\mathrm{ext}}(t)$ (16), and set $\tilde{\mathbf{y}}_l(t) \triangleq \mathbf{z}_{\mathrm{B},l}^{\mathrm{ext}}(t)$ and $\tilde{\gamma}_w(t) \triangleq 1/v_{\mathrm{B}}^{\mathrm{ext}}(t)$ in (18).
  **for** $\tau = 1, \cdots, T_{\mathrm{inner},1}$ **do**
    Perform the LMMSE estimate of $\mathbf{x}_l$, i.e., the posterior means $\hat{\mathbf{x}}_{2,l}(t)$ and covariance matrix $\boldsymbol{\Xi}_{\mathbf{x}_{2,l}}(t)$ shown in (21) and (22).
    Update $\boldsymbol{\theta}_A(t)$ (23a) and $\tilde{\gamma}_w(t)$ (23b).
  **end for**
  Calculate $\hat{\mathbf{x}}_{2,l}(t)$ (21) and $\eta_{2,l}(t)$ (28).
  Calculate $\mathbf{r}_{1,l}(t)$ (31) and $\gamma_{1,l}(t)$ (30).
  **for** $\tau = 1, \cdots, T_{\mathrm{inner},2}$ **do**
    Perform the input denoising operation to obtain the posterior means $\hat{\mathbf{x}}_{1,l}(t)$ (34) and variances $\eta_{1,l}^{-1}(t)$ (33).
    Update $\boldsymbol{\theta}_X$ (36).
    Calculate $\mathbf{r}_{2,l}(t)$ (41) and $\gamma_{2,l}(t)$ (40).
  **end for**
  Calculate the posterior means $\mathbf{z}_{\mathrm{A},l}^{\mathrm{post}}(t)$ (44) and variance $v_{\mathrm{A}}^{\mathrm{post}}(t)$ (47).
  Calculate the extrinsic means $\mathbf{z}_{\mathrm{A},l}^{\mathrm{ext}}(t)$ (50) and variance $v_{\mathrm{A}}^{\mathrm{ext}}(t)$ (49).
  Update $\boldsymbol{\theta}_Y(t)$ as (17).
**end for**
Return $\hat{\mathbf{X}}$ and $\hat{\boldsymbol{\Theta}}$.

## IV. NUMERICAL SIMULATION

In this section, numerical experiments are conducted to investigate the performance of the proposed BAd-GVAMP algorithm.[3] In particular, quantized measurements are considered. As for the inner iteration of BAd-GVAMP in Algorithm 1, we set $T_{\mathrm{inner},1} = 1$ and $T_{\mathrm{inner},2} = 2$, which are the same as [33]. In addition, several strategies are proposed to enhance the robustness.

- Damping: We perform damping for variables $\mathbf{r}_{1,l}(t)$, $\gamma_{1,l}(t)$, $\gamma_{2,l}(t)$, $\mathbf{r}_{2,l}(t)$. The damping factor is set as 0.8.
- Clipping precisions: Sometimes the variances $v_{\mathrm{A}}^{\mathrm{ext}}(t)$ and $v_{\mathrm{B}}^{\mathrm{ext}}(t)$ or precisions $\{\gamma_{1,l}(t), \gamma_{2,l}(t)\}_{l=1}^{L}$ can be either negative or too large, we suggest to clip the precisions and variances to the interval $[\gamma_{\min}, \gamma_{\max}]$. In our simulation, we set $\gamma_{\min} = 10^{-8}$ and $\gamma_{\max} = 10^{12}$.

