Approximate Message Passing With Consistent Parameter Estimation and Applications to Sparse Learning

Ulugbek S. Kamilov, *Student Member, IEEE*, Sundeep Rangan, *Member, IEEE*, Alyson K. Fletcher, *Member, IEEE*, and Michael Unser, *Fellow, IEEE*

***Abstract*— We consider the estimation of an independent and identically distributed (i.i.d.) (possibly non-Gaussian) vector $\mathbf{x} \in \mathbb{R}^n$ from measurements $\mathbf{y} \in \mathbb{R}^m$ obtained by a general cascade model consisting of a known linear transform followed by a probabilistic componentwise (possibly nonlinear) measurement channel. A novel method, called adaptive generalized approximate message passing (adaptive GAMP) is presented. It enables the joint learning of the statistics of the prior and measurement channel along with estimation of the unknown vector x. We prove that, for large i.i.d. Gaussian transform matrices, the asymptotic componentwise behavior of the adaptive GAMP is predicted by a simple set of scalar state evolution equations. In addition, we show that the adaptive GAMP yields asymptotically consistent parameter estimates, when a certain maximum-likelihood estimation can be performed in each step. This implies that the algorithm achieves a reconstruction quality equivalent to the oracle algorithm that knows the correct parameter values. Remarkably, this result applies to essentially arbitrary parametrizations of the unknown distributions, including nonlinear and non-Gaussian ones. The adaptive GAMP methodology thus provides a systematic, general and computationally efficient method applicable to a large range of linear–nonlinear models with provable guarantees.**

***Index Terms*— Approximate message passing, parameter estimation, belief propagation, compressive sensing.**

I. Introduction

CONSIDER the estimation of a random vector $\mathbf{x} \in \mathbb{R}^n$ from the measurement model illustrated in Fig. 1. The random vector $\mathbf{x}$, which is assumed to have independent and identically distributed (i.i.d.) components $x_j \sim P_X$, is passed through a known linear transform that

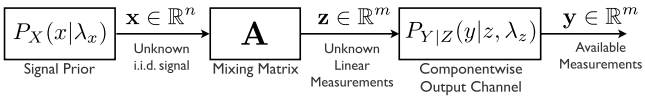

Fig. 1. Measurement model considered in this work.

outputs $\mathbf{z} = \mathbf{A}\mathbf{x}$. The components of $\mathbf{y} \in \mathbb{R}^m$ are generated by the component-wise transfer function $P_{Y|Z}$. This work addresses the problem of the estimation of $\mathbf{x}$ when the distributions P_X and $P_{Y|Z}$ have finite number of unknown parameters, λ_x and λ_z, that must be learned during the process.

Such joint-estimation and learning problems with linear transforms and component-wise nonlinearities arise in a range of applications, including empirical Bayesian approaches to inverse problems in signal processing, linear regression, and classification [1], [2]. It is equally relevant for Bayesian compressed sensing for the estimation of sparse vectors $\mathbf{x}$ from underdetermined measurements [3]–[5]. Also, since the parameters in the output transfer function $P_{Y|Z}$ can model unknown nonlinearities, this problem formulation can be applied to the identification of linear-nonlinear cascade models of dynamical systems, in particular for neural spike responses [6]–[8].

When the distributions P_X and $P_{Y|Z}$ are known, there are a number of estimation methods available. In recent years, there has been significant interest in so-called approximate message passing (AMP) and related methods based on Gaussian approximations of loopy belief propagation (LBP) [9]–[18]. These methods originate from CDMA multiuser detection problems [9]–[11] and have received considerable recent attention in the context of compressed sensing [13]–[19]. A survey article is available in [20]. The Gaussian approximations used in AMP are also closely related to expectation propagation techniques [21], [22], but with additional simplifications that exploit the linear coupling between the variables $\mathbf{x}$ and $\mathbf{z}$. The key benefits of AMP methods are their computational simplicity, their broad range of application, and, for certain large random $\mathbf{A}$, their exact asymptotic performance characterizations with testable conditions for optimality [11], [12], [16], [17]. This paper considers the so-called generalized AMP (GAMP) method [18], [23] that extends the algorithm in [13] to arbitrary output distributions $P_{Y|Z}$.

Although the current formulation of AMP and GAMP methods is attractive conceptually, in practice, one rarely knows the prior and noise distributions exactly. The expectation-

Manuscript received February 3, 2013; revised October 7, 2013; accepted February 14, 2014. Date of publication March 19, 2014; date of current version April 17, 2014. U. S. Kamilov and M. Unser were supported by the European Research Council under the European Union's Seventh Framework Programme (FP7/2007-2013)/ERC grant agreement n°267439. S. Rangan was supported by the National Science Foundation under Grant 1116589. This paper was presented at the 2012 25th Annual Conference on Neural Information Processing Systems.

U. S. Kamilov and M. Unser are with Biomedical Imaging Group, École Polytechnique Fédérale de Lausanne, Lausanne 1015, Switzerland (e-mail: ulugbek.kamilov@epfl.ch; michael.unser@epfl.ch).

S. Rangan is with Polytechnic Institute of New York University, Brooklyn, NY 11201 USA (e-mail: srangan@poly.edu).

A. K. Fletcher is with the Department of Electrical Engineering, University of California, Santa Cruz, CA 95064 USA (e-mail: afletcher@ucsc.edu).

Communicated by O. Milenkovic, Associate Editor for Coding Theory.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIT.2014.2309005

maximization-based (EM) approach [24], [25] overcomes this limitation by jointly learning the parameters (λ_x, λ_z) along with the estimation of the vector $\mathbf{x}$. EM-GAMP inspired our preliminary work [26]. While simulations indicate excellent performance, no analysis of these methods is available in the literature. This work provides a unifying analytic framework for such AMP-based joint estimation and learning methods. The main contributions of this paper are as follows:

- The generalization of the GAMP method of [18] to a class of algorithms we call *adaptive GAMP* that enable the joint estimation of the parameters λ_x and λ_z along with vector $\mathbf{x}$. The methods are computationally fast and general. In addition, adaptive GAMP includes the EM-GAMP algorithms of [24], [25], [27], and [28] as special cases.
- The exact characterization of the asymptotic behavior of adaptive GAMP. We show that, similar to the analysis of the AMP and GAMP algorithms in [11], [12], and [16]–[18], the component-wise asymptotic behavior of adaptive GAMP can be described exactly by a simple scalar *state-evolution* (SE) equations.
- The demonstration of the asymptotic consistency of adaptive GAMP with maximum-likelihood (ML) parameter estimation. We show that, when the ML parameter estimation is computed exactly, the estimated parameters converge to the true values and the performance of adaptive GAMP asymptotically coincides with the performance of the oracle GAMP algorithm that knows the correct parameter values. Remarkably, this result applies to essentially arbitrary parameterizations of the unknown distributions P_X and $P_{Y|Z}$, thus enabling provably consistent estimation on non-convex and nonlinear problems.
- The experimental evaluation of the algorithm for the problems of learning sparse priors in compressed sensing and of identification of linear-nonlinear cascade models in neural spiking processes. Our simulations illustrate the performance gain of adaptive GAMP and its asymptotic consistency.

A. Related Literature

The adaptive GAMP method proposed here can be seen as a generalization of the EM methods in [24], [25], [27], and [28]. In [24] and [25], the prior P_X is described by a generic L-term Gaussian mixture (GM) whose parameters are identified by an EM procedure [29]. The "expectation" step or E-step is performed by GAMP, which can approximately determine the marginal posterior distributions of the components x_j given the observations $\mathbf{y}$ and the current parameter estimates of the GM distribution P_X. A related EM-GAMP algorithm has also appeared in [27] and [28] for the case of certain sparse priors and AWGN outputs. Simulations in [24] and [25] show remarkably good performance and computational speed for EM-GAMP over a wide class of distributions, particularly in the context of compressed sensing. Also, using arguments from statistical physics, Krzakala *et al.* [27], [28] present SE equations for the joint evolution of the parameters and vector estimates and confirm them numerically.

As discussed in Section III-B, EM-GAMP is a special case of adaptive GAMP with a particular choice of the adaptation functions. Therefore, one contribution of this paper is to provide a rigorous theoretical justification of the EM-GAMP methodology. In particular, Theorem 2 in the current work provides a rigorous justification of the SE analysis in [27] and [28] along with extensions to a broader class of input and output channels and adaptation methods. However, the methodology in [27] and [28] is in some ways more general, in the sense that it can also study "seeded" or "spatially-coupled" matrices as proposed in [27], [28], and [30]. An interesting open question is to know if the analysis methods in this paper can be extended to these scenarios as well.

An alternate method for joint learning and estimation has been presented in [31], which assumes that the distributions on the source and output channels are themselves described by graphical models with the parameters λ_x and λ_z appearing as unknown variables. The method in [31], called hybrid-GAMP, iteratively combines standard loopy BP with AMP methods. One avenue of future work is to see if the methodology in this paper can be applied to analyze the hybrid-GAMP methods as well.

Finally, it should be pointed out that, while the simultaneous recovery of unknown parameters is appealing conceptually, it is not a strict requirement. An alternate solution to the problem is to assume that the signal belongs to a known class of distributions and to minimize the maximal mean-squared error (MSE) for the class. This minimax approach [32] was proposed for the AMP recovery of sparse signals in [13]. Although minimax yields estimators that are uniformly good over the entire class of distributions, there may be a significant gap between the MSE achieved by the minimax approach and the oracle algorithm that knows the distribution exactly. Indeed, reducing this gap was the main justification of the EM-GAMP methods in [24] and [25]. Due to its asymptotic consistency with ML parameter estimation, adaptive GAMP provably achieves the performance of the oracle algorithm.

B. Outline of the Paper

The paper is organized as follows: In Section II, we review the non-adaptive GAMP and the corresponding SE equations. In Section III, we present adaptive GAMP and describe ML parameter learning. In Section IV, we provide the main theorems that characterize the asymptotic performance of adaptive GAMP and demonstrate its consistency. A key requirement for consistency are certain identifiability conditions — these are discussed in Section V. In Section VI, we provide numerical experiments that illustrate the applicability of the method. Section VII concludes the paper. A preliminary version of this work has appeared in [26]. The current paper contains all the proofs, more detailed descriptions and additional simulations.

II. Review of GAMP

A. Sum-Product GAMP

Before describing the adaptive GAMP algorithm, it is useful to review the basic (non-adaptive) GAMP algorithm of [18]. We begin with a description of the most basic – and perhaps most important – variant of GAMP, namely

sum-product GAMP. To describe the algorithm, consider the estimation problem in Fig. 1 where the componentwise probability density functions on the inputs and outputs have some parametric form,

$$P_X(x|\lambda_x), \quad P_{Y|Z}(y|z,\lambda_z), \tag{1}$$

where $\lambda_x \in \Lambda_x$ and $\lambda_z \in \Lambda_z$ represent parameters of the densities and $\Lambda_x \subseteq \mathbb{R}^{d_x}$ and $\Lambda_z \subseteq \mathbb{R}^{d_z}$ denote the corresponding parameter sets that are of finite dimensions. Now, suppose that the components of $\mathbf{x}$ are i.i.d. with $x_j \sim P_X(x_j|\lambda_x)$ and, conditional on the transform output $\mathbf{z} = \mathbf{A}\mathbf{x}$, the components of the observations $\mathbf{y}$ have a likelihood $y_i \sim P_{Y|Z}(y_i|z_i,\lambda_z)$. Then, the posterior joint probability density of $\mathbf{x}$ and $\mathbf{z}$ will be given by

$$P(\mathbf{x},\mathbf{z}|\mathbf{y},\lambda_x,\lambda_z) \propto \mathbb{1}_{\{\mathbf{z}=\mathbf{A}\mathbf{x}\}} \times \prod_{i=1}^{m} P_{Y|Z}(y_i|z_i,\lambda_z) \prod_{j=1}^{n} P_X(x_j|\lambda_x), \tag{2}$$

where $\propto$ denotes identity after normalization, and the indicator function is used to indicate that the density is defined on the set $\mathbf{z} = \mathbf{A}\mathbf{x}$. The GAMP algorithm of [18] can be seen as a class of methods for approximately estimating the vectors $\mathbf{x}$ and $\mathbf{z}$ under this joint distribution in the case when the parameters λ_x and λ_z are *known*.

As described in [18], there are two important variants of GAMP:

- *Sum-product GAMP:* This method is used for approximately computing the posterior marginals

$$P(x_j|\mathbf{y},\lambda_x,\lambda_z), \quad P(z_i|\mathbf{y},\lambda_x,\lambda_z), \tag{3}$$

with respect to the joint density (2). From these posterior marginals, one can compute the posterior means and variances,

$$\widehat{x}_j = \mathbb{E}\left[x_j \mid \mathbf{y},\lambda_x,\lambda_z\right] \tag{4a}$$
$$\tau_{x_j} = \mathrm{var}\left[x_j \mid \mathbf{y},\lambda_x,\lambda_z\right] \tag{4b}$$
$$\widehat{z}_i = \mathbb{E}\left[z_i \mid \mathbf{y},\lambda_x,\lambda_z\right] \tag{4c}$$
$$\tau_{z_i} = \mathrm{var}\left[z_i \mid \mathbf{y},\lambda_x,\lambda_z\right]. \tag{4d}$$

The GAMP algorithm in this case is based on a Gaussian approximation of sum-product loopy belief propagation.

- *Max-sum GAMP:* This variant is used to approximately compute the maximum a posteriori (MAP) estimate

$$(\widehat{\mathbf{x}},\widehat{\mathbf{z}}) := \arg\max_{(\mathbf{x},\mathbf{z})} P(\mathbf{x},\mathbf{z}|\mathbf{y},\lambda_x,\lambda_z), \tag{5}$$

and is based on a quadratic approximation of the max-sum loopy belief propagation.

In this paper, we focus mostly on sum-product GAMP algorithm, although many of the statements can be applied to the max-sum GAMP algorithm equally. The basic steps of the sum-product GAMP algorithm from [18] are shown in Algorithm 1. The algorithm is an iterative procedure generating a sequence of estimates $\widehat{\mathbf{x}}^t$, $\boldsymbol{\tau}_x^t$ representing estimates of the posterior means and variances in (4).

Exact computation of the means and variance of the components x_j and z_i of the posterior joint density (2) is generally intractable, since it involves a marginalization over n variables. The main concept in the GAMP algorithm is to approximately reduce this vector-valued estimation problem to a sequence of scalar mean and variance computations. Specifically, the expectations and variances in lines 16 and 17 are to be taken with respect to the probability density

$$P(x_j|r_j^t,\tau_r^t,\lambda_x) \propto P_X(x_j|\lambda_x)\exp\left[-\frac{1}{2\tau_r^t}|x_j - r_j^t|^2\right]. \tag{6}$$

The density (6) is also the GAMP approximation of the posterior marginal density $P(x_j|\mathbf{y},\lambda_x,\lambda_z)$. Similarly, in lines 8 and 9, the expectation and variance are to be taken with respect to the distribution

$$P(z_i|y_i,p_i^t,\tau_p^t,\lambda_z) \propto P_{Y|Z}(y_i|z_i,\lambda_z)\exp\left[-\frac{1}{2\tau_p^t}|z_i - p_i^t|^2\right]. \tag{7}$$

The density (7) can also be taken as an approximation of the posterior marginal density $P(z_i|\mathbf{y},\lambda_x,\lambda_z)$.

