Generalized Approximate Message Passing for Estimation with Random Linear Mixing

Sundeep Rangan

Abstract—We consider the estimation of a random vector observed through a linear transform followed by a componentwise probabilistic measurement channel. Although such linear mixing estimation problems are generally highly non-convex, Gaussian approximations of belief propagation (BP) have proven to be computationally attractive and highly effective in a range of applications. Recently, Bayati and Montanari have provided a rigorous and extremely general analysis of a large class of approximate message passing (AMP) algorithms that includes many Gaussian approximate BP methods. This paper extends their analysis to a larger class of algorithms to include what we call generalized AMP (G-AMP). G-AMP incorporates general (possibly non-AWGN) measurement channels. Similar to the AWGN output channel case, we show that the asymptotic behavior of the G-AMP algorithm under large i.i.d. Gaussian transform matrices is described by a simple set of state evolution (SE) equations. The general SE equations recover and extend several earlier results, including SE equations for approximate BP on general output channels by Guo and Wang.

Index Terms—Optimization, random matrices, estimation, belief propagation, compressed sensing.

I. INTRODUCTION

Consider the general estimation problem shown in Fig. 1. An input vector \( q \in Q^n \) has components \( q_j \in Q \) for some set \( Q \) and generates an unknown random vector \( x \in \mathbb{R}^n \) through a componentwise input channel described by a conditional distribution \( p_{X|Q}(x_j|q_j) \). The vector \( x \) is then passed through a linear transform

\[
z = Ax,
\]

where \( A \in \mathbb{R}^{m \times n} \) is a known transform matrix. Finally, each component of \( z \) randomly generates an output component \( y_i \) of a vector \( y \in Y^m \) through a second scalar conditional distribution \( p_{Y|Z}(y_i|z_i) \), where \( Y \) is some output set. The problem is to estimate the transform input \( x \) and output \( z \) from the system input vector \( q \), system output \( y \) and linear transform \( A \). The challenge in this problem is that the matrix \( A \) couples, or “mixes,” the input components \( x_j \) with the output components \( z_j \). As a result, computation of the posterior distribution of any individual component \( x_j \) or \( z_i \) is typically intractable as it involves marginalizing a high-dimensional joint distribution of the vectors \( x \) and \( z \).

However, one common approximate, but highly effective, approach for such linear mixing estimation problems is based on loopy belief propagation (BP) [1]. BP has been successfully applied to linear mixing problems in CDMA multiuser detection [2]–[4], lattice codes [5] and compressed sensing [6]–[8]. Although exact implementation of BP for dense graphs is generally computationally difficult, in the linear mixing estimation problem, BP often admits very good Gaussian approximations for large, dense matrices \( A \) [2], [8]–[12].

Bayati and Montanari [12] have recently provided a highly general and rigorous analysis of a large class of approximate message passing (AMP) algorithms that includes Gaussian approximated BP. Their analysis provides a general state evolution (SE) description of the behavior of AMP for large Gaussian i.i.d. matrices \( A \). Prior to their work, analyses of approximate BP algorithms were based on either heuristic arguments such as [8], [9], [13], [14] or certain large sparse limit assumptions for the matrix \( A \) [2], [10], [11], [15], [16].

The main contribution of this paper is to extend Bayati and Montanari’s analysis further to an even larger class of algorithms, which we call generalized AMP (G-AMP). The AMP algorithm analyzed by Bayati and Montanari assumes a linear operation at the output nodes, which corresponds to an additive white Gaussian noise (AWGN) output channel for the mapping between \( z \) and \( y \) in Fig. 1. The work here, in contrast, will consider arbitrary, possibly nonlinear, operations at the output, and outputs \( y \) that are generated by essentially arbitrary conditional distributions \( p_{Y|Z} \) from the transform output \( z \).

Similar to the analysis of AMP and related algorithms, our results show that the asymptotic behavior of G-AMP algorithms under i.i.d. Gaussian matrices is described by a simple set of state evolution (SE) equations. In the case when the estimator uses distributions that are “matched” to the true input and output channels, we recover the SE equations of Guo and Wang [16] that was proven for large sparse matrices.

