Accelerated Generalized Approximate Message Passing

Zilu Zhao, Fangqing Xiao, Jichao Chen, Dirk Slock,
Communication Systems Department, EURECOM, France
{zilu.zhao, fangqing.xiao, jichao.chen, dirk.slock}@eurecom.fr

***Abstract*—Generalized Approximate Message Passing (GAMP) enables Bayesian inference in linear models with non-identically and independently distributed (n.i.i.d.) priors and n.i.i.d. measurements of the linear mixture outputs. It represents an efficient technique for approximate inference, which becomes accurate when both rows and columns of the measurement matrix can be treated as sets of independent vectors and both dimensions become large. The fixed points of GAMP correspond to the extrema of a large system limit of the Bethe Free Energy (LSL-BFE), which represents a meaningful approximation criterion regardless of whether the measurement matrix exhibits the independence properties. However, the convergence of (G)AMP can be problematic for certain measurement matrices. In this paper, we revisit the LSL-BFE and its Lagrangian function. We derive an augmented GAMP algorithm by alternately enforcing the Karush-Kuhn-Tucker (KKT) conditions, called KKT-GAMP (KGAMP). To avoid matrix inversions, we introduce Adaptive (Accelerated) Gradient Descent (A(A)GD) techniques. Analysis shows convergence under relaxed conditions. Simulations indicate accelerated convergence compared to existing low complexity methods and illustrate the importance of adaptation.**

I. Introduction

Low complexity Bayesian inference techniques are required for large dimensional sparse signal recovery. Sparse Bayesian Learning (SBL) is a Bayesian inference algorithm proposed by [1] and [2]. SBL is based on a hierarchical prior on the sparse signal, inducing sparsity in an underdetermined scenario by the fact of having to estimate the prior variance hyperparameters. The Linear Minimum Mean Squared Error (LMMSE) estimation step in SBL at each iteration involves matrix inversion, which makes it computationally complex [3]. The Approximate Message Passing (AMP) algorithm can be interpreted as Belief Propagation under the large system limit (LSL) [4], which represents messages by beliefs. Generalized AMP (GAMP) enables non-Gaussian priors and measurement processes. However, the convergence of (G)AMP can be problematic for some measurement/model matrices $\mathbf{A}$. Existing converging AMP versions before [5]: 1) adding the Alternating Direction Method of Multipliers (ADMM) [6] leading to a higher complexity ADMM-GAMP, 2) exploiting part of the singular value decomposition (SVD) of the measurement matrix in Vector AMP (VAMP) [7], [8] or esp. Unitarily Transformed UT-AMP [9] (but which do not allow to handle n.i.i.d. priors conveniently), 3) introducing damping [10], but with typically difficult to determine damping requirements.

- We propose a version of GAMP that alternatingly enforces the KKT conditions (KGAMP) of the large system limit of the Bethe free Energy (LSL-BFE) [11].
- To avoid solving the least-squares subproblem through matrix inversion, we introduce Adaptive Accelerated Gradient Descent (AAGD).
- We provide a convergence analysis of the algorithm under relaxed conditions.
- Simulations show that the proposed KGAMP converges substantially faster than existing low-complexity methods, including AMBGAMP proposed in [5].

II. System Model

We consider here a generalized linear model

$$p_{\boldsymbol{x}}(\boldsymbol{x})=\prod_{i=1}^{N} p_{x_i}(x_i),\ \mathbf{z}=\mathbf{A}\boldsymbol{x},\ p_{\boldsymbol{y}|\mathbf{z}}(\mathbf{z}) = \prod_{j=1}^{M} p_{y_j|z_j}(z_j), \quad (1)$$

where $\mathbf{A}$ is $M \times N$. We interpret the linear mixing as a conditional probability

$$p(\mathbf{z}|\boldsymbol{x}) = \delta(\mathbf{z} - \mathbf{A}\boldsymbol{x}). \quad (2)$$

In the signal recovery problem, we want to recover $\boldsymbol{x}$ from known $\boldsymbol{y}$ and $\mathbf{A}$. In later discussions, we introduce $\mathbf{.}$ as element-wise multiplication, $./$ as element-wise division, and $\mathbf{S} = \mathbf{A.A}$.