As for the quantizer, let $Q(\cdot)$ denote a quantization operation. For the quantizer with bit-depth $N_{\mathrm{b}}$, uniform quantization is adopted with the thresholds being $\tau_i = Z_{\min} + i\Delta$, $i = 1, 2, \cdots, 2^{N_{\mathrm{b}}} - 1$, where $\Delta = (Z_{\max} - Z_{\min})/2^{N_{\mathrm{b}}}$, $Z_{\min}$ and $Z_{\max}$ denote the minimum and maximum value of $\mathbf{Z}$. With such uniform quantization, the measurements $\mathbf{Y}$ becomes $\mathbf{Y} = Q(\mathbf{Z} + \mathbf{W})$ where $W_{ij} \sim \mathcal{N}(W_{ij}; 0, \gamma_w^{-1})$. In this setting, $\theta_Y = \{\gamma_w\}$ and is updated approximately as $\theta_Y(t) = \tilde{\gamma}_w(t)$.

### A. CS WITH MATRIX UNCERTAINTY FROM QUANTIZED MEASUREMENTS

Consider the problem $\mathbf{y} = Q(\mathbf{A}(\mathbf{b})\mathbf{c} + \mathbf{w})$, where $\mathbf{w} \sim \mathcal{N}(\mathbf{w}; \mathbf{0}, \mathbf{I}_M/\gamma_w)$. Here $\mathbf{A}(\mathbf{b}) = \mathbf{A}_0 + \sum_{i=1}^{G} b_i \mathbf{A}_i$, where $\{\mathbf{A}_i\}_{i=0}^{G} \in \mathbb{R}^{M\times N}$ are known, $\mathbf{b}$ are the unknown uncertainty parameters. We set $\gamma_w$ according to the SNR (dB) defined as $\mathrm{SNR} \triangleq 10\log \mathrm{E}\|\mathbf{Ac}\|^2/\mathrm{E}\|\mathbf{w}\|^2 = 40$ dB. The uncertainty parameters $\mathbf{b}$ are drawn from $\mathcal{N}(\mathbf{0}, \mathbf{I}_G)$, and $\mathbf{c}$ is generated with uniformly random support with $K$ nonzero elements from $\mathcal{N}(\mathbf{0}, \mathbf{I}_K)$. We set $G = 10$ and $K = 10$. For one-bit quantization, the debiased normalized mean square error (dNMSE) $\min_{\xi_c} 10\log(\|\mathbf{c} - \xi_c\hat{\mathbf{c}}\|_2/\|\mathbf{c}\|_2)$ and $\min_{\xi_b} 10\log(\|\mathbf{b} - \xi_b\hat{\mathbf{b}}\|_2/\|\mathbf{b}\|_2)$ are used to characterize the performance. While for multi-bit quantization settings, the NMSEs $10\log(\|\mathbf{c} - \hat{\mathbf{c}}\|_2/\|\mathbf{c}\|_2)$ and $10\log(\|\mathbf{b} - \hat{\mathbf{b}}\|_2/\|\mathbf{b}\|_2)$ are used instead. The elements of $\mathbf{A}_0$ are drawn i.i.d. from $\mathcal{N}(0, 20)$. To provide a benchmark performance of the BAd-GVAMP algorithm, we also evaluate the oracle performance by assuming known $\mathbf{b}$ or $\mathbf{c}$.

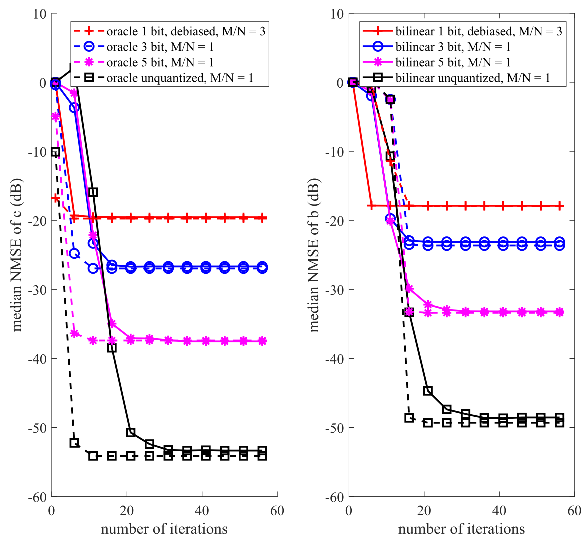



**FIGURE 3.** Matrix uncertainty scenario: Median NMSE (over 50 Monte Carlo (MC) trials) on signal c and uncertainty parameters b versus the number of iterations.