A full version of the paper can be found in [17] that includes proofs, examples, detailed technical discussions and more background.
II. GENERALIZED APPROXIMATE MESSAGE PASSING

The G-AMP algorithm is a generalization of the AMP method in [12]. The algorithm is parameterized by two functions, $g_{\text{out}}(\cdot)$ and $g_{\text{in}}(\cdot)$, the selection of which will be discussed below. Given the functions, in each iteration $t = 0, 1, 2, \ldots$ the G-AMP algorithm produces estimates $\hat{x}_j(t)$ and $\hat{z}_i(t)$ of the transform input and output variables $x_j$ and $z_i$ as follows:

1) Initialization: Set $t = 0$ and let $\hat{x}_j(t)$ and $\mu_j^x(t) > 0$ be some initial sequences.

2) Output linear step: For each $i$, compute:

\[
\mu_i^x(t) = \sum_j a_{ij}^2 \mu_j^x(t) \tag{2a}
\]

\[
\hat{p}_i(t) = \sum_j a_{ij} \hat{x}_j(t) - \mu_i^x(t) \hat{s}_i(t-1), \tag{2b}
\]

\[
\hat{z}_i(t) = \sum_j a_{ij} \hat{x}_j(t) \tag{2c}
\]

where initially, we take $\hat{s}_i(-1) = 0$.

3) Output nonlinear step: For each $i$,

\[
\hat{s}_i(t) = g_{\text{out}}(t, \hat{p}_i(t), y_i, \mu_i^x(t)) \tag{3a}
\]

\[
\mu_i^x(t) = -\frac{\partial}{\partial \mu_i^x(t)} g_{\text{out}}(t, \hat{p}_i(t), y_i, \mu_i^x(t)). \tag{3b}
\]

4) Input linear step: For each $j$,

\[
\mu_j^z(t) = \left( \sum_i a_{ij}^2 \mu_i^x(t) \right)^{-1} \tag{4a}
\]

\[
\hat{x}_j(t) = \hat{x}_j(t) + \mu_j^z(t) \sum_i a_{ij} \hat{s}_i(t). \tag{4b}
\]

5) Input nonlinear step: For each $j$,

\[
\hat{x}_j(t+1) = g_{\text{in}}(t, \hat{x}_j(t), q_j, \mu_j^z(t)) \tag{5a}
\]

\[
\mu_j^z(t+1) = \mu_j^z(t) \frac{\partial}{\partial \mu_j^z(t)} g_{\text{in}}(t, \hat{x}_j(t), q_j, \mu_j^z(t)). \tag{5b}
\]

Then increment $t = t + 1$ and return to step 2 until a sufficient number of iterations have been performed.

Although our analysis of the G-AMP algorithm applies to essentially arbitrary functions $g_{\text{in}}(\cdot)$ and $g_{\text{out}}(\cdot)$, and initial conditions, we will be most interested in the case when the functions are selected to approximate belief propagation (BP). The exact form of the functions depends on whether we use BP to approximate maximum a posteriori (MAP) estimation or minimum mean squared error (MMSE) estimation.

a) MAP Estimation: As discussed in [17], for MAP estimation, a heuristic approximation of BP can be implemented with the input function

\[
g_{\text{in}}(\hat{r}, q, \mu^r) := \arg\max_x F_{\text{in}}(x, \hat{r}, q, \mu^r) \tag{6a}
\]

\[
F_{\text{in}}(x, \hat{r}, q, \mu^r) := \log p_{X|Q}(x|q) - \frac{1}{2\mu^r}(\hat{r} - x)^2. \tag{6b}
\]

and output function

\[
g_{\text{out}}(\hat{p}, y, \mu^p) := \frac{1}{\mu^p}(\hat{z}^0 - \hat{p}) \tag{7a}
\]

\[
\hat{z}^0_i = \arg\max_z F_{\text{out}}(z, \hat{p}, y, \mu^p) \tag{7b}
\]

\[
F_{\text{out}}(z, \hat{p}, y, \mu^p) = \log p_{Y|Z}(y|z) - \frac{1}{2\mu^p}(z - \hat{p})^2. \tag{7c}
\]

Note that the input and output functions can be interpreted as a scalar MAP estimators for the variables corrupted by additive white Gaussian noise. Thus, the G-AMP algorithm reduces the vector MAP problem with non-Gaussian outputs to a series of scalar Gaussian MAP problems.