Now, the probability densities (6) and (7) are over one-dimensional random variables. Thus, even if their means and variances do not have closed-form expressions, they can be computed via numerical integration. In addition, the densities can be interpreted as posterior distributions on scalar random variables x_j and z_i with respect to observations r_j^t and (y_i, p_i^t) of the form

$$r_j^t = x_j + \mathcal{N}(0,\tau_r^t), \quad x_j \sim P_X(x_j|\lambda_x) \tag{8a}$$
$$y_i \sim P_{Y|Z}(y_i|z_i), \quad z_i \sim \mathcal{N}(p_i^t,\tau_p^t). \tag{8b}$$

Hence, computing the posterior means and variances of x_j and z_i in lines 8, 9, 16 and 17 is equivalent to a set of scalar AWGN estimation problems. In this way, the sum-product GAMP algorithm reduces the inherently vector-valued

Algorithm 1 Non-Adaptive Sum-Product GAMP

Require: Matrix $\mathbf{A}$ and densities P_X and $P_{Y|Z}$ with known parameters λ_x and λ_z.

1: {Initialize}
2: $t \leftarrow 0$, $\mathbf{s}^{t-1} \leftarrow 0$
3: $\widehat{\mathbf{x}}^0 \leftarrow \mathbb{E}(x|\lambda_x)$, $\tau_x^0 \leftarrow \mathrm{var}(x|\lambda_x)$.
4: **repeat**
5: {Output node update}
6: $\tau_p^t \leftarrow \|\mathbf{A}\|_F^2 \tau_x^t / m$
7: $\mathbf{p}^t \leftarrow \mathbf{A}\widehat{\mathbf{x}}^t - \mathbf{s}^{t-1}\tau_p^t$
8: $\widehat{z}_i^t \leftarrow \mathbb{E}(z_i|y_i, p_i^t, \tau_p^t, \lambda_z)$ for all $i = 1,\ldots,m$
9: $\tau_{z_i}^t \leftarrow \mathrm{var}(z_i|y_i, p_i^t, \tau_p^t, \lambda_z)$ for all $i = 1,\ldots,m$
10: $s_i^t \leftarrow (z_i^t - p_i^t)/\tau_p^t$ for all $i = 1,\ldots,m$
11: $\tau_s^t \leftarrow (1/m)\sum_i (1 - \tau_{z_i}^t/\tau_p^t)/\tau_p^t$
12:
13: {Input node update}
14: $1/\tau_r^t \leftarrow \|\mathbf{A}\|_F^2 \tau_s^t / n$
15: $\mathbf{r}^t \leftarrow \widehat{\mathbf{x}}^t + \tau_r^t \mathbf{A}^{\mathrm{T}}\mathbf{s}^t$
16: $\widehat{x}_j^{t+1} \leftarrow \mathbb{E}(x_j|r_j^t, \tau_r^t, \lambda_x)$ for all $j = 1,\ldots,n$
17: $\tau_x^{t+1} \leftarrow (\tau_r^t/n)\sum_j \mathrm{var}(x_j|r_j^t, \tau_r^t, \lambda_x)$
18: **until** Terminated

Algorithm 2 General (Non-Adaptive) GAMP

Require: Matrix $\mathbf{A}$, estimation functions G_x^t, G_s^t and G_z^t and parameter estimates $\overline{\lambda}_x^t$ and $\overline{\lambda}_z^t$.

1: Set $t \leftarrow 0$, $\mathbf{s}^{t-1} \leftarrow 0$ and select some initial values for $\widehat{\mathbf{x}}^0$ and τ_x^0.
2: **repeat**
3: {Output node update}
4: $\tau_p^t \leftarrow \|\mathbf{A}\|_F^2 \tau_x^t / m$
5: $\mathbf{p}^t \leftarrow \mathbf{A}\widehat{\mathbf{x}}^t - \mathbf{s}^{t-1}\tau_p^t$
6: $\widehat{z}_i^t \leftarrow G_z^t(p_i^t, y_i, \tau_p^t, \overline{\lambda}_z^t)$ for all $i = 1, \ldots, m$
7: $s_i^t \leftarrow G_s^t(p_i^t, y_i, \tau_p^t, \overline{\lambda}_z^t)$ for all $i = 1, \ldots, m$
8: $\tau_s^t \leftarrow -(1/m)\sum_i \partial G_s^t(p_i^t, y_i, \tau_p^t, \overline{\lambda}_z^t)/\partial p_i^t$
9:
10: {Input node update}
11: $1/\tau_r^t \leftarrow \|\mathbf{A}\|_F^2 \tau_s^t / n$
12: $\mathbf{r}^t \leftarrow \widehat{\mathbf{x}}^t + \tau_r^t \mathbf{A}^{\mathrm{T}} \mathbf{s}^t$
13: $\widehat{x}_j^{t+1} \leftarrow G_x^t(r_j^t, \tau_r^t, \overline{\lambda}_x^t)$ for all $j = 1, \ldots, n$
14: $\tau_x^{t+1} \leftarrow (\tau_r^t/n)\sum_j \partial G_x^t(r_j^t, \tau_r^t, \overline{\lambda}_x^t)/\partial r_j$
15: **until** Terminated

inference problem to a sequence of scalar AWGN estimation problems at the input and output, along with transform by $\mathbf{A}$ and $\mathbf{A}^{\mathrm{T}}$. This is computationally attractive since the algorithm involves no vector-valued estimation steps or matrix inverses.

Of course, the GAMP algorithm is only an approximation of the true inference problem. The performance of the method and convergence results can be found in references mentioned above.

B. General GAMP

As mentioned above, the sum-product GAMP algorithm is a particular instance of a more general class of algorithms that includes the max-sum GAMP algorithm for MAP estimation. To provide the most general results for the adaptive GAMP, we briefly review the general (non-adaptive) GAMP algorithm. Full details of the general GAMP algorithm can be found in [18]. For completeness, we restate the steps of the general GAMP algorithm in Algorithm 2.

Comparing Algorithms 1 and 2, we see that there are two generalizations in the general GAMP algorithm. First, the mean and variance computations in lines 8, 9, 16 and 17 of the sum-product GAMP algorithm, Algorithm 1, are replaced with general *estimation functions* G_x^t, G_s^t and G_z^t. These estimation functions take the outputs $\mathbf{r}^t$ and $\mathbf{p}^t$ and generate the estimates $\mathbf{x}^t$, $\mathbf{s}^t$ and $\mathbf{z}^t$. Their derivatives results in the variance terms $\boldsymbol{\tau}_x^t$, $\boldsymbol{\tau}_s^t$ and $\boldsymbol{\tau}_z^t$. It is shown in [18] that with appropriate selection of these estimation functions, one can incorporate both the sum-product and max-sum variants of the GAMP algorithm.

For the case of the sum-product GAMP, we can recover Algorithm 1 with the estimation functions

$$G_x^t(r, \tau_r, \lambda_x) := \mathbb{E}[x|r, \tau_r, \lambda_x], \tag{9a}$$

$$G_z^t(p, y, \tau_p, \lambda_z) := \mathbb{E}[z|p, y, \tau_p, \lambda_z], \tag{9b}$$

$$G_s^t(p, y, \tau_p, \lambda_z) := \frac{1}{\tau_p}\big(G_z^t(p, y, \tau_p, \lambda_z) - p\big), \tag{9c}$$

where the expectations are with respect to the distributions in (6) and (7). It is shown in [18] that the derivatives of these estimation functions for lines 8 and 14 of Algorithm 2 agree with the variance computations in lines 11 and 17 of Algorithm 1. Also, note that the vector $\mathbf{s}^t$ can be interpreted as the current estimate of dual parameters [41].

The second difference between the the sum-product GAMP algorithm in Algorithm 1 and the more general Algorithm 2 is that the fixed parameter values λ_x and λ_z are replaced by a deterministic sequence of parameter values $\overline{\lambda}_x^t$ and $\overline{\lambda}_z^t$. Of course, if the parameters are known, there is no reason to change the parameter estimates on each iteration. However, we need to consider this generalization to enable the study of the adaptive GAMP algorithm below.

C. AWGN Outputs With Sparse Priors

As discussed in the Introduction, much of the current interest in the AMP and GAMP methods have been in the context of compressed sensing [13]–[19]. Thus, it is useful to briefly describe this particular application in more detail. The original AMP formulations in [13]–[15] consider the special case of AWGN output

$$y_i = z_i + w_i, \quad w_i \sim \mathcal{N}(0, \tau_w), \tag{10}$$

where the additive noise w_i is i.i.d. and independent of $\mathbf{z}$. In this case, as shown in [18], the output updates in line 7 and 8 reduce to

$$s_i^t = (y_i - p_i^t)/(\tau_p^t + \tau_w), \quad \tau_s^t = 1/(\tau_p^t + \tau_w).$$

For Bayesian forms of compressed sensing, one then takes a sparse prior for the density P_X. A common density is the Laplacian prior,

$$P_X(x_j|\lambda_x) = \frac{\lambda_x}{2} e^{-\lambda_x |x_j|}.$$

In this case, the MAP estimate (5) corresponds to the classic LASSO estimate [33]. Although we have not discussed the max-sum GAMP algorithm, as shown in [18], the equations for the estimation function G_x^t in line (2) of Algorithm 2 reduce to the classic soft-thresholding operator. In this way, the max-sum GAMP with a Laplacian prior reduces to a variant of an iterative soft-thresholding algorithm – see [13], [34] for a general discussion.

D. State Evolution Analysis

In addition to its computational simplicity and generality, a key motivation of the GAMP algorithm is that its asymptotic behavior can be precisely characterized when $\mathbf{A}$ is a large i.i.d. Gaussian transform. The asymptotic behavior is described by what is known as a *state evolution* (SE) analysis. By now, there are a large number of SE results for AMP-related algorithms [9], [11]–[18]. Here, we review the particular SE analysis from [18] and [23] which is based on the framework in [16].

Assumption 1: Consider a sequence of random realizations of the general GAMP algorithm, Algorithm 2, indexed by the dimension n, satisfying the following assumptions:

(a) For each n, the matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$ has i.i.d. components with $A_{ij} \sim \mathcal{N}(0, 1/m)$ and the dimension $m = m(n)$ is a deterministic function of n satisfying $n/m \to \beta$ for some $\beta > 0$ as $n \to \infty$.

(b) The input vectors $\mathbf{x}$ and initial condition $\widehat{\mathbf{x}}^0$ are deterministic sequences whose components converge empirically with bounded moments of order $s = 2k - 2$ as

$$\lim_{n\to\infty} (\mathbf{x}, \widehat{\mathbf{x}}^0) \overset{\text{PL}(s)}{=} (X, \widehat{X}^0), \tag{11}$$

to some random vector $(X, \widehat{X}^0)$ for some $k \geq 2$. Loosely, this convergence implies that the empirical distribution of the components of $(\mathbf{x}, \widehat{\mathbf{x}}^0)$ converge to the distribution of $(X, \widehat{X}^0)$. A precise definition is given in Appendix A.

(c) The output vectors $\mathbf{z}$ and $\mathbf{y} \in \mathbb{R}^m$ are generated by

$$\mathbf{z} = \mathbf{A}\mathbf{x}, \text{ and } y_i = h(z_i, w_i) \text{ for all } i = 1, \ldots, m, \tag{12}$$

for some scalar-valued function $h(z, w)$ and vector $\mathbf{w} \in \mathbb{R}^m$ representing an output disturbance. It is assumed that the output disturbance vector $\mathbf{w}$ is deterministic, but empirically converges as

$$\lim_{n\to\infty} \mathbf{w} \overset{\text{PL}(s)}{=} W, \tag{13}$$

where $s = 2k - 2$ is as in Assumption 1(b) and W is some random variable. We let $P_{Y|Z}$ denote the conditional distribution of the random variable $Y = h(Z, W)$.

(d) The estimation function $G_x^t(r, \tau_r, \lambda_x)$ and its derivative with respect to r, is Lipschitz continuous in r at $(\tau_r, \lambda_x) = (\overline{\tau}_r^t, \overline{\lambda}_x^t)$, where $\overline{\tau}_r^t$ is a deterministic parameter from the SE equations below. A similar assumptions holds for $G_z^t(p, \tau_p, \lambda_z)$.

Assumption 3(a) simply states that we are considering large, Gaussian i.i.d. matrices $\mathbf{A}$. Assumptions (b) and (c) state that the input vector $\mathbf{x}$ and output disturbance $\mathbf{w}$ are modeled as deterministic, but whose empirical distributions asymptotically appear as i.i.d. This deterministic model is one of key features of Bayati and Montanari's analysis in [16]. Assumption (d) is a mild continuity condition.

Note that, for now, there is no assumption that the "true" distribution of X or the true conditional distribution of Y given Z must belong to the class of distributions (1) for any parameters λ_x and λ_z. The analysis can thus model the effects of model mismatch.

Next, we define the sets of two vectors

$$\theta_x^t := \{(x_j, r_j^t, \widehat{x}_j^{t+1}), j = 1, \ldots, n\}, \tag{14a}$$

$$\theta_z^t := \{(z_i, \widehat{z}_i^t, y_i, p_i^t), i = 1, \ldots, m\}. \tag{14b}$$

The first set θ_x^t represents the components of the "true," but unknown, input vector $\mathbf{x}$, its GAMP estimate $\widehat{\mathbf{x}}^t$ as well as $\mathbf{r}^t$. The second set θ_z^t contains the components of the "true," but unknown, output vector $\mathbf{z}$, its GAMP estimate $\widehat{\mathbf{z}}^t$, as well as $\mathbf{p}^t$ and the observed output $\mathbf{y}$. The sets θ_x^t and θ_z^t are implicitly functions of the dimension n.

The main result of [18] shows that if we fix the iteration t, and let $n \to \infty$, the asymptotic joint empirical distribution of the components of these two sets θ_x^t and θ_z^t converges to random vectors of the form

$$\overline{\theta}_x^t := (X, R^t, \widehat{X}^{t+1}), \qquad \overline{\theta}_z^t := (Z, \widehat{Z}^t, Y, P^t). \tag{15}$$

We precisely state the nature of convergence momentarily (see Theorem 1). In (15), X is the random variable in Assumption 1(b), while R^t and $\widehat{X}^{t+1}$ are given by

$$R^t = \alpha_r^t X + V^t, \qquad V^t \sim \mathcal{N}(0, \xi_r^t), \tag{16a}$$

$$\widehat{X}^{t+1} = G_x^t(R^t, \overline{\tau}_r^t, \overline{\lambda}_x^t) \tag{16b}$$

for some deterministic constants α_r^t, ξ_r^t, and $\overline{\tau}_r^t$ that are defined below. Similarly, $(Z, P^t) \sim \mathcal{N}(0, \mathbf{K}_p^t)$ for some covariance matrix $\mathbf{K}_p^t$, and

$$Y = h(Z, W), \qquad \widehat{Z}^t = G_z^t(P^t, Y, \overline{\tau}_p^t, \overline{\lambda}_z^t), \tag{17}$$

where W is the random variable in (13) and $\mathbf{K}_p^t$ contains deterministic constants.

The deterministic constants α_r^t, ξ_r^t, $\overline{\tau}_r^t$ and $\mathbf{K}_p^t$ represent parameters of the distributions of $\overline{\theta}_x^t$ and $\overline{\theta}_z^t$ and depend on both the distributions of the input and outputs as well as the choice of the estimation and adaptation functions. The SE equations provide a simple method for recursively computing these parameters. The equations are best described algorithmically as shown in Algorithm 4. In order not to repeat ourselves, in Algorithm 4, we have written the SE equations for adaptive GAMP: For non-adaptive GAMP, the updates (32b) and (33a) can be ignored as the values of $\overline{\lambda}_z^t$ and $\overline{\lambda}_x^t$ are pre-computed.

With these definitions, we can state the main result from [18].