For the first experiment, we demonstrate the convergence performance of the BAd-GVAMP algorithm. We set the sampling ratio of one-bit quantization as $M/N = 3$, while for 3 bit, 5 bit and unquantized case, we set $M/N = 1$. From Fig. 3, it can be seen that the BAd-GVAMP algorithm converges after $20 \sim 30$ iterations, and its performance is close to the oracle scenario. For the second experiment, the performance versus the sampling rate $M/N$ is investigated.

[3] BAdGVAMP code: https://github.com/mengxiangming/BAd_GVAMP_code

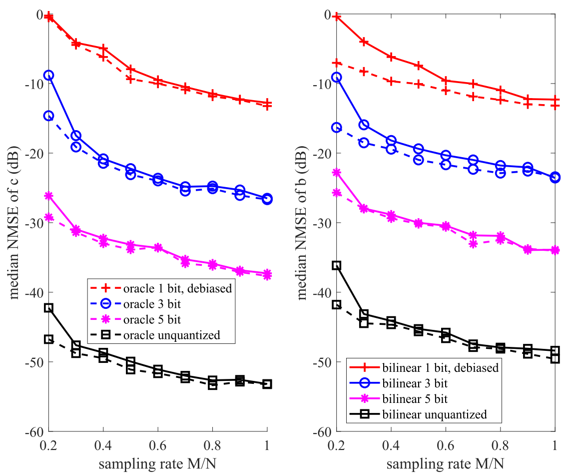



**FIGURE 4.** Matrix uncertainty scenario: Median NMSE (over 50 MC trials) on signal **c** and uncertainty parameters **b** versus sampling rate $M/N$.

As shown in Fig. 4, the performance under one-bit quantization is poor for both bilinear and oracle scenarios under low sampling rate. As the sampling rate $M/N$ increases, the performances of BAd-GVAMP algorithm improve. In addition, the proposed algorithm gives near oracle performance for the tested range of $M/N$.

### B. SELF-CALIBRATION FROM QUANTIZED MEASUREMENTS

Here we investigate the self-calibration from quantized measurements which aims to recover the $K$-sparse signal vector $\mathbf{c}$ and the calibration parameters $\mathbf{b}$ from

$$\begin{aligned}\mathbf{y} &= Q(\text{diag}(\mathbf{H}\mathbf{b})\boldsymbol{\Psi}\mathbf{c} + \mathbf{w}) \\ &= Q\left(\left[\sum_{i=1}^{G} b_i \text{diag}(\mathbf{h}_i)\boldsymbol{\Psi}\right]\mathbf{c} + \mathbf{w}\right). \end{aligned} \tag{51}$$

with known $\mathbf{H} \in \mathbb{R}^{M\times G}$ and $\boldsymbol{\Psi} \in \mathbb{R}^{M\times N}$. The normalized MSE is defined as NMSE $\triangleq 10\log(\|\hat{\mathbf{b}}\hat{\mathbf{c}}^{\mathrm{T}} - \mathbf{b}\mathbf{c}^{\mathrm{T}}\|_{\mathrm{F}}^2/\|\mathbf{b}\mathbf{c}^{\mathrm{T}}\|_{\mathrm{F}}^2)$.