The initial condition for the G-AMP algorithm should be taken as

\[
\hat{x}_j(0) = \arg\max_{x_j} \log p_{X|Q}(x_j, q_j) \tag{8a}
\]

\[
\frac{1}{\mu_j^x(0)} = \frac{\partial^2}{\partial x_j^2} \log p_{X|Q}(\hat{x}_j(0), q_j). \tag{8b}
\]

b) MMSE Estimation: For MMSE estimation, [17] shows that a heuristic approximation of standard BP can be implemented with the input function

\[
g_{\text{in}}(\hat{r}, q, \mu^r) := \mathbb{E}(X|\hat{R} = \hat{r}, Q = q), \tag{9}
\]

where the conditional expectation is with respect to the distribution

\[
p_{X|Q,\hat{R}}(x|q, \hat{r}) = \frac{1}{Z} \exp F_{\text{in}}(x, \hat{r}, q, \mu^r), \tag{10}
\]

where $Z$ is a normalization constant and $F_{\text{in}}(\cdot)$ is given in (6b). Similarly, the output function is given by (7b) with

\[
\hat{z}^0 = \mathbb{E}(Z|\hat{P} = \hat{p}, Y = y), \tag{11}
\]

where the conditional expectation is with respect to the distribution

\[
p_{Z|Y,\hat{P}}(z|y, \hat{p}) = \frac{1}{Z} \exp F_{\text{out}}(z, \hat{p}, y, \mu^p), \tag{12}
\]

and $F_{\text{out}}(\cdot)$ is defined in (7c). The initial condition is taken where $\hat{x}_j(0)$ and $\mu_j^z(0)$ are the mean and variance of the priors on the variables $x_j$.

Similar to the MAP estimation case, the input and output functions for the MMSE case can be interpreted as scalar MMSE problems with Gaussian noise.

c) AWGN Output Channels: Suppose that the output channel is additive white Gaussian noise (AWGN) with

\[
y_i = z_i + w_i, \quad w_i \sim \mathcal{N}(0, \mu^w), \tag{13}
\]

for some noise variance $\mu^w > 0$. It is shown in [17] that the output functions for both the MMSE and MAP estimators are identical and given by

\[
g_{\text{out}}(\hat{p}, y, \mu^p) := \frac{y - \hat{p}}{\mu^w + \mu^p}. \tag{14}
\]

This output function is precisely the output assumed in the original AMP algorithm in [12].
III. State Evolution Asymptotic Analysis

A. Assumptions

We first review some definitions from [12], since the analysis here follows along the same framework. A function $\phi : \mathbb{R}^r \to \mathbb{R}^s$ is said to be pseudo-Lipschitz of order $k > 1$, if there exists an $L > 0$ such for any $x, y \in \mathbb{R}^r$,
\[
\|\phi(x) - \phi(y)\| \leq L(1 + \|x\|^{k-1} + \|y\|^{k-1})\|x - y\|. \tag{20}
\]

Now suppose that for each $n = 1, 2, \ldots, v(n)$ is a block vector with components $v_i(n) \in \mathbb{R}^s$, $i = 1, 2, \ldots, \ell(n)$ for some $\ell(n)$. So, the total dimension of $v(n)$ is $s\ell(n)$. We say that the components of the vectors $v(n)$ empirically converge with bounded moments of order $k$ as $n \to \infty$ to a random vector $V$ on $\mathbb{R}^s$ if: For all pseudo-Lipschitz continuous functions, $\phi$, of order $k$,
\[
\lim_{n \to \infty} \frac{1}{\ell(n)} \sum_{i=1}^{\ell(n)} \phi(v_i(n)) = \mathbb{E}(\phi(V)) < \infty. \tag{18}
\]

When the nature of convergence is clear, we may write (with some abuse of notation)
\[
\lim_{n \to \infty} v_i(n) \overset{d}{=} V. \tag{19}
\]

With these definitions, we consider a sequence of random realizations of the estimation problem in Section I indexed by the input signal dimension $n$. For each $n$, we assume the output dimension $m = m(n)$ is deterministic and scales linearly with the input dimension in that
\[
\lim_{n \to \infty} n/m(n) = \beta, \tag{20}
\]
for some $\beta > 0$ called the measurement ratio. We also assume that the transform matrix $A \in \mathbb{R}^{m \times n}$ has i.i.d. Gaussian components $a_{ij} \sim \mathcal{N}(0, 1/m)$ and $z = Ax$.