Theorem 1 ([18]): Consider the random vectors θ_x^t and θ_z^t generated by the outputs of GAMP under Assumption 1. Let $\overline{\theta}_x^t$ and $\overline{\theta}_z^t$ be the random vectors in (15) with the parameters determined by the SE equations in Algorithm 4. Then, for any fixed t, the elements of the sets θ_x^t and θ_z^t converge empirically with bounded moments of order k as

$$\lim_{n\to\infty} \theta_x^t \overset{\text{PL}(k)}{=} \overline{\theta}_x^t, \qquad \lim_{n\to\infty} \theta_z^t \overset{\text{PL}(k)}{=} \overline{\theta}_z^t. \tag{18}$$

where $\overline{\theta}_x^t$ and $\overline{\theta}_z^t$ are given in (15). In addition, for any t, the limits

$$\lim_{n\to\infty} \tau_r^t = \overline{\tau}_r^t, \qquad \lim_{n\to\infty} \tau_p^t = \overline{\tau}_p^t, \tag{19}$$

also hold almost surely.

The theorem shows that the components of the vectors $\mathbf{x}$ and $\mathbf{z}$, and their GAMP estimates $\widehat{\mathbf{x}}^t$ and $\widehat{\mathbf{z}}^t$ have the same statistical distribution as random variables X, Z, $\widehat{X}^t$ and $\widehat{Z}^t$ in a simple scalar equivalent system. This scalar equivalent model appears in several analyses and can be thought of as a *single-letter characterization* [35] of the system. Remarkably, this limiting property holds for essentially arbitrary distributions and estimation functions, even the ones that arise from problems that are highly nonlinear or noncovex. From the single-letter characterization, one can compute the asymptotic value of essentially any component-wise performance metric such as mean-squared error or detection accuracy. Similar single-letter characterizations can also be derived by arguments from statistical physics [27], [36]–[39].

Algorithm 3 Adaptive GAMP

Require: Matrix $\mathbf{A}$, estimation functions G_x^t, G_s^t and G_z^t and adaptation functions H_x^t and H_z^t.

Set $t \leftarrow 0$, $\mathbf{s}^{t-1} \leftarrow 0$ and select some initial values for $\widehat{\mathbf{x}}^0$ and τ_x^0.
repeat
  {Output node update}
  $\tau_p^t \leftarrow \|\mathbf{A}\|_F^2 \tau_x^t / m$
  $\mathbf{p}^t \leftarrow \mathbf{A}\widehat{\mathbf{x}}^t - \mathbf{s}^{t-1}\tau_p^t$
  $\widehat{\lambda}_z^t \leftarrow H_z^t(\mathbf{p}^t, \mathbf{y}, \tau_p^t)$
  $\widehat{z}_i^t \leftarrow G_z^t(p_i^t, y_i, \tau_p^t, \widehat{\lambda}_z^t)$ for all $i = 1, \ldots, m$
  $s_i^t \leftarrow G_s^t(p_i^t, y_i, \tau_p^t, \widehat{\lambda}_z^t)$ for all $i = 1, \ldots, m$
  $\tau_s^t \leftarrow -(1/m)\sum_i \partial G_s^t(p_i^t, y_i, \tau_p^t, \widehat{\lambda}_z^t)/\partial p_i^t$

  {Input node update}
  $1/\tau_r^t \leftarrow \|\mathbf{A}\|_F^2 \tau_s^t / n$
  $\mathbf{r}^t \leftarrow \mathbf{x}^t + \tau_r^t \mathbf{A}^{\mathrm{T}} \mathbf{s}^t$
  $\widehat{\lambda}_x^t \leftarrow H_x^t(\mathbf{r}^t, \tau_r^t)$
  $\widehat{x}_j^{t+1} \leftarrow G_x^t(r_j^t, \tau_r^t, \widehat{\lambda}_x^t)$ for all $j = 1, \ldots, n$
  $\tau_x^{t+1} \leftarrow (\tau_r^t/n)\sum_j \partial G_x^t(r_j^t, \tau_r^t, \widehat{\lambda}_x^t)/\partial r_j$
until Terminated

E. State Evolution Analysis for Sum-Product GAMP

For the special case of the sum-product GAMP algorithm in Algorithm 1, the SE equations in Algorithm 4 reduce to a particularly simple form. As shown in [18], the variance terms $\overline{\tau}_r^t$ and ξ_r^t in (32) are given by

$$\overline{\tau}_r^t = \xi_r^t = \mathbb{E}^{-1}\left[\frac{\partial^2}{\partial p^2}\log p_{Y|P}(y|p^t)\right], \tag{20a}$$

where the expectations are over the random variables $(Z, P^t) \sim \mathcal{N}(0, \mathbf{K}_p^t)$ and Y is given in (17). The covariance matrix $\mathbf{K}_p^t$ has the form

$$\mathbf{K}_p^t = \begin{bmatrix} \beta\tau_{x0} & \beta\tau_{x0} - \overline{\tau}_p^t \\ \beta\tau_{x0} - \overline{\tau}_p^t & \beta\tau_{x0} - \overline{\tau}_p^t \end{bmatrix}, \tag{20b}$$

where τ_{x0} is the variance of X and $\beta > 0$ is the asymptotic measurement ratio (see Assumption 1 for details). The scaling constant (32e) becomes $\alpha_r^t = 1$. The update rule for $\overline{\tau}_x^{t+1}$ also simplifies to

$$\overline{\tau}_x^{t+1} = \mathbb{E}\left[\mathrm{var}\left(X|R^t\right)\right], \tag{20c}$$

where the expectation is over the random variables in (16). The SE equations for the sum-product GAMP will be initialized with

$$\overline{\tau}_p^0 = \beta\tau_{x0} \tag{21}$$

so that the initial value of the covariance matrix in (20b) is

$$\mathbf{K}_p^t = \begin{bmatrix} \beta\tau_{x0} & 0 \\ 0 & 0 \end{bmatrix}. \tag{22}$$

III. Adaptive GAMP

The above review of the standard GAMP algorithms in Algorithms 1 and 2 show that the methods apply to the case when the parameters λ_x and λ_z in the distributions in (1) are known. The adaptive GAMP method proposed here, and shown in Algorithm 3, is an extension of Algorithm 2 that enables simultaneous identification of finite dimensional λ_x and λ_z along with estimation of $\mathbf{x}$ and $\mathbf{z}$.

The key modification is the introduction of the two *adaptation functions*: $H_z^t(\mathbf{p}^t, \mathbf{y}, \tau_p^t)$ and $H_x^t(\mathbf{r}^t, \tau_r^t)$. In each iteration, these functions output estimates, $\widehat{\lambda}_z^t$ and $\widehat{\lambda}_x^t$ of the parameters based on the data $\mathbf{p}^t$, $\mathbf{y}$, $\mathbf{r}^t$, τ_p^t and τ_r^t.

The basic (non-adaptive) GAMP algorithm in Algorithm 2 is a special case when the estimation functions H_x^t and H_z^t output fixed values

$$H_z^t(\mathbf{p}^t, \mathbf{y}, \tau_p^t) = \overline{\lambda}_z^t, \qquad H_x^t(\mathbf{r}^t, \tau_r^t) = \overline{\lambda}_x^t, \tag{23}$$

for the *pre-computed* sequence of parameters $\overline{\lambda}_x^t$ and $\overline{\lambda}_z^t$. We call these values precomputed since, in the case of the non-adaptive GAMP algorithm, the parameter estimates $\overline{\lambda}_x^t$ and $\overline{\lambda}_z^t$ do not depend on the data through the vectors $\mathbf{p}^t$, $\mathbf{y}^t$, and $\mathbf{r}^t$. A particular case of the non-adaptive algorithm would be the oracle scenario, where $\overline{\lambda}_x^t$ and $\overline{\lambda}_z^t$ are set to the true values of the parameters and do not change with the iteration number t.

However, the adaptive GAMP algorithm in Algorithm 3 is significantly more general and enables a large class of methods for estimating the parameters based on the data. One particular adaptation method is based on on maximum likelihood (ML) as described next.

A. ML Parameter Estimation

As one possible method to estimate the parameters, recall from Theorem 1 that the empirical distribution of the components of $\mathbf{r}^t$ converges weakly to the distribution of R^t in (16). Now, the distribution of R^t only depends on three parameters – α_r^t, ξ_r^t and λ_x. It is thus natural to attempt to estimate these parameters from the empirical distribution of the components of $\mathbf{r}^t$ and thereby recover the parameter λ_x.

To this end, let $\phi_x(r, \lambda_x, \alpha_r, \xi_r)$ be the log likelihood

$$\phi_x(r, \lambda_x, \alpha_r, \xi_r) := \log P_R(r|\lambda_x, \alpha_r, \xi_r), \tag{24}$$

where the right-hand side is the probability density of a random variable R with distribution

$$R = \alpha_r X + V, \quad X \sim P_X(\cdot|\lambda_x), \quad V \sim \mathcal{N}(0, \xi_r). \tag{25}$$

Then, at any iteration t, we can attempt to perform a maximum-likelihood (ML) estimate

$$\widehat{\lambda}_x^t = H_x^t(\mathbf{r}^t, \tau_r^t) := \arg\max_{\lambda_x \in \Lambda_x} \max_{(\alpha_r, \xi_r) \in S_x(\tau_r^t)} \left\{ \frac{1}{n}\sum_{j=1}^n \phi_x(r_j^t, \lambda_x, \alpha_r, \xi_r) \right\}. \tag{26}$$

Here, the set $S_x(\tau_r^t)$ is a set of possible values for the parameters α_r, ξ_r. This set may depend on the measured variance τ_r^t and we will see its precise role below. The selection of the sets is critical and discussed in detail in Section V.

Similarly, the individual components of $\mathbf{p}^t$ and $\mathbf{y}$ have the same distribution as (P^t, Y) which depend only on the parameters $\mathbf{K}_p$ and λ_z. Thus, we can define the likelihood

$$\phi_z(p, y, \lambda_z, \mathbf{K}_p) := \log P_{Y,P}(y, p|\lambda_z, \mathbf{K}_p), \tag{27}$$

where the right-hand side is the joint probability density of (P, Y) with distribution

$$Y \sim P_{Y|Z}(\cdot|Z, \lambda_z), \quad (Z, P) \sim \mathcal{N}(0, \mathbf{K}_p). \tag{28}$$

Then, we estimate λ_z via the ML estimate

$$\begin{aligned}\widehat{\lambda}_z^t &= H_z^t(\mathbf{p}^t, \mathbf{y}, \tau_p^t) \\ &:= \arg\max_{\lambda_z \in \Lambda_z} \max_{\mathbf{K}_p \in S_z(\tau_p^t)} \left\{ \frac{1}{m} \sum_{i=1}^{m} \phi_z(p_i^t, y_i, \lambda_z, \mathbf{K}_p) \right\}. \end{aligned} \tag{29}$$

Again, the set $S_z(\tau_p^t)$ is a set of possible covariance matrices $\mathbf{K}_p$.

B. Relation to EM-GAMP

Before discussing the convergence of the adaptive GAMP algorithm with ML parameter estimation, it is useful to briefly compare the ML parameter estimation with the EM-GAMP method proposed by Vila and Schniter [24], [25] and Krzakala *et. al.* [27], [28]. Both of these methods combine the Bayesian AMP [14], [15] or GAMP algorithms [18] with a standard EM procedure [29] as follows. First, the algorithms use the sum-product version of the AMP/GAMP, so that the outputs provide an estimate of the posterior distributions on the components of $\mathbf{x}$ given the current parameter values. From the discussion in Section II-A, we know that (6) and (7) can be taken as an approximations of the true posteriors of x_j and z_i for a *given* set of parameter values λ_x and λ_z. Using the approximation, we approximately implement the EM procedure to update the parameter estimate via a maximization

$$\begin{aligned}\widehat{\lambda}_x^t &= H_x^t(\mathbf{r}^t, \tau_r^t) \\ &:= \arg\max_{\lambda_x \in \Lambda_x} \frac{1}{n} \sum_{j=1}^{n} \mathbb{E}\Big[\log P_X(x_j|\lambda_x) | r_j^t, \tau_r^t, \widehat{\lambda}_x^{t-1} \Big], \end{aligned} \tag{30}$$

where the expectation is with respect to the distribution in (6). In [24] and [25], the parameter update (30) is performed only once every few iterations to allow $\widehat{P}^t$ to converge to the approximation of the posterior distribution of x_j given the current parameter estimates. In [27] and [28], the parameter estimate is updated at every iteration. A similar procedure is performed for the estimation of λ_z.

We thus see that the EM-GAMP procedures in [24] and [25] and in [27] and [28] are thus both special cases of the adaptive GAMP algorithm in Algorithm 3 with particular choices of the adaptation functions H_x^t and H_z^t. As a result, our analysis in Theorem 2 below applies to these algorithms as well and provides rigorous asymptotic characterizations of the EM-GAMP performance. However, at the current time, we can only prove the asymptotic consistency result for the ML adaptation functions (26) and (29) described above.

That being said, it should be pointed out that the EM-GAMP update (30) is generally computationally much simpler than the ML updates in (26) and (29). For example, when $P_X(x|\lambda_x)$ is an exponential family, the optimization in (30) is convex. Also, the optimizations in (26) and (29) require searches over additional parameters such as α_r and ξ_r. Thus, an interesting avenue of future work is to apply the analysis result of Theorem 3 below, to see if the EM-GAMP method or some similarly computationally simple technique can be developed which also provides asymptotic consistency.

Algorithm 4 Adaptive GAMP State Evolution

Given the distributions in Assumption 1, compute the sequence of parameters as follows:

- *Initialization* Set $t = 0$ with

$$\mathbf{K}_x^0 = \mathrm{cov}(X, \widehat{X}^0), \qquad \overline{\tau}_x^0 = \tau_x^0, \tag{31}$$

  where the expectation is over the random variables $(X, \widehat{X}^0)$ in Assumption 1(b) and τ_x^0 is the initial value in the GAMP algorithm.
- *Output node update:* Compute the variables

$$\overline{\tau}_p^t = \beta \overline{\tau}_x^t, \qquad \mathbf{K}_p^t = \beta \mathbf{K}_x^t, \tag{32a}$$

$$\overline{\lambda}_z^t = H_z^t(P^t, Y, \overline{\tau}_p^t), \tag{32b}$$

$$\overline{\tau}_r^t = -\mathbb{E}^{-1}\left[\frac{\partial}{\partial p} G_s^t(P^t, Y, \overline{\tau}_p^t, \overline{\lambda}_z^t) \right], \tag{32c}$$

$$\xi_r^t = (\overline{\tau}_r^t)^2 \mathbb{E}\left[G_s^t(P^t, Y, \overline{\tau}_p^t, \overline{\lambda}_z^t) \right], \tag{32d}$$

$$\alpha_r^t = \overline{\tau}_r^t \mathbb{E}\left[\frac{\partial}{\partial z} G_s^t(P^t, h(Z, W), \overline{\tau}_p^t, \overline{\lambda}_z^t) \right], \tag{32e}$$

  where the expectations are over the random variables $(Z, P^t) \sim \mathcal{N}(0, \mathbf{K}_p^t)$ and Y is given in (17).
- *Input node update:* Compute

$$\overline{\lambda}_x^t = H_x^t(R^t, \overline{\tau}_r^t), \tag{33a}$$

$$\overline{\tau}_x^{t+1} = \overline{\tau}_r^t \mathbb{E}\left[\frac{\partial}{\partial r} G_x^t(R^t, \overline{\tau}_r^t, \overline{\lambda}_x^t) \right], \tag{33b}$$

$$\mathbf{K}_x^{t+1} = \mathrm{cov}(X, \widehat{X}^{t+1}), \tag{33c}$$

  where the expectations are over the random variables in (16).

IV. Convergence and Asymptotic Consistency With Gaussian Transforms

A. General State Evolution Analysis

Before proving the asymptotic consistency of adaptive GAMP with ML adaptation, we first prove the following more general convergence result.