The simulation parameters are set as follows: $K = 10$, $G = 8$, $M = 128$ and SNR $= 40$ dB. Here $\mathbf{H}$ is constructed using $Q$ randomly selected columns of the Hadamard matrix, the elements of $\mathbf{b}$ and $\boldsymbol{\Psi}$ are i.i.d. drawn from $\mathcal{N}(0, 1)$, and $\mathbf{c}$ is generated with $K$ nonzero elements i.i.d. drawn from $\mathcal{N}(0, 1)$. The NMSE versus the sampling rate $M/N$ are presented in Fig. 5 and Fig. 6 for one-bit and multi-bit quantization settings, respectively. It can be seen that as the sampling rate increases, the median NMSE decreases. Also, the reconstruction performance improves as the bit-depth increases.

### C. STRUCTURED DICTIONARY LEARNING FROM QUANTIZED MEASUREMENTS

The goal of dictionary learning is to find a dictionary matrix $\mathbf{A} \in \mathbb{R}^{M\times N}$ and a sparse matrix $\mathbf{X} \in \mathbb{R}^{N\times L}$ such that $\mathbf{Y} \approx \mathbf{AX}$ for a given matrix $\mathbf{Y} \in \mathbb{R}^{M\times L}$. We consider structured dictionary $\mathbf{A}$ such that $\mathbf{A} = \sum_{i=1}^{G} b_i\mathbf{A}_i$ with known

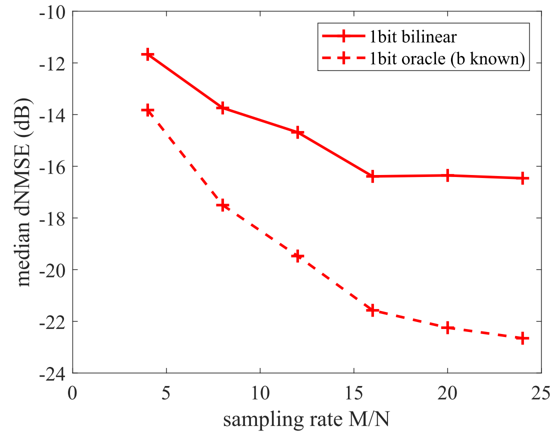



**FIGURE 5.** Self-Calibration scenario: median NMSE (over 50 MC trials) on signal **c** and uncertainty parameters **b** versus sampling rate $M/N$ under one-bit quantization.

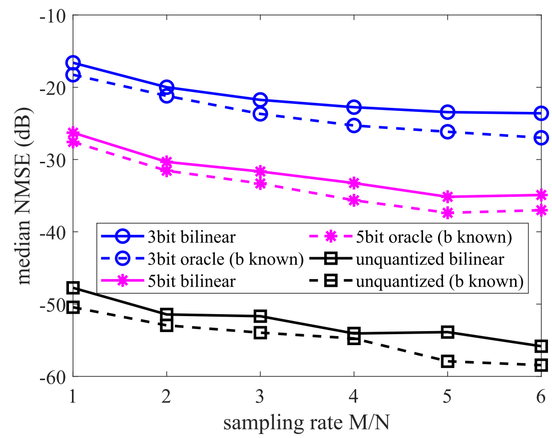



**FIGURE 6.** Self-Calibration scenario: median NMSE (over 50 MC trials) on signal **c** and uncertainty parameters **b** versus sampling rate $M/N$ under multi-bit quantization.

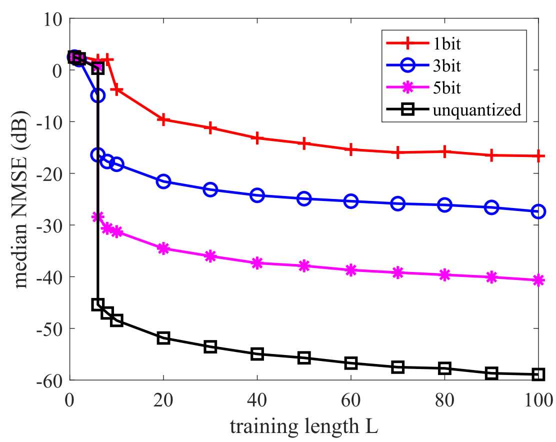



**FIGURE 7.** Structured dictionary learning scenario: Median NMSE (over 50 MC trials) on dictionary **A** versus the training length $L$.