We assume that for some order $k > 2$, the components of initial condition $\hat{x}(0)$, $\mu^x(0)$ and input vectors $x$ and $q$ empirically converge with bounded moments of order $2k - 2$ as
\[
\lim_{n \to \infty} (\hat{x}(0), \mu^x(0), x_j, q_j) \overset{d}{=} (\bar{x}_0, \bar{\mu}^x(0), X, Q), \tag{21}
\]
for some random variable triple $(\bar{x}_0, X, Q)$ with distribution $p_{\bar{x}_0, X, Q}(\bar{x}_0, x, q)$ and constant $\bar{\mu}^x(0)$.

To model the dependence of the system output vector $y$ on the transform output $z$, we assume that, for each $n$, there is a deterministic vector $w \in W^m$ for some set $W$, which we can think of as a noise vector. Then, for every $i = 1, \ldots, m$, we assume that
\[
y_i = h(z_{i}, w_i). \tag{22}
\]
where $h$ is some function $h : \mathbb{R} \times W \to Y$ and $Y$ is the output set. Finally, we assume that the components of $w$ empirically converge with bounded moments of order $2k - 2$ to some random variable $W$ with distribution $p_W(w)$.

We also need certain continuity assumptions. Specifically, we assume that the partial derivatives of the functions $g_{in}(t, \bar{r}, q, \mu^x)$ and $g_{out}(t, \bar{p}, h(z, w), \mu^p)$ with respect to $\bar{r}, \bar{p}$ and $z$ exist almost everywhere and are pseudo-Lipschitz continuous of order $k$. This assumption implies that the functions $g_{in}(t, \bar{r}, q, \mu^x)$ and $g_{out}(t, \bar{p}, h(z, w), \mu^p)$ are Lipschitz continuous in $\bar{r}$ and $\bar{p}$ respectively.

B. State Evolution Equations

Similar to [12], the key result here is that the behavior of the G-AMP algorithm is described by a set of state evolution (SE) equations. To describe the equations, it is convenient to introduce two random vectors – one corresponding to the input channel, and the other corresponding to the output channel. At the input channel, given $\alpha^r, \tau^r \in \mathbb{R}$ and distributions $p_{X|Q}(q)$ and $p_{X|Q}(x|q)$, define the random variable triple
\[
\theta^r(\tau^r, \alpha^r) := (X, Q, \bar{R}), \tag{23a}
\]
where $Q \sim p_{X|Q}(q)$, $X \sim p_{X|Q}(x|q)$ and $\bar{R}$ is given
\[
\bar{R} = \alpha^r X + V, \quad V \sim \mathcal{N}(0, \tau^r), \tag{23b}
\]
with $V$ independent of $X$ and $Q$. At the output channel, given a covariance matrix $K^p \in \mathbb{R}^{\ell \times 2}$, $K^p > 0$, and distribution $p_{Y|Z}(y|z)$, define the four-dimensional random vector
\[
\theta^p(K^p) := (Z, \bar{P}, W, Y), \quad Y = h(Z, W), \tag{23c}
\]
such that $W$ and $(Z, \bar{P})$ are independent with distributions $Z \sim \mathcal{N}(0, K^p)$, $W \sim p_W(w)$.

We will write $p_{Y|Z}(y|z)$ for the conditional distribution of $Y$ given $Z$ in this model.

With these definitions, we can now state the state evolution (SE) equations for the G-AMP algorithm:

- **Initialization:** Set $t = 0$, let $\pi^x(0)$ be the initial value in (16) and set
  \[
  K^x(0) = \mathbb{E} \left[ X \bar{X}(0) \right] = \left[ X \bar{X}(0) \right], \tag{24a}
  \]
  where the expectation is over the variables $(X, \bar{X}(0))$ in the limit (16).

- **Output node update:** Compute
  \[
  \pi^x(t) = \beta \pi^p(t), \quad K^p(t) = \beta K^x(t) \tag{24b}
  \]
  for some random variable triple $(\bar{x}_0, X, Q)$ with distribution $p_{\bar{x}_0, X, Q}(\bar{x}_0, x, q)$ and constant $\bar{\mu}^x(0)$.

- **Input node update:** Compute
  \[
  \pi^p(t) = \frac{\partial}{\partial \pi} g_{out}(t, \bar{P}, h(z, w), \pi^p(t)) \tag{24c}
  \]
  where the expectations are over the random variable triples $\theta^p(K^p(t)) = (Z, \bar{P}, W, Y)$ in (20). Also, let
  \[
  \alpha^r(t) = \pi^r(t) \times \mathbb{E} \left[ \frac{\partial}{\partial z} g_{out}(t, \bar{P}, h(z, W), \pi^p(t)) \bigl|_{z = Z} \right]. \tag{24d}
  \]
where the expectation is over the random variable triple \( \theta^r(\tau(t), \alpha^r(t)) = (X, Q, \tilde{R}) \) in (18), and

\[
\tilde{X}(t+1) = g_{\text{in}}(t, Q, \tilde{R}, \bar{\pi}^r(t)).
\]

Increment \( t = t + 1 \) and return to step 2.