Assumption 2: Consider the adaptive GAMP algorithm running on a sequence of problems indexed by the dimension n, satisfying the following assumptions:

(a) Same as Assumption 1(a) to (c) with $k = 2$.

(b) For every t, the adaptation function $H_x^t(\mathbf{r}, \tau_r)$ is a functional over $\mathbf{r}$ satisfying the following weak pseudo-Lipschitz continuity property: Consider any sequence of vectors $\mathbf{r} = \mathbf{r}^{(n)}$ and sequence of scalars $\tau_r = \tau_r^{(n)}$, indexed by n satisfying

$$\lim_{n\to\infty} \mathbf{r}^{(n)} \overset{\mathrm{PL}(k)}{=} R^t, \qquad \lim_{n\to\infty} \tau_r^{(n)} = \overline{\tau}_r^t,$$

where R^t and $\overline{\tau}_r^t$ are the outputs of the state evolution equations defined below. Then,

$$\lim_{n\to\infty} H_x^t(\mathbf{r}^{(n)}, \tau_r^{(n)}) = H_x^t(R^t, \overline{\tau}_r^t).$$

Similarly, $H_z^t(\mathbf{y}, \mathbf{p}, \tau_p)$ satisfies analogous continuity conditions in τ_p and $(\mathbf{y}, \mathbf{p})$. See Appendix A for a general definition of weakly pseudo-Lipschitz continuous functionals.

(c) The scalar-valued function $G_x^t(r, \tau_r, \lambda_x)$ and its derivative $G'^t_x(r, \tau_r, \lambda_x)$ with respect to r are continuous in λ_x uniformly over r in the following sense: For every $\epsilon > 0$, t, τ_r^* and $\lambda_x^* \in \Lambda_x$, there exists an open neighborhood U of (τ_r^*, λ_x^*) such that for all $(\tau_r, \lambda_x) \in U$ and r,

$$|G_x^t(r, \tau_r, \lambda_x) - G_x^t(r, \tau_r^*, \lambda_x^*)| < \epsilon,$$
$$|G'^t_x(r, \tau_r, \lambda_x) - G'^t_x(r, \tau_r^*, \lambda_x^*)| < \epsilon.$$

In addition, the functions $G_x^t(r, \tau_r, \lambda_x)$ and $G'^t_x(r, \tau_r, \lambda_x)$ must be Lipschitz continuous in r with a Lipschitz constant that can be selected continuously in τ_r and λ_x. The functions $G_s^t(p, y, \tau_p, \lambda_z)$, $G_z^t(p, y, \tau_p, \lambda_z)$ and their derivatives $G'^t_s(p, y, \tau_p, \lambda_z)$ $G'^t_z(p, y, \tau_p, \lambda_z)$ satisfy analogous continuity assumptions with respect to p, y, τ_p and λ_z.

Although technical, assumptions (b) and (c) are mild continuity conditions that can be satisfied by a large class of adaptation functionals and estimation functions. For example, from the definitions in Appendix A, the continuity assumption (b) will be satisfied for any functional given by an empirical average

$$H_x^t(\mathbf{r}, \tau_r) = \frac{1}{n}\sum_{j=1}^{n} \phi_x^t(r_j, \tau_r),$$

where, for each t, $\phi_x^t(r_j, \tau_r)$ is pseudo-Lipschitz continuous in r of order p and continuous in τ_r uniformly over r. A similar functional can be used for H_z^t. As we will see in Section IV-B, the ML functionals (26) and (29) also satisfy the required conditions.

Theorem 2: Consider the random vectors θ_x^t and θ_z^t generated by the outputs of the adaptive GAMP under Assumption 2. Let $\overline{\theta}_x^t$ and $\overline{\theta}_z^t$ be the random vectors in (15) with the parameters determined by the SE equations in Algorithm 4. Then, for any fixed t, the components of θ_x^t and θ_z^t converge empirically with bounded moments of order $k = 2$ as

$$\lim_{n\to\infty} \theta_x^t \overset{\text{PL}(k)}{=} \overline{\theta}_x^t, \qquad \lim_{n\to\infty} \theta_z^t \overset{\text{PL}(k)}{=} \overline{\theta}_z^t, \tag{34}$$

where $\overline{\theta}_x^t$ and $\overline{\theta}_z^t$ are given in (15). In addition, for any t, the limits

$$\lim_{n\to\infty} \lambda_x^t = \overline{\lambda}_x^t, \qquad \lim_{n\to\infty} \lambda_z^t = \overline{\lambda}_z^t, \tag{35a}$$
$$\lim_{n\to\infty} \tau_r^t = \overline{\tau}_r^t, \qquad \lim_{n\to\infty} \tau_p^t = \overline{\tau}_p^t, \tag{35b}$$

also hold almost surely.

Proof: See Appendix B.

The result is a natural generalization of Theorem 1 and provides a simple extension of the SE analysis to incorporate the adaptation. The SE analysis applies to essentially arbitrary adaptation functions. It particular, it can be used to analyze both the behavior of the adaptive GAMP algorithm with either ML and EM-GAMP adaptation functions in the previous section.

The proof uses the standard techniques and is based on the same continuity argument as [40].

B. Asymptotic Consistency With ML Adaptation

We now use Theorem 2 to prove the asymptotic consistency of adaptive GAMP with the ML parameter estimation described in Section III-A. To guarantee consistency of the adaptive GAMP algorithm, we need to impose certain *identifiability* conditions. To understand the conditions, given parameters $(\lambda_x, \alpha_r, \xi_r)$ and $(\lambda_z, \mathbf{K}_p)$, let

$$P_R(\cdot|\lambda_x, \alpha_r, \xi_r), \quad P_{Y,P}(\cdot|\lambda_z, \mathbf{K}_p) \tag{36}$$

be the distributions of the random variables R and (Y, P) in (25) and (28), respectively.

Definition 1: Consider a family of distributions, $\{P_X(x|\lambda_x), \lambda_x \in \Lambda_x\}$, a set S_x of parameters (α_r, ξ_r) of a Gaussian channel, and the function $\phi_x(r, \lambda_x, \alpha_r, \xi_r)$. We say that $P_X(x|\lambda_x)$ is *identifiable with Gaussian outputs* with parameter set S_x and function ϕ_x if:

(a) The sets S_x and Λ_x are compact.

(b) For any "true" parameters $\lambda_x^* \in \Lambda_x$, and $(\alpha_r^*, \xi_r^*) \in S_x$, the maximization

$$\widehat{\lambda}_x = \arg\max_{\lambda_x \in \Lambda_x} \max_{(\alpha_r, \xi_r) \in S_x} \mathbb{E}\left[\phi_x(R, \lambda_x, \alpha_r, \xi_r)|\lambda_x^*, \alpha_r^*, \xi_r^*\right], \tag{37}$$

is well-defined, unique and returns the true value, $\widehat{\lambda}_x = \lambda_x^*$. The expectation in (37) is with respect to the distribution $R \sim P_R(\cdot|\lambda_x^*, \alpha_r^*, \xi_r^*)$ in (36).

(c) For every λ_x and α_r, ξ_r, the function $\phi_x(r, \lambda_x, \alpha_r, \xi_r)$ is pseudo-Lipschitz continuous of order $k = 2$ in r. In addition, it is continuous in $\lambda_x, \alpha_r, \xi_r$ uniformly over r in the following sense: For every $\epsilon > 0$ and $\widehat{\lambda}_x, \widehat{\alpha}_r, \widehat{\xi}_r$, there exists an open neighborhood U of $\widehat{\lambda}_x, \widehat{\alpha}_r, \widehat{\xi}_r$, such that for all $(\lambda_x, \alpha_r, \xi_r) \in U$ and all r,

$$|\phi_x(r, \lambda_x, \alpha_r, \xi_r) - \phi_x(r, \widehat{\lambda}_x, \widehat{\alpha}_r, \widehat{\xi}_r)| < \epsilon.$$

Definition 2: Consider a family of conditional distributions, $\{P_{Y|Z}(y|z, \lambda_z), \lambda_z \in \Lambda_z\}$ generated by the mapping $Y = h(Z, W, \lambda_z)$ where $W \sim P_W$ is some random variable and $h(z, \omega, \lambda_z)$ is a scalar-valued function. Let S_z be a set of covariance matrices $\mathbf{K}_p$ and let $\phi_z(y, p, \lambda_z, \mathbf{K}_p)$ be some function. We say that the conditional distribution family $P_{Y|Z}(\cdot|\cdot, \lambda_z)$ is *identifiable with Gaussian inputs* with covariance set S_z and function ϕ_z if:

(a) The parameter sets S_z and Λ_z are compact.

(b) For any "true" parameter $\lambda_z^* \in \Lambda_z$ and true covariance $\mathbf{K}_p^*$, the maximization

$$\widehat{\lambda}_z = \arg\max_{\lambda_z \in \Lambda_z} \max_{\mathbf{K}_p \in S_z} \mathbb{E}\big[\phi_z(Y, P, \lambda_z, \mathbf{K}_p)|\lambda_z^*, \mathbf{K}_p^*\big], \tag{38}$$

is well-defined, unique and returns the true value, $\widehat{\lambda}_z = \lambda_z^*$, The expectation in (38) is with respect to $(Y, P) \sim P_{Y,P}(\cdot|\lambda_z^*, \mathbf{K}_p^*)$.

(c) For every λ_z and $\mathbf{K}_p$, the function $\phi_z(y, p, \lambda_z, \mathbf{K}_p)$ is pseudo-Lipschitz continuous in (p, y) of order $k = 2$. In addition, it is continuous in $\lambda_p, \mathbf{K}_p$ uniformly over p and y.

Conditions (a) and (c) in both definitions are mild continuity and boundedness conditions. The main requirements is condition (b). Qualitatively, the definitions state that if R and (Y, P) are generated by models of the form (25) and (28), then the parameters in those models can be estimated through maximization of the functions ϕ_x and ϕ_z. The functions ϕ_x and ϕ_z can be the log likelihood functions (24) and (27), although we permit other functions as well, since the maximization may be computationally simpler. Such functions are sometimes called *pseudo-likelihoods*. We will discuss these conditions and the role of the sets S_x and S_z in more detail in Section V.

Assumption 3: Let $P_X(x|\lambda_x)$ and $P_{Y|Z}(y|z, \lambda_z)$ be families of distributions and consider the adaptive GAMP algorithm, Algorithm 3, run on a sequence of problems, indexed by the dimension n satisfying the following assumptions:

(a) Same as Assumption 1(a) to (c) with $k = 2$. In addition, the distributions for the vector X is given by $P_X(\cdot|\lambda_x^*)$ for some "true" parameter $\lambda_x^* \in \Lambda_x$ and the conditional distribution of Y given Z is given by $P_{Y|Z}(y|z, \lambda_z^*)$ for some "true" parameter $\lambda_z^* \in \Lambda_z$.

(b) Same as Assumption 2(c).

(c) The adaptation functions are set to (26) and (29).

Theorem 3: Consider the outputs of the adaptive GAMP algorithm with ML adaptation as described in Assumption 3. Then, for any fixed t,

(a) The components of θ_x^t and θ_z^t in (14) converge empirically with bounded moments of order $k = 2$ as in (34) and the limits (35) hold almost surely.

(b) In addition, if $(\alpha_r^t, \xi_r^t) \in S_x(\tau_r^t)$, and the family of distributions $P_X(\cdot|\lambda_x)$, $\lambda_x \in \Lambda_x$ is identifiable in Gaussian noise with parameter set $S_x(\tau_r^t)$ and pseudo-likelihood ϕ_x (see Definition 1), then

$$\lim_{n\to\infty} \widehat{\lambda}_x^t = \overline{\lambda}_x^t = \lambda_x^* \tag{39}$$

almost surely.

(c) Similarly, if $\mathbf{K}_p^t \in S_z(\tau_p^t)$ for some t, and the family of distributions $P_{Y|Z}(\cdot|\lambda_z)$, $\lambda_z \in \Lambda_z$ is identifiable with Gaussian inputs with parameter set $S_z(\tau_p^t)$ and pseudo-likelihood ϕ_z (see Definition 2) then

$$\lim_{n\to\infty} \widehat{\lambda}_z^t = \overline{\lambda}_z^t = \lambda_z^* \tag{40}$$

almost surely.

Proof: See Appendix C.

Remarkably, the theorem shows that for a very large class of the parameterized distributions, adaptive GAMP with ML adaptation is able to asymptotically estimate the correct parameters. Moreover, there is asymptotically no performance loss between adaptive GAMP and a corresponding oracle GAMP algorithm that knows the correct parameters in the sense that the empirical distributions of the algorithm outputs are described by the same SE equations.

C. Computational Issues

While Theorem 3 shows that adaptive GAMP with ML adaptation can recover consistent parameter estimates, the ML optimizations in (26) and (29) theoretically need to be computed exactly. In general, these optimizations will be non-convex. This requirement can be seen as the main disadvantage of the ML adaptation proposed in this paper relative to the EM-GAMP methods in [24], [25], [27], and [28]: while the proposed ML adaptation may have guaranteed consistency, the optimizations in each iteration may be non-convex. The EM iterations, in general are simpler.

Indeed, in the simulations in Section VI, we will need to approximate the optimization either through gradient ascent or other nonlinear optimization methods. Thus, the theory will not hold exactly. However, we will still observe a close match between the adaptive GAMP with an oracle GAMP with the correct parameters. Moreover, the ML adaptation is a non-convex optimization only over a number of variables only equal to the number of unknown parameters in λ_x and λ_z, not the vectors $\mathbf{x}$ and $\mathbf{z}$. Thus, for many practical problem, the overall optimization can be significantly simpler than the original non-convex problem.

V. Identifiability and Parameter Set Selection

In addition to the numerical optimization issues, Theorem 3 also imposes certain restrictions on the sets S_x and S_z over which the ML optimization must be performed. On the one hand, Theorem 3 requires that, to guarantee consistency, the sets must be sufficiently large to ensure that, for some iteration t, either $(\alpha_r^t, \xi_r^t) \in S_x(\tau_r^t)$ or $\mathbf{K}_p^t \in S_z(\tau_p^t)$. On the other hand, as we will see now, the sets may need to be constrained in order to satisfy the identifiability conditions in Definitions 1 and 2. In this section, we briefly provide some examples to illustrate under what cases these conditions can be met.

As discussed in the previous section, the main challenge in meeting the identifiability requirements in both Definitions 1 and 2 is condition (b). To understand this condition, we begin with the following simple lemma.

Lemma 1: Consider the distributions P_R and $P_{Y,P}$ in (36).

(a) When ϕ_x is the log-likelihood function in (24), then condition (b) of Definition 1 is satisfied if the mapping

$$(\lambda_x, \alpha_r, \xi_r) \mapsto P_R(\cdot|\lambda_x, \alpha_r, \xi_r) \tag{41}$$

is one-to-one in the set $\lambda_x \in \Lambda_x$ and $(\alpha_r, \xi_r) \in S_x$.

(b) Similarly, when ϕ_z is the log-likelihood function in (27), then condition (b) of Definition 2 is satisfied if the mapping

$$(\lambda_z, \mathbf{K}_p) \mapsto P_{Y,P}(\cdot|\lambda_z, \mathbf{K}_p) \tag{42}$$

is one-to-one in the set $\lambda_z \in \Lambda_z$ and $\mathbf{K}_z \in S_z$.