$\{\mathbf{A}_i\}_{i=1}^{G}$, where the elements of $\mathbf{A}_i$ and $b_i$ are i.i.d. drawn from $\mathcal{N}(0, 1)$ with $G = M = N = 64$ in the structured case. Then the measurements are obtained as $\mathbf{Y} = Q(\mathbf{AX}+\mathbf{W})$ such that each column of $\mathbf{X}$ is $K$ sparse. We set SNR $= 40$ dB where SNR is defined as $\text{SNR} \triangleq 10\log \text{E}[\|\mathbf{AX}\|_\text{F}^2]/\text{E}[\|\mathbf{W}\|_\text{F}^2]$. Since the dictionary can not be recovered exactly, the NMSE for the structured case is defined as [33]

$$\text{NMSE}(\hat{\mathbf{A}}) \triangleq \min_{\lambda\in\mathbb{R}} 10\log \frac{\|\mathbf{A}-\lambda\hat{\mathbf{A}}\|_\text{F}^2}{\|\mathbf{A}\|_\text{F}^2}.$$

The median NMSE versus the training length $L$ is shown in Fig. 7. It can be seen that as the training length increases, the NMSE decreases. In addition, the structured dictionary can be learned from one-bit measurements.

## V. CONCLUSION

Many problems in science and engineering can be formulated as the generalized bilinear inference problem. To address this problem, this paper proposed a novel algorithm called Bilinear Adaptive Generalized Vector Approximate Message Passing (BAd-GVAMP), which extends the recently proposed BAd-VAMP from linear measurements to nonlinear measurements. In the special case of linear measurements, BAd-GVAMP reduces to the BAd-VAMP. Numerical simulations are conducted for compressed sensing with matrix uncertainty, self-calibration as well as structured dictionary learning from quantized measurements, which demonstrates the effectiveness of the proposed algorithm.

## REFERENCES

[1] H. Zhu, G. Leus, and G. B. Giannakis, "Sparsity-cognizant total least-squares for perturbed compressive sampling," *IEEE Trans. Signal Process.*, vol. 59, no. 5, pp. 2002–2016, May 2011.

[2] M. Rosenbaum and A. B. Tsybakov, "Sparse recovery under matrix uncertainty," *Ann. Statist.*, vol. 38, no. 5, pp. 2620–2651, 2010.

[3] J. T. Parker, V. Cevher, and P. Schniter, "Compressive sensing under matrix uncertainties: An approximate message passing approach," in *Proc. ASILOMAR*, Pacific Grove, CA, USA, Nov. 2011, pp. 804–808.

[4] F. Krzakala, M. Mézard, and L. Zdeborová, "Compressed sensing under matrix uncertainty: Optimum thresholds and robust approximate message passing," in *Proc. ICASSP*, May 2013, pp. 5519–5523.

[5] E. J. Candès and Y. Plan, "Matrix completion with noise," *Proc. IEEE*, vol. 98, no. 6, pp. 925–936, Jun. 2010.

[6] R. Matsushita and T. Tanaka, "Low-rank matrix reconstruction and clustering via approximate message passing," in *Proc. Neural Inf. Process. Syst. Conf.*, 2013, pp. 917–925.

[7] T. Lesieur, F. Krzakala, and L. Zdeborová, "Constrained low-rank matrix estimation: Phase transitions, approximate message passing and applications," *J. Stat. Mech., Theory Exp.*, vol. 2017, no. 7, p. 073403, 2017.

[8] E. J. Candès, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?" *J. ACM*, vol. 58, no. 3, p. 11, May 2011.

[9] J. T. 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.

[10] J. T. 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.

[11] G. K. Kaleh and R. Vallet, "Joint parameter estimation and symbol detection for linear or nonlinear unknown channels," *IEEE Trans. Commun.*, vol. 42, no. 7, pp. 2406–2413, Jul. 1994.