C. Main Result

Unfortunately, we cannot directly analyze the G-AMP algorithm as stated in Section II. Instead, we need to consider a slight variant of the algorithm with two small modifications: First, we replace the equations (2a) and (4a) in the G-AMP algorithm with

\[
\mu_i^r(t) = \frac{1}{m} \sum_{j=1}^{m} \mu_j^r(t), \quad \frac{1}{\mu_j^r(t)} = \frac{1}{m} \sum_{i=1}^{m} \mu_i^r(t).
\]

Second, in (3) and (5), we replace the parameters \( \mu_i^r(t) \) and \( \mu_j^r(t) \) with \( \bar{\pi}^r(t) \) and \( \bar{\pi}^w(t) \) from the SE equations. We also replace \( \mu_j^r(t) \) with \( \bar{\pi}^w(t) \) in (4b). As discussed in [17], this modified algorithm is still entirely implementable and should, at least heuristically, closely approximate the true algorithm in the case where \( a_{ij} \sim N(0, 1/m) \) as in our assumptions.

**Theorem 1:** Consider the G-AMP algorithm in Section II under the assumptions in Section III-A and with the above modifications. Then, for any fixed iteration number \( t \):

(a) For all \( i \) and \( j \), almost surely, we have the limits

\[
\lim_{n \to \infty} \mu_j^r(t) = \bar{\pi}^r(t), \quad \lim_{n \to \infty} \mu_i^r(t) = \bar{\pi}^w(t). \tag{25}
\]

(b) The components of the vectors \( x, q, \bar{r} \) and \( \bar{x} \) empirically converge with bounded moments of order \( k \) as

\[
\lim_{n \to \infty} (x_j, q_j, \bar{r}_j(t)) \distr \theta^r(\tau(t), \alpha^r(t)), \tag{26}
\]

where \( \theta^r(\tau(t), \alpha^r(t)) = (X, Q, \tilde{R}) \) is the random variable triple in (18) and \( \tau^r(t) \) is from the SE equations.

(c) The components of the vectors \( z, \bar{p}, w \) and \( y \) empirically converge with bounded moments of order \( k \) as

\[
\lim_{n \to \infty} (z_i, \bar{p}_i(t), w_i, y_i) \distr \theta^w(K^p(t)). \tag{27}
\]

The theorem is proven in [17] where it is shown to be a special case of a more general result on certain vector-valued AMP-like recursions.

A useful interpretation of Theorem 1 is that it provides a scalar equivalent model for the behavior of the G-AMP estimates. The equivalent scalar model is illustrated in Fig. 2. To understand this diagram, observe that part (b) of Theorem 1 shows that the joint empirical distribution of the components \((x_j, q_j, \bar{r}_j(t))\) converges to \( \theta^r(\tau^r(t), \alpha^r(t)) = (X, Q, \tilde{R}) \) in (18). This distribution is identical to \( \bar{r}_j(t) \) being a scaled and noise-corrupted version of \( x_j \). Then, the estimate \( \hat{x}_j(t) \) is a nonlinear scalar function of \( \bar{r}_j(t) \) and \( q_j \). A similar interpretation can be drawn at the output nodes.

### IV. Special Cases

Several previous results can be recovered from the general SE equations in Section III-B as special cases:

(a) **AWGN Output Channels for MMSE and MAP Estimation**

First suppose that the estimator uses an input function \( g_{\text{in}}(\cdot) \) based on the MMSE estimator \( g_{\text{out}}(\cdot) \) for an AWGN channel (13). To account for possible mismatches with the estimator, suppose that the G-AMP algorithm is run with some postulated distribution \( p_{X|Q}(\cdot) \) and postulated noise variance \( \mu^w_{\text{post}} \) that may differ from the true distribution \( p_{X|Q}(\cdot) \) and noise variance \( \mu^w \). Let \( x_{\text{mmse}}(\bar{r}, q, \mu^r) \) be the corresponding scalar MMSE estimator