Proof: See Appendix D. ∎

Lemma 1 essentially states that if the true likelihood functions are used, then identifiability is equivalent to the

parametrizations of the distributions R and (Y, P) in in (25) and (28) being unique. That is, with sufficient observations of these variables, we should be able to uniquely recover the parameter values. To understand this in this context of the adaptive GAMP algorithm, recall from the state evolution analysis, that the components of the vectors $\mathbf{r}^t$ and $(\mathbf{y}, \mathbf{p}^t)$ are asymptotically distributed as R or (Y, P) in (25) and (28), respectively. Thus, if the parametrizations in (41) or (42) are not one-to-one, two different parameters values may give rise to the same asymptotic distributions on $\mathbf{r}^t$ and $(\mathbf{y}, \mathbf{p}^t)$. In this case, the adaptation functions in (26) and (29) that base the parameter estimates on $\mathbf{r}^t$ and $(\mathbf{y}, \mathbf{p}^t)$, cannot hope to distinguish between two such parameter values. On the other hand, if the parametrizations are one-to-one, the lemma shows that the ML parameter estimation will be able to correctly identify the parameter values. We now provide some examples.

A. Gaussian Mixtures

Suppose that X is a K-term Gaussian mixture with distribution,

$$X \sim \mathcal{N}(\mu_k, \tau_k) \text{ with probability } p_k,$$

with the unknown parameters being $\lambda_x = \{(\mu_k, \tau_k, p_k), k = 1, \ldots, K\}$. Then, the variable R in (25) will also be a Gaussian mixture, but with different components

$$R \sim \mathcal{N}(\alpha_r \mu_k, \alpha_r^2 \tau_k + \xi_r) \text{ with probability } p_k.$$

It is easy to check that two parameters λ_x and λ_x' will generically result in the same distribution on R if and only if

$$p_k' = p_k, \quad \alpha_r' \mu_k' = \alpha_r \mu_k, \tag{43a}$$

$$(\alpha_r')^2 \tau_k' + \xi_r = \alpha_r^2 \tau_k + \xi_r, \tag{43b}$$

for $k = 1, \ldots, K$. That is, the component means, variances and probabilities must match.

Now, λ_x has $3K$ parameters, so $(\lambda_x, \alpha_r, \xi_r)$ has a total of $3K+2$ parameters. Since (43) has $3K$ constraints, the mapping (41) would in general need two additional constraints to be one-to-one to meet condition (b) of Definition 1. As one example for such constraints, we could know *a priori* that X has a known mean and variance, thereby providing two constraints. Alternatively, we could know that one of the mixtures, say $k = 1$, is strictly zero so that $\mu_1 = \tau_1 = 0$. This requirement would also provide two additional constraints. In either of these two examples, we need no additional constraints on the set S_x to meet the conditions of Lemma 1. Alternatively, if S_x can be restricted in some manner, then we could relax those assumptions.

B. AWGN Output

Now consider an AWGN output channel where $P_{Y|Z}$ is given by

$$Y = Z + W, \quad W \sim \mathcal{N}(0, \tau_w), \tag{44}$$

where W is independent of Z. Here, the unknown parameter is $\lambda_z = \tau_w$. Then, given a covariance matrix $\mathbf{K}_p$, the distribution $P_{Y,P}$ in (36) is given by

$$(Y, P) \sim \mathcal{N}(0, \mathbf{Q}), \quad \mathbf{Q} = \mathbf{K}_p + \begin{bmatrix} \tau_w & 0 \\ 0 & 0 \end{bmatrix},$$

which is uniquely specified by the covariance matrix $\mathbf{Q}$. In this case, if we know the $(1, 1)$-element of $\mathbf{K}_p$, we can determine τ_w from $(1, 1)$ element of $\mathbf{Q}$.

C. Initialization

One case where the covariance matrix $\mathbf{K}_p^t$ could be known is in the initial step of the algorithm. Suppose, for example, that we know the mean and variance of X, the random variable describing the components of $\mathbf{x}$. That is, the mean and variance of variance of the input distribution $P_X(\cdot|\lambda_x)$ is the same for all values of $\lambda_x \in \Lambda_x$. In this case, even though we do not know the value of the parameter, we can perform the initialization in line 1 for the sum-product GAMP algorithm in Algorithm 1. Then, from the state evolution equations in Section II-E, we would then know the initial covariance matrix $\mathbf{K}_p^t$ for $t = 0$ as given in (22).

VI. Numerical Results

A. Estimation of a Gauss-Bernoulli Input

Recent findings [42] suggest that there is considerable value in learning of priors P_X in the context of compressed sensing, which considers the estimation of sparse vectors $\mathbf{x}$ from underdetermined measurements ($m < n$). It is known that estimators such as LASSO offer certain optimal min-max performance over a large class of sparse distributions [43]. However, for many particular distributions, there is a potentially large performance gap between LASSO and MMSE estimator with the correct prior. This gap was the main motivation for the work of Vila and Schniter [24], [25] which showed large gains of the EM-GAMP method due to its ability to learn the prior.

Here, we illustrate the performance and asymptotic consistency of adaptive GAMP in a simple compressed sensing example. Specifically, we consider the estimation of a sparse vector $\mathbf{x} \in \mathbb{R}^n$ from m noisy measurements

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

where the additive noise $\mathbf{w}$ is random with i.i.d. entries $w_i \sim \mathcal{N}(0, \sigma^2)$. Here, the "output" channel is determined by the statistics of $\mathbf{w}$, which are assumed to be known to the estimator. So, there are no unknown parameters λ_z.

As a model for the sparse input vector $\mathbf{x}$, we assumed that the components are i.i.d. with the Gauss-Bernoulli distribution,

$$x_j \sim \begin{cases} 0 & \text{prob } = 1 - \rho, \\ \mathcal{N}(0, \sigma_x^2) & \text{prob } = \rho \end{cases} \tag{45}$$

where ρ represents the probability that the component is non-zero (i.e. the vector's sparsity ratio) and σ_x^2 is the variance of the non-zero components. The parameters $\lambda_x = (\rho, \sigma_x^2)$ are treated as unknown.

Now, the Gaussian mixture in (45) has only two unknown parameters: ρ and σ_x^2. As described in Section V-A, this mixture is sufficiently constrained so that if we apply the full ML estimation in (26) with no restrictions in the set S_x, we can identify the parameters correctly. We thus use this ML adaption in the first iteration and the above theory suggests that the algorithm should recover the correct parameters right away.

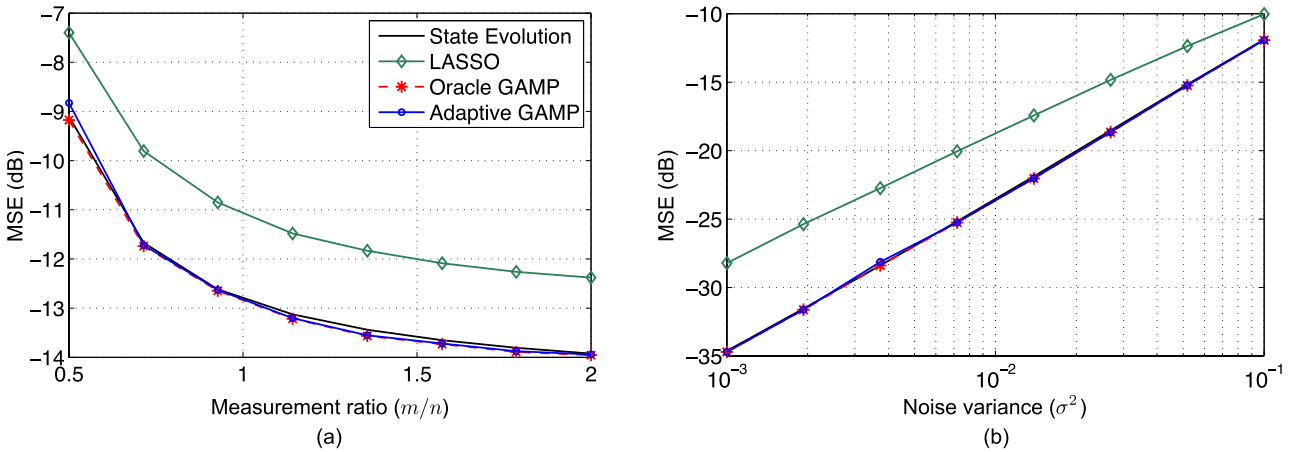

Fig. 2. Reconstruction of a Gauss-Bernoulli signal from noisy measurements. The average reconstruction MSE is plotted against (a) measurement ratio m/n and (b) AWGN variance σ^2. The plots illustrate that adaptive GAMP yields considerable improvement over ℓ_1-based LASSO estimator. Moreover, it matches the performance of oracle GAMP that knows the prior parameters.

However, in our implementation, we continue to update the parameters at all iterations since there may be parameter errors on finite sample sizes. However, to simplify the ML adaptation, we can restrict the set selection S_x for iterations $t > 1$ as follows. Assuming the parameters were selected correctly up to some iteration $t-1$, the adaptive GAMP algorithm should behave the same as an oracle sum-product GAMP algorithm with the correct parameters. Now, as described in Section II-E, for the sum-product GAMP, the SE equations simplify so that $\alpha_r^t = 1$ and $\xi_r^t = \overline{\tau}_r^t$. Thus, the parameters α_r^t and ξ_r^t do not need to be estimated, and (26) conveniently simplifies to

$$H_x(\mathbf{r}, \tau_r) = \underset{\lambda_x \in \Lambda_x}{\arg\max} \left\{ \frac{1}{n} \sum_{j=1}^{n} \log P_R(r_j | \lambda_x, \tau_r) \right\}, \quad (46)$$

where $\Lambda_x = [0,1] \times [0, +\infty)$. In our implementation, we approximate the ML adaptation (46) with the EM update (30), which is run for several iterations. At each iteration of adaptive GAMP, we run iteratively the EM updates either until $\|\widehat{\lambda}_x^t - \widehat{\lambda}_x^{t-1}\|_2^2 / \|\widehat{\lambda}_x^{t-1}\|_2^2 \leq 10^{-4}$ for 3 consecutive iterations, or for a maximum of 200 iterations.

Fig. 2 illustrates the performance of adaptive GAMP on signals of length $n = 400$ generated with the parameters $\lambda_x = (\rho = 0.2, \sigma_x^2 = 5)$. The performance of adaptive GAMP is compared to that of LASSO[1] with MSE optimal regularization parameter, and oracle GAMP that knows the parameters of the prior exactly. For generating the graphs, we performed 1000 random trials by forming the measurement matrix $\mathbf{A}$ from i.i.d. zero-mean Gaussian random variables of variance $1/m$. In Fig. 2(a), we keep the variance of the noise fixed to $\sigma^2 = 0.1$ and plot the average MSE of the reconstruction against the measurement ratio m/n. In Fig. 2(b), we keep the measurement ratio fixed to $m/n = 0.75$ and plot the average MSE of the reconstruction against the noise variance σ^2. For completeness, we also provide the asymptotic MSE values computed via SE recursion. The results illustrate that GAMP significantly outperforms LASSO over the whole range of m/n and σ^2. Moreover, the results corroborate the consistency of adaptive GAMP which nearly achieves the reconstruction quality of oracle GAMP. Note also that in Fig. 2 the average reconstruction times—across all realizations and undersampling rates—were 0.35, 0.06, and 0.22 seconds for LASSO, oracle GAMP, and adaptive GAMP, respectively. The results indicate that adaptive GAMP can be an effective method for estimation when the parameters of the problem are difficult to characterize and must be estimated from data.

[1]For a large-scale implementation of LASSO, we used `l1_ls` package that is readily available online [44].

B. Estimation of a Nonlinear Output Classification Function

As second example, we consider the estimation of the linear-nonlinear-Poisson (LNP) cascade model [8]. The model has been successfully used to characterize neural spike responses in early sensory pathways of the visual system. In the context of the LNP cascade model, the vector $\mathbf{x} \in \mathbb{R}^n$ represents the linear filter, which models the linear receptive field of the neuron. AMP techniques combined with the parameter estimation have been recently proposed for neural receptive field estimation and connectivity detection in [45].

As in Section VI-A, we model $\mathbf{x}$ as a Gauss-Bernoulli vector with unknown parameters $\lambda_x = (\rho, \sigma_x^2)$. To obtain the measurements $\mathbf{y}$, the vector $\mathbf{z} = \mathbf{A}\mathbf{x}$ is passed through a component-wise nonlinearity u specified by

$$u(z) = \frac{1}{1 + \mathrm{e}^{-z}}. \quad (47)$$

The final measurement vector $\mathbf{y}$ is generated by a measurement channel with a conditional density of the form

$$p_{Y|Z}(y_i | z_i, \lambda_z) = \frac{f(z_i)^{y_i}}{y_i!} \mathrm{e}^{-f(z_i)}, \quad (48)$$

where f denotes the nonlinearity given by

$$f(z; \lambda_z) = \exp\left(\sum_{i=1}^{r} \lambda_{z,i} u^{i-1}(z) \right).$$

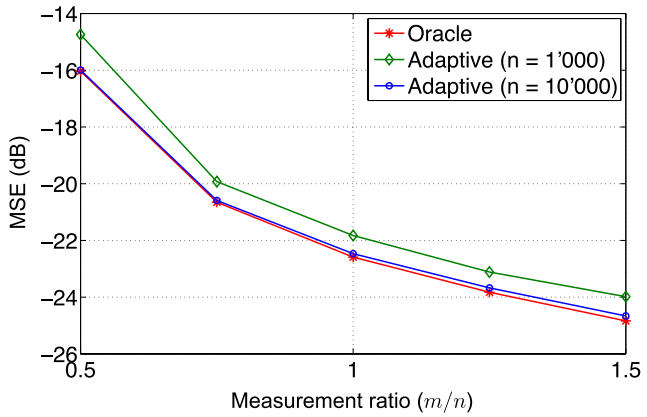

Fig. 3. Identification of linear-nonlinear-Poisson cascade model. The average reconstruction MSE is plotted against the measurement ratio m/n. This plots illustrates near consistency of adaptive GAMP for large n.

Adaptive GAMP can now be used to also estimate vector of polynomial coefficients λ_z, which together with $\mathbf{x}$, completely characterizes the LNP system.

The estimation of λ_z is performed with ML estimator described in Section III-A. We assume that the mean and variance of the vector $\mathbf{x}$ are known at iteration $t = 0$. As discussed in Section V-C, this implies that for sum-product GAMP the covariance $\mathbf{K}_p^0$ is initially known and the optimization (29) simplifies to

$$H_z(\mathbf{p}, \mathbf{y}, \tau_p) = \arg\max_{\lambda_z \in \Lambda_z} \left\{ \frac{1}{m} \sum_{i=1}^{m} \log p_Y(y_i|\lambda_z) \right\}, \qquad (49)$$

where $\Lambda_z \subset \mathbb{R}^r$. The estimation of λ_x is performed as in Section VI-A. As before, for iteration $t > 0$, we assume that the maximizations (46) and (49) yield correct parameter estimates $\widehat{\lambda}_x^t = \lambda_x$ and $\widehat{\lambda}_z^t = \lambda_z$, respectively. Thus we can conclude by induction that for $t > 0$ the adaptive GAMP algorithm should continue matching oracle GAMP for large enough n. In our simulations, we implemented (49) with a gradient ascend algorithm and run it until convergence.

In Fig. 3, we compare the reconstruction performance of adaptive GAMP against the oracle version that knows the true parameters (λ_x, λ_z) exactly. We consider the vector $\mathbf{x}$ generated with true parameters $\lambda_x = (\rho = 0.1, \sigma_x^2 = 30)$. We consider the case $r = 3$ and set the parameters of the output channel to $\lambda_z = [-4.88, 7.41, 2.58]$. To illustrate the asymptotic consistency of the adaptive algorithm, we consider the signals of length $n = 1000$ and $n = 10000$. We perform 10 and 100 random trials for long and short signals, respectively, and plot the average MSE of the reconstruction against m/n. As expected, for large n, the performance of adaptive GAMP is nearly identical (within 0.15) to that of oracle GAMP. For this experiment the average reconstruction times for $n = 1000$ were 120.76 and 1031.5 seconds for oracle and adaptive GAMP, respectively, where the output updates were responsible for the majority of the computation time.