[12] S. Wu, L. Kuang, Z. Ni, D. Huang, Q. Guo, and J. Lu, "Message-passing receiver for joint channel estimation and decoding in 3D massive MIMO-OFDM systems," *IEEE Trans. Wireless Commun.*, vol. 15, no. 12, pp. 8122–8138, Dec. 2016.

[13] C.-K. Wen, C.-J. Wang, S. Jin, K.-K. Wong, and P. Ting, "Bayes-optimal joint channel-and-data estimation for massive MIMO with low-precision ADCs," *IEEE Trans. Signal Process.*, vol. 64, no. 10, pp. 2541–2556, May 2016.

[14] L. Jacques, J. L. Laska, P. T. Boufounos, and R. G. Baraniuk, "Robust 1-bit compressive sensing via binary stable embeddings of sparse vectors," *IEEE Trans. Inf. Theory*, vol. 59, no. 4, pp. 2082–2102, Apr. 2013.

[15] E. G. Larsson, O. Edfors, F. Tufvesson, and T. L. Marzetta, "Massive MIMO for next generation wireless systems," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 186–195, Feb. 2014.

[16] J. Mo, P. Schniter, and R. W. Heath, Jr., "Channel estimation in broadband millimeter wave MIMO systems with few-bit ADCs," *IEEE Trans. Signal Process.*, vol. 66, no. 5, pp. 1141–1154, Mar. 2018.

[17] S. Ling and T. Strohmer, "Self-calibration and biconvex compressive sensing," *Inverse Problems*, vol. 31, no. 11, p. 115002, 2015.

[18] Ç. Bilen, G. Puy, R. Gribonval, and L. Daudet, "Convex optimization approaches for blind sensor calibration using sparsity," *IEEE Trans. Signal Process.*, vol. 62, no. 18, pp. 4847–4856, Sep. 2014.

[19] S. D. Babacan, M. Luessi, R. Molina, and A. K. Katsaggelos, "Sparse Bayesian methods for low-rank matrix estimation," *IEEE Trans. Signal Process.*, vol. 60, no. 8, pp. 3964–3977, Aug. 2012.

[20] J. T. Parker and P. Schniter, "Parametric bilinear generalized approximate message passing," *IEEE J. Sel. Topics Signal Process.*, vol. 10, no. 4, pp. 795–808, Jun. 2016.

[21] Y. Kabashima, F. Krzakala, M. Mézard, A. Sakata, and L. Zdeborová, "Phase transitions and sample complexity in Bayes-optimal matrix factorization," *IEEE Trans. Inf. Theory*, vol. 62, no. 7, pp. 4228–4265, Jul. 2016.

[22] 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.

[23] M. Bayati, M. Lelarge, and A. Montanari, "Universality in polytope phase transitions and message passing algorithms," *Ann. Appl. Probab.*, vol. 25, no. 2, pp. 753–822, 2015.

[24] S. Rangan, P. Schniter, and A. K. Fletcher. (2016). "Vector approximate message passing." [Online]. Available: https://arxiv.org/abs/1610.03082

[25] J. Ma and L. Ping, "Orthogonal AMP," *IEEE Access*, vol. 5, pp. 2020–2033, 2017.

[26] S. Rangan, "Generalized approximate message passing for estimation with random linear mixing," in *Proc. IEEE Int. Symp. Inf. Theory*, Jul. 2011, pp. 2168–2172.

[27] S. Rangan. (2012). "Generalized approximate message passing for estimation with random linear mixing." [Online]. Available: https://arxiv.org/abs/1010.5141v2

[28] P. Schniter, S. Rangan, and A. K. Fletcher, "Vector approximate message passing for the generalized linear model," in *Proc. 50th Asilomar Conf. Signals, Syst. Comput.*, Nov. 2016, pp. 1525–1529.