\[
x_{\text{mmse}}(\bar{r}, q, \mu^r) := E \left[ X | \tilde{R} = \bar{r}, Q = q \right],
\]

where the expectation is over the random vector \( \theta^r(\tau^r, \alpha^r) = (X, Q, \tilde{R}) \) in (18) with \( \tau^r = \mu^r, \alpha^r = 1 \) and the postulated distribution, \( p_{X|Q}(\cdot) \). Let \( x_{\text{mmse}}(\bar{r}, q, \mu^r) \) denote the corresponding postulated variance. Then, is shown in [17] that

\[
\bar{\pi}^r(t+1) = \mu^w + \beta E \left[ p_{\text{mmse}}(\tilde{R}, Q, \bar{\pi}^r(t)) \right] \tag{28a}
\]

\[
\tau^r(t+1) = \mu^w + \beta E \left[ X - x_{\text{mmse}}(\tilde{R}, Q, \bar{\pi}^r(t)) \right]^2. \tag{28b}
\]

where both expectations are over \( \theta^r(\tau^r, \alpha^r) = (X, Q, \tilde{R}) \) in (18) with \( \tau^r = \tau(t), \alpha^r = 1 \) and the true distribution \( p_{X|Q}(\cdot) \).

The fixed point of the updates (28) are precisely the equations given by Guo and Verdú in their replica analysis of MMSE estimation with AWGN outputs and non-Gaussian priors [18], which generalized several earlier analyses in [19]-[21]. Also, when the postulated distribution matches the true distribution, the general equations (28) reduce to the SE equations for Gaussian approximated BP on large sparse matrices derived originally by Boutros and Caire [2] along with other works including [10] and [15]. In this regard, the analysis here can be seen as an extension of that work to handle dense matrices and the case of possibly mismatched distributions.

Similarly, when \( g_{\text{in}}(\cdot) \) is based on the MAP estimator (6a) for some postulated distribution and noise variance, [17] shows that the SE equations reduce to

\[
\bar{\pi}^r(t+1) = \mu^w + \beta E \left[ p_{\text{map}}(\tilde{R}, Q, \bar{\pi}^r(t)) \right] \tag{29a}
\]

\[
\tau^r(t+1) = \mu^w + \beta E \left[ X - x_{\text{map}}(\tilde{R}, Q, \bar{\pi}^r(t)) \right]^2. \tag{29b}
\]
where $\ gamma_{map}^{post}$ and $\ gamma_{map}^{post}$ are analogous scalar MAP estimators. Similar to the MMSE case, the fixed point of these equations precisely agree with the equations for replica analysis of MAP estimation with AWGN output noise given in [22] and related works in [23]. Thus, again, the G-AMP framework provides some rigorous justification of the replica results along with an algorithm to achieve the predictions by the replica method.

b) General Output Channels with Matched Distributions

Next consider the case of general (possibly non-AWGN) output channels and the G-AMP algorithm with the MMSE estimation functions where the postulated distributions for $p_X|Q$ and $p_Y|Z$ match the true distributions. In this case, [17] shows that the general SE equations reduce to

$$\gamma^{(t)} = \gamma^{(t+1)} = E \left[ \text{var} \left( X | Q, R \right) \right] ,$$

where $F(\gamma^{(t)})$ is a certain Fisher information at the output; and the expectation in (30b) is over $\theta^{(\gamma^{(t)}}, \alpha^{(t)})$ in (18) with $\alpha^{(t)} = 1$. The updates are precisely the equations in Guo and Wang in [16] for standard BP with large sparse random matrices. The work [11] derived identical equations for a relaxed version of BP, also in the sparse matrix case. The current work thus shows that the identical SE equations hold for dense matrices $A$. In addition, the results here extend the earlier SE equations by considering the cases where the distributions postulated by the estimator may not be identical to the true distribution.

CONCLUSIONS

We have formulated a general linear mixing estimation problem and proposed a class of algorithms, called G-AMP, that includes many common Gaussian approximations of loopy belief propagation. The G-AMP method is computationally simple; and, through an extension of an analysis by Bayati and Montanari in [12], the algorithm admits precise a state evolution performance characterization in the case of large i.i.d. random mixing matrices. The results generalize earlier analyses [11], [16] that only applied to large sparse random matrices. Given the algorithm’s computational simplicity, generality and analytic tractability, we believe that the method will find applications in a large range of problems. Indeed, applications are already being considered in wireless scheduling, image denoising and turbo equalization [24]–[27].

REFERENCES