VII. Conclusion

We have presented an adaptive GAMP method for the estimation of i.i.d. vectors $\mathbf{x}$ observed through a known linear transforms followed by an arbitrary, component-wise random transform. The procedure, which is a generalization of EM-GAMP methodology of [24], [25], [27], and [28], estimates both the vector $\mathbf{x}$ as well as parameters in the source and component-wise output transform. In the case of large i.i.d. Gaussian transforms, it is shown that the adaptive GAMP method with ML parameter estimation is provably asymptotically consistent in that the parameter estimates converge to the true values. This convergence result holds over a large class of models with essentially arbitrarily complex parameterizations. Moreover, the algorithm is computationally efficient since it reduces the vector-valued estimation problem to a sequence of scalar estimation problems in Gaussian noise. We believe that this method is applicable to a large class of linear-nonlinear models with provable guarantees can have applications in a wide range of problems. We have mentioned the use of the method for learning sparse priors in compressed sensing.

There are however several limitations that may be addressed in future work. Most significantly, the SE results are currently limited to large i.i.d. matrices. However, many matrices in practice are not well-modeled as large i.i.d. Recent work of ours [46] has attempted to understand the behavior of GAMP in non-asymptotic settings and an avenue of future work is to see if these results can be extended to adaptive GAMP.

Also, a critical assumption in our analysis is that the parameters λ_x and λ_z are finite dimensional and whose dimensions do not grow. Another avenue of work would be see if the methods can be extended to non-parametric estimation of the densities in the adaptation steps or estimation with growing numbers of parameters.

Finally, as we discussed in Section IV-C, the ML adaptation is generally non-convex and thus must often be approximated. An open question is what tractable, approximate methods can be applied while guaranteeing consistency.

Appendix A
Convergence of Empirical Distributions

Bayati and Montanari's analysis in [16] employs certain deterministic models on the vectors and then proves convergence properties of related empirical distributions. To apply the same analysis here, we need to review some of their definitions. We say a function $\phi : \mathbb{R}^r \to \mathbb{R}^s$ is *pseudo-Lipschitz* of order $k > 1$, if there exists an $L > 0$ such for any $\mathbf{x}, \mathbf{y} \in \mathbb{R}^r$,

$$\|\phi(\mathbf{x}) - \phi(\mathbf{y})\| \le L(1 + \|\mathbf{x}\|^{k-1} + \|\mathbf{y}\|^{k-1})\|\mathbf{x} - \mathbf{y}\|.$$

Now suppose that for each $n = 1, 2, \ldots$, $\mathbf{v}^{(n)}$ is a set of vectors

$$\mathbf{v}^{(n)} = \{\mathbf{v}_i(n), i = 1, \ldots, \ell(n)\}, \qquad (50)$$

where each element $\mathbf{v}_i(n) \in \mathbb{R}^s$ and $\ell(n)$ is the number of elements in the set. Thus, $\mathbf{v}^{(n)}$ can itself be regarded as a vector with $s\ell(n)$ components. We say that $\mathbf{v}^{(n)}$ *empirically converges with bounded moments of order* k as $n \to \infty$ to a random vector $\mathbf{V}$ on $\mathbb{R}^s$ if: For all pseudo-Lipschitz continuous functions, ϕ, of order k,

$$\lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} \phi(\mathbf{v}_i(n)) = \mathbb{E}(\phi(\mathbf{V})) < \infty.$$

When the nature of convergence is clear, we may write (with some abuse of notation)

$$\mathbf{v}^{(n)} \overset{\text{PL}(k)}{\to} \mathbf{V} \text{ as } n \to \infty,$$

or

$$\lim_{n\to\infty} \mathbf{v}^{(n)} \overset{\text{PL}(k)}{=} \mathbf{V}.$$

Finally, let $\mathcal{P}_k^s$ be the set of probability distributions on $\mathbb{R}^s$ with bounded kth moments, and suppose that $H : \mathcal{P}_k^s \to \Lambda$ is a functional $\mathcal{P}_k^s$ to some topological space Λ. Given a set $\mathbf{v}^{(n)}$ as in (50), write $H(\mathbf{v})$ for $H(P_{\mathbf{v}})$ where $P_{\mathbf{v}}$ is the empirical distribution on the components of $\mathbf{v}$. Also, given a random vector $\mathbf{V}$ with distribution $P_{\mathbf{V}}$ write $H(\mathbf{V})$ for $H(P_{\mathbf{V}})$. Then, we say that the functional H is *weakly pseudo-Lipschitz continuous* of order k if

$$\lim_{n\to\infty} \mathbf{v}^{(n)} \overset{\text{PL}(k)}{=} \mathbf{V} \Longrightarrow \lim_{n\to\infty} H(\mathbf{v}^{(n)}) = H(\mathbf{V}),$$

where the limit on the right hand side is in the topology of Λ.

APPENDIX B
PROOF OF THEOREM 2

The proof follows along the adaptation argument of [40]. We use the tilde superscript on quantities such as $\tilde{\mathbf{x}}^t, \tilde{\mathbf{r}}^t, \tilde{\tau}_r^t, \tilde{\mathbf{p}}^t, \tilde{\tau}_p^t, \tilde{\mathbf{s}}^t$, and $\tilde{\mathbf{z}}^t$ to denote values generated via a non-adaptive version of the GAMP. The non-adaptive GAMP algorithm has the same initial conditions as the adaptive algorithm (i.e. $\tilde{\mathbf{x}}^0 = \widehat{\mathbf{x}}^0, \tilde{\tau}_p^0 = \tau_p^0, \tilde{\mathbf{s}}^{-1} = \mathbf{s}^{-1} = 0$), but with $\widehat{\lambda}_x^t$ and $\widehat{\lambda}_z^t$ replaced by their deterministic limits $\bar{\lambda}_x^t$ and $\bar{\lambda}_z^t$, respectively. That is, we replace lines 3, 3 and 3 in Algorithm 3 with

$$\tilde{\mathbf{z}}_i^t = G_z^t(p_i^t, y_i, \tau_p^t, \overline{\lambda}_z^t), \qquad \tilde{\mathbf{s}}_i^t = G_s^t(p_i^t, y_i, \tau_p^t, \overline{\lambda}_z^t),$$
$$\tilde{\mathbf{x}}_j^{t+1} = G_x^t(r_j^t, \tau_r^t, \overline{\lambda}_x^t).$$

This non-adaptive algorithm is precisely the standard GAMP method analyzed in [18]. The results in that paper show that the outputs of the non-adaptive algorithm satisfy all the required limits from the SE analysis. That is,

$$\lim_{n\to\infty} \tilde{\theta}_x^t \overset{\text{PL}(k)}{=} \overline{\theta}_x^t, \qquad \lim_{n\to\infty} \tilde{\theta}_z^t \overset{\text{PL}(k)}{=} \overline{\theta}_z^t,$$

where $\tilde{\theta}_x^t$ and $\tilde{\theta}_z^t$ are the sets generated by the non-adaptive GAMP algorithm:

$$\tilde{\theta}_x^t := \left\{(x_j, \tilde{r}_j^t, \tilde{x}_j^{t+1}) : \quad j = 1, \ldots, n\right\},$$
$$\tilde{\theta}_z^t := \left\{(z_i, \tilde{z}_i^t, y_i, \tilde{p}_i^t) : \quad i = 1, \ldots, m\right\}.$$

The limits (34) are now proven through a continuity argument that shows that the adaptive and non-adaptive quantities must asymptotically agree with one another. Specifically, we will start by proving that the following limits holds almost surely for all $t \geq 0$

$$\lim_{n\to\infty} \Delta_x^t = \lim_{n\to\infty} \frac{1}{n}\|\widehat{\mathbf{x}}^t - \tilde{\mathbf{x}}^t\|_k^k = 0 \tag{51a}$$
$$\lim_{n\to\infty} \Delta_{\tau_p}^t = \lim_{n\to\infty} |\tau_p^t - \tilde{\tau}_p^t| = 0 \tag{51b}$$

where $\|\cdot\|_k$ is usual the k-norm. Moreover, in the course of proving (51), we will also show that the following limits hold almost surely

$$\lim_{m\to\infty} \Delta_p^t = \lim_{m\to\infty} \frac{1}{m}\|\mathbf{p}^t - \tilde{\mathbf{p}}^t\|_k^k = 0, \tag{52a}$$
$$\lim_{n\to\infty} \Delta_r^t = \lim_{n\to\infty} \frac{1}{n}\|\mathbf{r}^t - \tilde{\mathbf{r}}^t\|_k^k = 0, \tag{52b}$$
$$\lim_{m\to\infty} \Delta_s^t = \lim_{m\to\infty} \frac{1}{m}\|\mathbf{s}^t - \tilde{\mathbf{s}}^t\|_k^k = 0, \tag{52c}$$
$$\lim_{m\to\infty} \Delta_z^t = \lim_{m\to\infty} \frac{1}{m}\|\widehat{\mathbf{z}}^t - \tilde{\mathbf{z}}^t\|_k^k = 0, \tag{52d}$$
$$\lim_{n\to\infty} \Delta_{\tau_r}^t = \lim_{n\to\infty} |\tau_r^t - \tilde{\tau}_r^t| = 0, \tag{52e}$$
$$\lim_{n\to\infty} \widehat{\lambda}_x^t = \bar{\lambda}_x^t, \tag{52f}$$
$$\lim_{n\to\infty} \widehat{\lambda}_z^t = \bar{\lambda}_z^t, \tag{52g}$$

The proof of the limits (51) and (52) is achieved by an induction on t. Although we only need to show the above limits for $k = 2$, most of the arguments hold for arbitrary $k \geq 2$. We thus present the general derivation where possible.

To begin the induction argument, first note that the non-adaptive algorithm has the same initial conditions as the adaptive algorithm. Thus the limits (51) and (52c) hold for $t = 0$ and $t = -1$, respectively.

We now proceed by induction. Suppose that $t \geq 0$ and the limits (51) and (52c) hold for some t and $t-1$, respectively. Since $\mathbf{A}$ has i.i.d. components with zero mean and variance $1/m$, it follows from the Marčenko-Pastur Theorem [47] that its 2-norm operator norm is bounded. That is, there exists a constant C_A such that almost surely we have

$$\lim_{n\to\infty} \|\mathbf{A}\|_k \leq C_A, \ \lim_{n\to\infty} \|\mathbf{A}^T\|_k \leq C_A. \tag{53}$$

This bound is the only part of the proof that specifically requires $k = 2$. From (53), we obtain

$$\begin{aligned}
\|\mathbf{p}^t - \tilde{\mathbf{p}}^t\|_k &= \|\mathbf{A}\widehat{\mathbf{x}}^t - \tau_p^t \mathbf{s}^{t-1} - \mathbf{A}\tilde{\mathbf{x}}^t + \tilde{\tau}_p^t \tilde{\mathbf{s}}^{t-1}\|_k \\
&= \|\mathbf{A}(\widehat{\mathbf{x}}^t - \tilde{\mathbf{x}}^t) + \tau_p^t(\tilde{\mathbf{s}}^{t-1} - \mathbf{s}^{t-1}) + (\tilde{\tau}_p^t - \tau_p^t)\tilde{\mathbf{s}}^{t-1}\|_k \\
&\leq \|\mathbf{A}(\widehat{\mathbf{x}}^t - \tilde{\mathbf{x}}^t)\|_k + |\tau_p^t|\|\tilde{\mathbf{s}}^{t-1} - \mathbf{s}^{t-1}\|_k + |\tilde{\tau}_p^t - \tau_p^t|\|\tilde{\mathbf{s}}^{t-1}\|_k \\
&\overset{\text{(a)}}{\leq} \|\mathbf{A}\|_k\|\widehat{\mathbf{x}}^t - \tilde{\mathbf{x}}^t\|_k + |\tau_p^t|\|\tilde{\mathbf{s}}^{t-1} - \mathbf{s}^{t-1}\|_k + |\tilde{\tau}_p^t - \tau_p^t|\|\tilde{\mathbf{s}}^{t-1}\|_k \\
&\leq C_A\|\widehat{\mathbf{x}}^t - \tilde{\mathbf{x}}^t\|_k + |\tau_p^t|\|\mathbf{s}^{t-1} - \tilde{\mathbf{s}}^{t-1}\|_k + |\tau_p^t - \tilde{\tau}_p^t|\|\tilde{\mathbf{s}}^{t-1}\|_k
\end{aligned} \tag{54}$$

almost surely, where (a) is due to the norm inequality $\|\mathbf{A}\mathbf{x}\|_k \leq \|\mathbf{A}\|_k\|\mathbf{x}\|_k$. Since $k \geq 1$, we have that for any positive numbers a and b

$$(a+b)^k \leq 2^k(a^k + b^k). \tag{55}$$

Applying the inequality (55) into (54), we obtain

$$\begin{aligned}
&\frac{1}{m}\|\mathbf{p}^t - \tilde{\mathbf{p}}^t\|_k^k \\
&\leq \frac{1}{m}\left(C_A\|\widehat{\mathbf{x}}^t - \tilde{\mathbf{x}}^t\|_k + |\tau_p^t|\|\mathbf{s}^{t-1} - \tilde{\mathbf{s}}^{t-1}\|_k + \Delta_{\tau_p}^t\|\tilde{\mathbf{s}}^{t-1}\|_k\right)^k \\
&\leq 2^k C_A \frac{n}{m}\Delta_x^t + 2^k|\tau_p^t|^k\Delta_s^{t-1} + 2^k(\Delta_{\tau_p}^t)^k\left(\frac{1}{m}\|\tilde{\mathbf{s}}^{t-1}\|_k^k\right).
\end{aligned} \tag{56}$$

Now, since $\tilde{\mathbf{s}}^t$ and $\tilde{\tau}_p^t$ are the outputs of the non-adaptive algorithm, they satisfy the limits

$$\lim_{n\to\infty}\frac{1}{m}\|\tilde{\mathbf{s}}^t\|_k^k = \lim_{n\to\infty}\frac{1}{m}\sum_{i=1}^m |\tilde{s}_i^t|^k = \mathbb{E}\left[|S^t|^k\right] < \infty, \quad (57a)$$

$$\lim_{n\to\infty}\tilde{\tau}_p^t = \overline{\tau}_p^t < \infty. \quad (57b)$$

Now, the induction hypotheses state that Δ_x^t, Δ_s^{t-1} and $\Delta_{\tau_p}^t \to 0$. Applying these along the bounds (57a), and the fact that $n/m \to \beta$, we obtain (52a).