[29] H. He, C.-K. Wen, and S. Jin, "Generalized expectation consistent signal recovery for nonlinear measurements," in *Proc. IEEE Int. Symp. Inf. Theory*, Jun. 2017, pp. 2333–2337.

[30] X. Meng, S. Wu, and J. Zhu, "A unified Bayesian inference framework for generalized linear models," *IEEE Signal Process. Lett.*, vol. 25, no. 3, pp. 398–402, Mar. 2018.

[31] A. K. Fletcher, S. Rangan, S. Sarkar, and P. Schniter. (2018). "Plug-in estimation in high-dimensional linear inverse problems: A rigorous analysis." [Online]. Available: https://arxiv.org/abs/1806.10466

[32] J. Zhu, Q. Zhang, X. Meng, and Z. Xu. (2018). "Vector approximate message passing algorithm for structured perturbed sensing matrix." [Online]. Available: https://arxiv.org/abs/1808.08579

[33] S. Sarkar, A. K. Fletcher, S. Rangan, and P. Schniter. (2018). "Bilinear recovery using adaptive vector-AMP." [Online]. Available: https://arxiv.org/pdf/1809.00024.pdf

[34] A. K. Fletcher, M. Sahraee-Ardakan, S. Rangan, and P. Schniter, "Rigorous dynamics and consistent estimation in arbitrarily conditioned linear systems," in *Proc. Neural Inf. Process. Syst. Conf.*, 2017, pp. 2542–2551.

[35] U. S. 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.

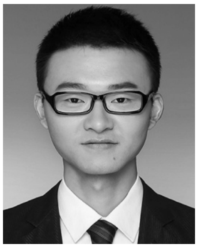
[36] T. P. Minka, "A family of algorithms for approximate Bayesian inference," Ph.D. dissertation, Dept. Elect. Eng. Comput. Sci., Massachusetts Inst. Technol., Cambridge, MA, USA, 2001.

[37] J. P. Vila and P. Schniter, "Expectation-maximization Gaussian-mixture approximate message passing," *IEEE Trans. Signal Process.*, vol. 61, no. 19, pp. 4658–4672, Oct. 2013.

[38] X. Meng, S. Wu, L. Kuang, and J. Lu, "An expectation propagation perspective on approximate message passing," *IEEE Signal Process. Lett.*, vol. 22, no. 8, pp. 1194–1197, Aug. 2015.

[39] Q. Zou, H. Zhang, C.-K. Wen, S. Jin, and R. Yu, "Concise derivation for generalized approximate message passing using expectation propagation," *IEEE Signal Process. Lett.*, vol. 25, no. 12, pp. 1835–1839, Dec. 2018.
[40] S. Wu, L. Kuang, Z. Ni, J. Lu, D. Huang, and Q. Guo, "Low-complexity iterative detection for large-scale multiuser MIMO-OFDM systems using approximate message passing," *IEEE J. Sel. Topics Signal Process.*, vol. 8, no. 5, pp. 902–915, May 2014.
[41] M. Opper and O. Winther, "Expectation consistent approximate inference," *J. Mach. Learn. Res.*, vol. 6, pp. 2177–2204, Dec. 2005.

**XIANGMING MENG** received the B.E. degree from Xidian University, Xi'an, China, in 2011, and the Ph.D. degree in electronic engineering from Tsinghua University, Beijing, China, in 2016. He is currently a research engineer with Huawei Technologies, Co., Ltd., Shanghai, China. His research interests include signal processing, wireless communications, and Bayesian machine learning.



**JIANG ZHU** received the B.E. degree in electronic science and technology from Harbin Engineering University, Harbin, in 2011, and the Ph.D. degree in information and communication engineering from Tsinghua University, Beijing, in 2016. Since 2016, he has been a lecturer with the Ocean College, Zhejiang University. His current research interests include statistical signal processing, Bayesian methods, and unlabeled sensing.