To establish (52g), we first prove the empirical convergence of $(\mathbf{p}^t, \mathbf{y})$ to (P^t, Y). Towards this end, let $\phi(p, y)$ be any pseudo-Lipschitz continuous function ϕ of order k. Then

$$\begin{aligned}
&\left|\frac{1}{m}\sum_{i=1}^m \phi(p_i^t, y_i) - \mathbb{E}\left[\phi(P^t, Y)\right]\right| \\
&\quad\le \frac{1}{m}\sum_{i=1}^m \left|\phi(p_i^t, y_i) - \phi(\tilde{p}_i^t, y_i)\right| \\
&\qquad + \left|\frac{1}{m}\sum_{i=1}^m \phi(\tilde{p}_i^t, y_i) - \mathbb{E}\left[\phi(P^t, Y)\right]\right| \\
&\quad\overset{(a)}{\le} \frac{L}{m}\sum_{i=1}^m \left(1 + |p_i^t|^{k-1} + |\tilde{p}_i^t|^{k-1} + |y_i|^{k-1}\right)|p_i^t - \tilde{p}_i^t| \\
&\qquad + \left|\frac{1}{m}\sum_{i=1}^m \phi(\tilde{p}_i^t, y_i) - \mathbb{E}\left[\phi(P^t, Y)\right]\right| \\
&\quad\overset{(b)}{\le} LC\Delta_p^t + \left|\frac{1}{m}\sum_{i=1}^m \phi(\tilde{p}_i^t, y_i) - \mathbb{E}\left[\phi(P^t, Y)\right]\right|. \quad (58)
\end{aligned}$$

In (a) we use the fact that ϕ is pseudo-Lipschitz, and in (b) we use Hölder's inequality $|\widehat{\mathbf{x}}^T\mathbf{y}| = \|\mathbf{x}\|_k\|\mathbf{y}\|_q$ with $q = p/(p-1)$. The constant is defined as

$$\begin{aligned}
C &:= \left[\frac{1}{m}\sum_{i=1}^m \left(1 + |p_i^t|^{k-1} + |\tilde{p}_i^t|^{k-1} + |y_i|^{k-1}\right)\right]^{k/(k-1)} \\
&\le \frac{1}{m}\sum_{i=1}^m \left(1 + |p_i^t|^{k-1} + |\tilde{p}_i^t|^{k-1} + |y_i|^{k-1}\right)^{k/(k-1)} \\
&\le \text{const}\times\left[1 + \left(\frac{1}{m}\left\|\mathbf{p}^t\right\|_k^k\right)^{\frac{k-1}{k}}\right. \\
&\qquad \left. + \left(\frac{1}{m}\left\|\tilde{\mathbf{p}}^t\right\|_k^k\right)^{\frac{k-1}{k}} + \left(\frac{1}{m}\|\mathbf{y}\|_k^k\right)^{\frac{k-1}{k}}\right], \quad (59)
\end{aligned}$$

where the first step is from Jensen's inequality. Since $(\tilde{\mathbf{p}}^t, \mathbf{y})$ satisfy the limits for the non-adaptive algorithm, we have:

$$\lim_{n\to\infty}\frac{1}{m}\|\tilde{\mathbf{p}}^t\|_k^k = \lim_{n\to\infty}\frac{1}{m}\sum_{i=1}^m |\tilde{p}_i^t|^k = \mathbb{E}\big[|P^t|^k\big] < \infty \quad (60a)$$

$$\lim_{n\to\infty}\frac{1}{m}\|\mathbf{y}\|_k^k = \lim_{n\to\infty}\frac{1}{m}\sum_{i=1}^m |y_i|^k = \mathbb{E}\big[|Y|^k\big] < \infty \quad (60b)$$

Also, from the induction hypothesis (52a), it follows that the adaptive output must satisfy the same limit

$$\lim_{n\to\infty}\frac{1}{m}\|\mathbf{p}^t\|_k^k = \lim_{n\to\infty}\frac{1}{m}\sum_{i=1}^m |p_i^t|^k = \mathbb{E}\big[|P^t|^k\big] < \infty. \quad (61)$$

Combining (58), (59), (60), (61), (52a) we conclude that for all $t \ge 0$

$$\lim_{n\to\infty}(\mathbf{p}^t, \mathbf{y}) \overset{\text{PL}(k)}{=} (P^t, Y). \quad (62)$$

The limit (62) along with (51b) and the continuity condition on H_z^t in Assumption 1(d) prove the limit in (52g).

The limit (52a) together with continuity conditions on G_z^t in Assumptions 1 show that (52c), (52d) and (52e) hold for t. For example, to show (52d), we consider the limit $m \to \infty$ of the following expression

$$\begin{aligned}
\frac{1}{m}\|\widehat{\mathbf{z}}^t - \tilde{\mathbf{z}}^t\|_k^k &= \frac{1}{m}\|G_z^t(\mathbf{p}^t, \mathbf{y}, \tau_p^t, \widehat{\lambda}_z^t) - G_z^t(\tilde{\mathbf{p}}^t, \mathbf{y}, \tau_p^t, \bar{\lambda}_z^t)\|_k^k \\
&\overset{(a)}{\le} \frac{L}{m}\|\mathbf{p}^t - \tilde{\mathbf{p}}^t\|_k^k = L\Delta_p^t,
\end{aligned}$$

where at (a) we used the Lipschitz continuity assumption. Similar arguments can be used for (52c) and (52e).

To prove (52b), we proceed exactly as for (52a). Due to the continuity assumptions on H_x, this limit in turn shows that (52f) holds almost surely. Then, (51a) and (51b) follow directly from the continuity of G_x in Assumptions 1, together with (52b) and (52f). We have thus shown that if the limits (51) and (52) hold for some t, they hold for $t+1$. Thus, by induction they hold for all t.

Finally, to establish (34), let ϕ be any pseudo-Lipschitz continuous function $\phi(x, r, \widehat{x})$, and define

$$\epsilon^t := \left|\tfrac{1}{n}\sum_{j=1}^m \phi(x_j, \tilde{r}_j^t, \tilde{x}_j^{t+1}) - \mathbb{E}\left[\phi(X, R^t, \widehat{X}^{t+1})\right]\right|, \quad (63)$$

which, due to convergence of non-adaptive GAMP, can be made arbitrarily small by choosing n large enough. Then, consider

$$\begin{aligned}
&\left|\frac{1}{n}\sum_{j=1}^m \phi(x_j, \hat{r}_j^t, \widehat{x}_j^{t+1}) - \mathbb{E}\left[\phi(X, R^t, \widehat{X}^{t+1})\right]\right| \\
&\le \epsilon_n^t + \frac{1}{n}\sum_{j=1}^n \left|\phi(x_j, \hat{r}_j^t, \widehat{x}_j^{t+1}) - \phi(x_j, \tilde{r}_j^t, \tilde{x}_j^{t+1})\right| \\
&\overset{(a)}{\le} \epsilon_n^t + L\|\mathbf{r}^t - \tilde{\mathbf{r}}^t\|_1 + L\|\hat{\mathbf{x}}^{t+1} - \tilde{\mathbf{x}}^{t+1}\|_1 \\
&\quad + \frac{L'}{n}\sum_{j=1}^n \left(|\hat{r}_j^t|^{k-1} + |\tilde{r}_j^t|^{k-1}\right)\left(|\hat{r}_j^t - \tilde{r}_j^t| + |\widehat{x}_j^{t+1} - \tilde{x}_j^{t+1}|\right) \\
&\quad + \frac{L'}{n}\sum_{j=1}^n \left(|\widehat{x}_j^{t+1}|^{k-1} + |\tilde{x}_j^{t+1}|^{k-1}\right)\left(|\hat{r}_j^t - \tilde{r}_j^t| + |\widehat{x}_j^{t+1} - \tilde{x}_j^{t+1}|\right) \\
&\overset{(b)}{\le} \epsilon_n^t + L\left(\Delta_r^t\right)^{\frac{1}{k}} + L\left(\Delta_x^t\right)^{\frac{1}{k}} \\
&\quad + L'\left(\Delta_r^t\right)^{\frac{1}{k}}\left((\tilde{M}_x^{t+1})^{\frac{k-1}{k}} + (\hat{M}_x^{t+1})^{\frac{k-1}{k}} + (\tilde{M}_r^t)^{\frac{k-1}{k}} + (\hat{M}_r^t)^{\frac{k-1}{k}}\right) \\
&\quad + L'\left(\Delta_x^t\right)^{\frac{1}{k}}\left((\tilde{M}_x^{t+1})^{\frac{k-1}{k}} + (\hat{M}_x^{t+1})^{\frac{k-1}{k}} + (\tilde{M}_r^t)^{\frac{k-1}{k}} + (\hat{M}_r^t)^{\frac{k-1}{k}}\right) \quad (64)
\end{aligned}$$

where L, L' are constants independent of n and

$$\hat{M}_x^{t+1} := \frac{1}{n}\left\|\hat{\mathbf{x}}^{t+1}\right\|_k^k, \quad \hat{M}_r^t := \frac{1}{n}\left\|\mathbf{r}^t\right\|_k^k,$$
$$\tilde{M}_x^{t+1} := \frac{1}{n}\left\|\tilde{\mathbf{x}}^{t+1}\right\|_k^k, \quad \tilde{M}_r^t := \frac{1}{n}\left\|\tilde{\mathbf{r}}^t\right\|_k^k$$

In (a) we use the fact that ϕ is pseudo-Lipshitz, in (b) we use ℓ_p-norm equivalence $\|\mathbf{x}\|_1 \leq n^{1-1/p}\|\mathbf{x}\|_k$ and Hölder's inequality $|\widehat{\mathbf{x}}^T\mathbf{y}| = \|\mathbf{x}\|_k\|\mathbf{y}\|_q$ with $q = p/(p-1)$. By applying of (51a), (52b) and since, $\hat{M}_x^{t+1}$, $\tilde{M}_x^{t+1}$, $\hat{M}_r^t$, and $\tilde{M}_r^t$ converge to a finite value we can obtain the first equation of (34) by taking $n \to \infty$. The second equation in (34) can be shown in a similar way. This proves the limits (34).

Also, the first two limits in (35) are a consequence of (52f) and (52f). The second two limits follow from continuity assumptions in Assumption 1(e) and the convergence of the empirical distributions in (34). This completes the proof.

APPENDIX C
PROOF OF THEOREM 3

Part (a) of Theorem 3 is an application of Theorem 2. To apply this general result, first observe that Assumptions 3(a) and (c) immediately imply the corresponding items in Assumptions 2. So, we only need to verify the continuity condition in Assumption 2(b) for the adaptation functions in (26) and (29).

We begin by proving the continuity of H_z^t. Fix t, and let $(\mathbf{y}^{(n)}, \mathbf{p}^{(n)})$ be a sequence of vectors and $\tau_p^{(n)}$ be a sequence of scalars such that

$$\lim_{n\to\infty}(\mathbf{y}^{(n)}, \mathbf{p}^{(n)}) \overset{\text{PL}(p)}{=} (Y, P^t) \qquad \lim_{n\to\infty}\tau_p^{(n)} = \overline{\tau}_p^t, \tag{65}$$

where (Y, P^t) and $\overline{\tau}_p^t$ are the outputs of the state evolution equations. For each n, let

$$\widehat{\lambda}_z^{(n)} := H_z^t(\mathbf{y}^{(n)}, \mathbf{p}^{(n)}, \tau_p^{(n)}). \tag{66}$$

We wish to show that $\widehat{\lambda}_z^{(n)} \to \lambda_z^*$, the true parameter. Since $\widehat{\lambda}_z^{(n)} \in \Lambda_z$ and Λ_z is compact, it suffices to show that any limit point of any convergent subsequence is equal to λ_z^*. So, suppose that $\widehat{\lambda}_z^{(n)} \to \widehat{\lambda}_z$ to some limit point $\widehat{\lambda}_z$ on some subsequence $\widehat{\lambda}_z^{(n)}$.

From $\widehat{\lambda}_z^{(n)}$ and the definition (29), it follows that

$$\frac{1}{m}\sum_{i=1}^m \phi_z(p_i^{(n)}, y_i^{(n)}, \widehat{\lambda}_z^{(n)}, \mathbf{K}_p) \geq \frac{1}{m}\sum_{i=1}^m \phi_z(p_i^{(n)}, y_i^{(n)}, \lambda_z^*, \mathbf{K}_p), \tag{67}$$

where $\mathbf{K}_p \in S_z(\tau_p^{(n)})$ is the solution of the first maximization of (29). Now, since $\tau_p^{(n)} \to \overline{\tau}_p^t$ and $\widehat{\lambda}_z^{(n)} \to \widehat{\lambda}_z$, we apply the continuity condition in Definition 2(c) to obtain

$$\liminf_{n\to\infty}\frac{1}{m}\sum_{i=1}^m\Big[\phi_z(p_i^{(n)}, y_i^{(n)}, \widehat{\lambda}_z, \mathbf{K}_p) - \phi_z(p_i^{(n)}, y_i^{(n)}, \lambda_z^*, \mathbf{K}_p)\Big] \geq 0. \tag{68}$$

Also, the limit (65) and the fact that ϕ_z is pseuedo-Lipschitz continuous of order k implies that

$$\mathbb{E}[\phi_z(P^t, Y, \widehat{\lambda}_z, \mathbf{K}_p)] \geq \mathbb{E}[\phi_z(P^t, Y, \lambda_z^*, \mathbf{K}_p)]. \tag{69}$$

But, Property (b) of Definition 2 shows that λ_z^* is the unique maxima of the right-hand side, so

$$\mathbb{E}[\phi_z(P^t, Y, \widehat{\lambda}_z, \mathbf{K}_p)] = \mathbb{E}[\phi_z(P^t, Y, \lambda_z^*, \mathbf{K}_p)], \tag{70}$$

with $\widehat{\lambda}_z = \lambda_z^*$. Since this limit point is the same for all convergent subsequences, we see that $\widehat{\lambda}_z^{(n)} \to \lambda_z^*$ over the entire sequence. We have thus shown that given limits (65), the outputs of the adaptation function converge as

$$H_z^t(\mathbf{y}^{(n)}, \mathbf{p}^{(n)}, \tau_p^{(n)}) = \widehat{\lambda}_z^{(n)} \to \lambda_z^* = H_z^t(Y, P^t, \overline{\tau}_p^t).$$

Thus, the continuity condition on H_z^t in Assumption 2(b) is satisfied. The analogous continuity condition on H_x^t can be proven in a similar manner.

Therefore, all the conditions of Assumption 2 are satisfied and we can apply Theorem 2. Part (a) of Theorem 3 immediately follows from Theorem 2.

So, it remains to show parts (b) and (c) of Theorem 3. We will only prove (b); the proof of (c) is similar. Also, since we have already established (35), we only need to show that the output of the SE equations matches the true parameter. That is, we need to show $\overline{\lambda}_x^t = \lambda_x^*$. This fact follows immediately from the selection of the adaptation functions:

$$\begin{aligned}
\overline{\lambda}_x^t &\overset{(a)}{=} H_x^t(R^t, \overline{\tau}_r^t) \\
&\overset{(b)}{=} \arg\max_{\lambda_x\in\Lambda_x}\max_{(\alpha_r,\xi_r)\in S_x(\overline{\tau}_r^t)} \mathbb{E}\left[\phi_x(R^t, \lambda_x, \alpha_r, \xi_r)\right] \\
&\overset{(c)}{=} \arg\max_{\lambda_x\in\Lambda_x}\max_{(\alpha_r,\xi_r)\in S_x(\overline{\tau}_r^t)} \mathbb{E}\left[\phi_x(\alpha_r^t X + V^t, \lambda_x, \alpha_r, \xi_r)|\lambda_x^*, \xi_r^t\right] \qquad (71)\\
&\overset{(d)}{=} \overline{\lambda}_x^* \qquad (72)
\end{aligned}$$

where (a) follows from the SE equation (33a); (b) is the definition of the ML adaptation function H_x^t when interpreted as a functional on a random variable R^t; (c) is the definition of the random variable R^t in (16) where $V^t \sim \mathcal{N}(0, \xi_r^t)$; and (d) follows from Definition 1(b) and the hypothesis that $(\alpha_r^*, \xi_r^*) \in S_x(\overline{\tau}_r^t)$. Thus, we have proven that $\overline{\lambda}_x^t = \lambda_x^*$, and this completes the proof of part (b) of Theorem 3. The proof of part (c) is similar.

APPENDIX D
PROOF OF LEMMA 1

We will just prove part (a). The proof of (b) is similar. Suppose that $R \sim P_R(\cdot|\lambda_x^*, \alpha_r^*, \xi_r^*)$ for some "true" parameter λ_x^* and (α_r^*, ξ_r^*). Let

$$L(\lambda_x, \alpha_r, \xi_r) := \mathbb{E}\left[\phi_x(R, \lambda_x, \alpha_r, \xi_r) \mid \lambda_x^*, \alpha_r^*, \xi_r^*\right],$$

be the expected value of ϕ_x under the true parameters for R. According to Definition 1(b), we need to show that $L(\cdot)$ is maximized uniquely at $(\lambda_x^*, \alpha_r^*, \xi_r^*)$. To this end, consider any other parameter set $(\lambda_x, \alpha_r, \xi_r)$. Then, if $\phi_x(\cdot)$ is the log-likelihood function in (24),

$$\begin{aligned}
&L(\lambda_x^*, \alpha_r^*, \xi_r^*) - L(\lambda_x, \alpha_r, \xi_r) \\
&\quad\overset{(a)}{=} \mathbb{E}\left[\phi_x(R, \lambda_x^*, \alpha_r^*, \xi_r^*) \mid \lambda_x^*, \alpha_r^*, \xi_r^*\right] - \mathbb{E}\left[\phi_x(R, \lambda_x, \alpha_r, \xi_r) \mid \lambda_x^*, \alpha_r^*, \xi_r^*\right] \\
&\quad\overset{(b)}{=} \mathbb{E}\left[\log P_R(R|\lambda_x^*, \alpha_r^*, \xi_r^*) \mid \lambda_x^*, \alpha_r^*, \xi_r^*\right] - \mathbb{E}\left[\log P_R(R|\lambda_x, \alpha_r, \xi_r) \mid \lambda_x^*, \alpha_r^*, \xi_r^*\right] \\
&\quad\overset{(c)}{=} D\Big(P_R(\cdot|\lambda_x^*, \alpha_r^*, \xi_r^*), P_R(\cdot|\lambda_x, \alpha_r, \xi_r)\Big) \qquad (73)
\end{aligned}$$

where (a) follows from the definition of $L(\cdot)$; (b) follows from the fact that $\phi_x(\cdot)$ is the log likelihood in (24) and (c) is the Kullback-Liebler divergence. Now, if

$$(\lambda_x^*, \alpha_r^*, \xi_r^*) \neq (\lambda_x, \alpha_r, \xi_r),$$

the hypothesis that the map (41) is one-to-one implies that the two distributions in (73) are not equal. Therefore, the Kullback-Liebler divergence will be strictly positive [48] and thus the function $L(\cdot)$ is uniquely maximized at $(\lambda_x^*, \alpha_r^*, \xi_r^*)$.

REFERENCES

[1] M. Tipping, "Sparse Bayesian learning and the relevance vector machine," *J. Mach. Learn. Res.*, vol. 1, pp. 211–244, Sep. 2001.

[2] M. West, "Bayesian factor regression models in the 'large p, small n' paradigm," *Bayesian Statist.*, vol. 7, pp. 723–732, 2003.

[3] D. Wipf and B. Rao, "Sparse Bayesian learning for basis selection," *IEEE Trans. Signal Process.*, vol. 52, no. 8, pp. 2153–2164, Aug. 2004.

[4] S. Ji, Y. Xue, and L. Carin, "Bayesian compressive sensing," *IEEE Trans. Signal Process.*, vol. 56, no. 6, pp. 2346–2356, Jun. 2008.

[5] V. Cevher, "Learning with compressible priors," in *Proc. NIPS*, Vancouver, BC, Canada, Dec. 2009, pp. 261–269.

[6] S. Billings and S. Fakhouri, "Identification of systems containing linear dynamic and static nonlinear elements," *Automatica*, vol. 18, no. 1, pp. 15–26, 1982.

[7] I. W. Hunter and M. J. Korenberg, "The identification of nonlinear biological systems: Wiener and Hammerstein cascade models," *Biol. Cybern.*, vol. 55, nos. 2–3, pp. 135–144, 1986.

[8] O. Schwartz, J. W. Pillow, N. C. Rust, and E. P. Simoncelli, "Spike-triggered neural characterization," *J. Vis.*, vol. 6, no. 4, pp. 484–507, Jul. 2006.

[9] J. Boutros and G. Caire, "Iterative multiuser joint decoding: Unified framework and asymptotic analysis," *IEEE Trans. Inform. Theory*, vol. 48, no. 7, pp. 1772–1793, Jul. 2002.

[10] T. Tanaka and M. Okada, "Approximate belief propagation, density evolution, and neurodynamics for CDMA multiuser detection," *IEEE Trans. Inform. Theory*, vol. 51, no. 2, pp. 700–706, Feb. 2005.

[11] D. Guo and C.-C. Wang, "Asymptotic mean-square optimality of belief propagation for sparse linear systems," in *Proc. IEEE Inf. Theory Workshop*, Chengdu, China, Oct. 2006, pp. 194–198.

[12] D. Guo and C.-C. Wang, "Random sparse linear systems observed via arbitrary channels: A decoupling principle," in *Proc. IEEE Int. Symp. Inf. Theory*, Nice, France, Jun. 2007, pp. 946–950.

[13] D. L. Donoho, A. Maleki, and A. Montanari, "Message-passing algorithms for compressed sensing," *Proc. Nat. Acad. Sci. United States Amer.*, vol. 106, no. 45, pp. 18914–18919, Nov. 2009.

[14] D. L. Donoho, A. Maleki, and A. Montanari, "Message passing algorithms for compressed sensing I: Motivation and construction," in *Proc. ITW*, Jan. 2010, pp. 1–5.

[15] D. L. Donoho, A. Maleki, and A. Montanari, "Message passing algorithms for compressed sensing II: Analysis and validation," in *Proc. ITW*, Jan. 2010, pp. 1–5.

[16] 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.

[17] S. Rangan, "Estimation with random linear mixing, belief propagation and compressed sensing," in *Proc. Conf. Inf. Sci. Sys.*, Princeton, NJ, USA, Mar. 2010, pp. 1–6.

[18] S. Rangan, "Generalized approximate message passing for estimation with random linear mixing," in *Proc. IEEE ISIT*, Saint Petersburg, Russia, Jul./Aug. 2011, pp. 2174–2178.

[19] U. S. Kamilov, V. K. Goyal, and S. Rangan, "Message-passing dequantization with applications to compressed sensing," *IEEE Trans. Signal Process.*, vol. 60, no. 12, pp. 6270–6281, Dec. 2012.

[20] A. Montanari, "Graphical model concepts in compressed sensing," in *Compressed Sensing: Theory and Applications*, Y. C. Eldar and G. Kutyniok, Eds. Cambridge, U.K.: Cambridge Univ. Press, Jun. 2012, pp. 394–438.

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

[22] M. Seeger, "Bayesian inference and optimal design for the sparse linear model," *J. Mach. Learn. Res.*, vol. 9, pp. 759–813, Sep. 2008.

[23] A. Javanmard and A. Montanari, "State evolution for general approximate message passing algorithms, with applications to spatial coupling," Nov. 2012, arXiv:1211.5164 [math.PR].

[24] J. P. Vila and P. Schniter, "Expectation-maximization Bernoulli-Gaussian approximate message passing," in *Proc. Conf. Rec. 45th Asilomar Conf. Signals, Syst. Comput.*, Pacific Grove, CA, USA, Nov. 2011, pp. 799–803.

[25] J. P. Vila and P. Schniter, "Expectation-maximization Gaussian-mixture approximate message passing," in *Proc. Conf. Inf. Sci. Sys.*, Princeton, NJ, USA, Mar. 2012, pp. 1–6.

[26] U. S. Kamilov, S. Rangan, A. K. Fletcher, and M. Unser, "Approximate message passing with consistent parameter estimation and applications to sparse learning," in *Proc. NIPS*, Lake Tahoe, NV, USA, Dec. 2012, pp. 2447–2455.

[27] F. Krzakala, M. Mézard, F. Sausset, Y. Sun, and L. Zdeborová, "Statistical physics-based reconstruction in compressed sensing," *Phys. Rev.*, vol. 2, no. 2, p. 021005, Sep. 2011.

[28] F. Krzakala, M. Mézard, F. Sausset, Y. Sun, and L. Zdeborová, "Probabilistic reconstruction in compressed sensing: Algorithms, phase diagrams, and threshold achieving matrices," *J. Statist. Mech., Theory Experim.*, vol. 2012, no. 8, p. P08009, Jun. 2012.

[29] A. Dempster, N. M. Laird, and D. B. Rubin, "Maximum-likelihood from incomplete data via the EM algorithm," *J. R. Statist. Soc.*, vol. 39, no. 1, pp. 1–17, 1977.

[30] 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.

[31] S. Rangan, A. K. Fletcher, V. K. Goyal, and P. Schniter, "Hybrid generalized approximation message passing with applications to structured sparsity," in *Proc. IEEE ISIT*, Cambridge, MA, USA, Jul. 2012, pp. 1241–1245.

[32] D. L. Donoho and I. M. Johnstone, "Minimax risk over ℓ_p-balls for ℓ_q-error," *Probab. Theory Rel. Fields*, vol. 99, no. 4, pp. 277–303, Jun. 1994.

[33] R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. R. Statist. Soc., Ser. B*, vol. 58, no. 1, pp. 267–288, 1996.

[34] A. Montanari, "Graphical models concepts in compressed sensing," Mar. 2011, arXiv:1011.4328 [cs.IT].

[35] D. Baron, S. Sarvotham, and R. G. Baraniuk, "Bayesian compressive sensing via belief propagation," *IEEE Trans. Signal Process.*, vol. 58, no. 1, pp. 269–280, Jan. 2010.

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

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

[38] A. Tulino, G. Caire, S. Shamai, and S. Verdú, "Support recovery in compressed sensing: Information-theoretic bounds," in *Proc. UCSD Workshop Inf. Theory Appl.*, La Jolla, CA, USA, Feb. 2011.

[39] S. Rangan, A. Fletcher, and V. K. Goyal, "Asymptotic analysis of MAP estimation via the replica method and applications to compressed sensing," *IEEE Trans. Inf. Theory*, vol. 58, no. 3, pp. 1902–1923, Mar. 2012.

[40] S. Rangan and A. K. Fletcher, "Iterative estimation of constrained rank-one matrices in noise," in *Proc. IEEE ISIT*, Cambridge, MA, USA, Jul. 2012, pp. 1246–1250.

[41] S. Rangan, P. Schniter, E. Riegler, A. K. Fletcher, and V. Cevher, "Fixed points of generalized approximate message passing with arbitrary matrices," in *Proc. IEEE ISIT*, Istambul, Turkey, Jul. 2013, pp. 664–668.

[42] E. J. Candès and T. Tao, "Near-optimal signal recovery from random projections: Universal encoding strategies?" *IEEE Trans. Inf. Theory*, vol. 52, no. 12, pp. 5406–5425, Dec. 2006.

[43] D. Donoho, I. Johnstone, A. Maleki, and A. Montanari, "Compressed sensing over ℓ^p-balls: Minimax mean square error," in *Proc. IEEE ISIT*, St. Petersburg, Russia, Jun. 2011, pp. 129–133.

[44] S. Kim, K. Koh, M. Lustig, S. Boyd, and D. Gorinvesky, "An interior point method for large-scale ℓ_1-regularized least squares," *IEEE J. Sel. Topics Signal Process.*, vol. 1, no. 4, pp. 606–617, Dec. 2007.

[45] A. K. Fletcher, S. Rangan, L. Varshney, and A. Bhargava, "Neural reconstruction with approximate message passing (NeuRAMP)," in *Proc. NIPS*, Granada, Spain, Dec. 2011, pp. 2555–2563.

[46] S. Rangan, P. Schniter, E. Riegler, A. Fletcher, and V. Cevher, "Fixed points of generalized approximate message passing with arbitrary matrices," in *Proc. IEEE ISIT*, Jul. 2013, pp. 664–668.
[47] V. A. Marčenko and L. A. Pastur, "Distribution of eigenvalues for some sets of random matrices," *Math. USSR–Sbornik*, vol. 1, no. 4, pp. 457–483, 1967.
[48] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York, NY, USA: Wiley, 1991.

Ulugbek S. Kamilov (S'11) received his M.Sc. degree in Communications Systems from the École Polytechnique Fédérale de Lausanne (EPFL), Switzerland, in 2011.

In 2007–08, he was an exchange student in Electrical and Computer Engineering at Carnegie Mellon University. In 2008, he worked as a research intern at the Telecommunications Research Center in Vienna, Austria. In 2009, he worked as a software engineering intern at Microsoft. In 2010–11, he was a visiting student at the Massachusetts Institute of Technology. In 2011, he joined the Biomedical Imaging Group at EPFL where he is currently working toward his Ph.D. His research interests include message-passing algorithms and the application of signal processing techniques to various biomedical problems.

Sundeep Rangan (M'02) received the B.A.Sc. degree from the University of Waterloo, Canada, and the M.S. and Ph.D. degrees from the University of California, Berkeley, all in electrical engineering. He held postdoctoral appointments at the University of Michigan, Ann Arbor, and Bell Labs. In 2000, he co-founded (with four others) Flarion Technologies, a spin-off of Bell Labs, that developed Flash OFDM, one of the first cellular OFDM data systems. In 2006, Flarion was acquired by Qualcomm Technologies, where Dr. Rangan was a Director of Engineering involved in OFDM infrastructure products. He joined the Department of Electrical and Computer Engineering at the Polytechnic Institute of New York University in 2010, where he is currently an Associate Professor. His research interests are in wireless communications, signal processing, information theory and control theory.

Alyson K. Fletcher (S'03–M'04) received the B.S. degree in mathematics from the University of Iowa. From the University of California, Berkeley, she received the M.S. degree in electrical engineering in 2002, and the M.A. degree in mathematics and Ph.D. degree in electrical engineering, both in 2006.

Dr. Fletcher is a member of SWE, SIAM, and Sigma Xi. In 2005, she received the University of California Eugene L. Lawler Award, the Henry Luce Foundations Clare Boothe Luce Fellowship, the Soroptimist Dissertation Fellowship, and University of California Presidents Postdoctoral Fellowship. Her research interests include signal processing, information theory, machine learning, and neuroscience.

Michael Unser (M'89–SM'94–F'99) received the M.S. (*summa cum laude*) and Ph.D. degrees in Electrical Engineering in 1981 and 1984, respectively, from the Ecole Polytechnique Fdrale de Lausanne (EPFL), Switzerland. From 1985 to 1997, he worked as a scientist with the National Institutes of Health, Bethesda USA. He is now full professor and Director of the Biomedical Imaging Group at the EPFL.

His main research area is biomedical image processing. He has a strong interest in sampling theories, multiresolution algorithms, wavelets, and the use of splines for image processing. He has published 200 journal papers on those topics, and is one of ISIs Highly Cited authors in Engineering (http://isihighlycited.com).

Dr. Unser has held the position of associate Editor-in-Chief (2003-2005) for the IEEE TRANSACTIONS ON MEDICAL IMAGING and has served as Associate Editor for the same journal (1999-2002; 2006-2007), the IEEE TRANSACTIONS ON IMAGE PROCESSING (1992-1995), and the IEEE SIGNAL PROCESSING LETTERS (1994-1998). He is currently member of the editorial boards of Foundations and Trends in Signal Processing, and Sampling Theory in Signal and Image Processing. He co-organized the first IEEE International Symposium on Biomedical Imaging (ISBI2002) and was the founding chair of the technical committee of the IEEE-SP Society on Bio Imaging and Signal Processing (BISP).

Dr. Unser received the 1995 and 2003 Best Paper Awards, the 2000 Magazine Award, and two IEEE Technical Achievement Awards (2008 SPS and 2010 EMBS). He is an EURASIP Fellow and a member of the Swiss Academy of Engineering Sciences.