III. Large System Limit Bethe Free Energy (LSL-BFE)

In [5], we arrived at a converging version of GAMP with a twist on ADMM, where, similarly to [10] and [6] , we exploit a Lagrangian that is augmented with quadratic versions of the mean constraint while introducing an extra variable $\boldsymbol{u}$ for the estimated posterior mean of $\boldsymbol{x}$ to facilitate the alternating minimization. However, unlike ADMM for which the weights are almost arbitrary, the approach requires introducing very precise quadratic moment weights, which furthermore get ignored when optimizing over the corresponding variables. In the KKT approach pursued here, we can do away with the quadratic moment terms by introducing $\boldsymbol{u}$ in the variance expressions. Below, $\widehat{\boldsymbol{x}} = \mathbb{E}[\boldsymbol{x}|q_{\boldsymbol{x}}]$ and $\boldsymbol{\tau}_{\boldsymbol{x}} = \text{var}(\boldsymbol{x}|q_{\boldsymbol{x}})$ will denote the vectors of means and variances of approximate posterior $q_{\boldsymbol{x}}$, and similarly for $\mathbf{z}$.

After the LSL simplifications [15], the BFE with marginal pdf consistency constraints leading to belief propagation can

be seen to become equivalent to the following LSL-BFE with moment consistency constraints [5], [11]:

$$\begin{aligned}&\min_{q_{\boldsymbol{x}},q_{\mathbf{z}},\boldsymbol{\tau_p}} D[q_{\boldsymbol{x}}\|p_{\boldsymbol{x}}] + D[q_{\mathbf{z}}\|p_{\boldsymbol{y}|\mathbf{z}}] + H_G(q_{\mathbf{z}},\boldsymbol{\tau_p})\\ &\text{s.t. } \widehat{\mathbf{z}} = \mathbf{A}\,\widehat{\boldsymbol{x}}\,,\ \boldsymbol{\tau_p} = \mathbf{S}\,\boldsymbol{\tau_x},\ \text{where}\\ &H_G(q_{\mathbf{z}},\boldsymbol{\tau_p}) = \tfrac{1}{2}\mathbf{1}^T(\boldsymbol{\tau_z}./\boldsymbol{\tau_p}) + \tfrac{1}{2}\textstyle\sum_k \ln(\tau_{p_k})\end{aligned} \tag{3}$$

Here $\mathbf{1}$ denotes a vector of 1's. We introduce the following mean square value (msv) definitions

$$\begin{aligned}(\mathbf{1}./\boldsymbol{\tau}_{\mathbf{r}})^T \text{msv}(\boldsymbol{x}|q_{\boldsymbol{x}}) &= \int \|\boldsymbol{x}\|^2_{\boldsymbol{\tau}_{\mathbf{r}}} q_{\boldsymbol{x}}(\boldsymbol{x}) d\boldsymbol{x}\\ (\mathbf{1}./\boldsymbol{\tau_p})^T \text{msv}(\mathbf{z}|q_{\mathbf{z}}) &= \int \|\mathbf{z}\|^2_{\boldsymbol{\tau_p}} q_{\mathbf{z}}(\mathbf{z}) d\mathbf{z}\end{aligned} \tag{4}$$

where $\|\boldsymbol{x}\|^2_{\boldsymbol{\tau}} = \boldsymbol{x}^T(\boldsymbol{x}./\boldsymbol{\tau})$ and $\text{msv}(\boldsymbol{x}|q_{\boldsymbol{x}})$ on a vector $\boldsymbol{x}$ operates elementwise. Also, $\text{var}(\boldsymbol{x}|q_{\boldsymbol{x}}) = \text{msv}(\boldsymbol{x} - \mathbb{E}\,\boldsymbol{x}|q_{\boldsymbol{x}})$. The Lagrangian of (3) becomes

$$\begin{aligned}&L = D[q_{\boldsymbol{x}}\|p_{\boldsymbol{x}}] + D[q_{\mathbf{z}}\|p_{\boldsymbol{y}|\mathbf{z}}] + \tfrac{1}{2}(\mathbf{1}./\boldsymbol{\tau_p})^T \text{msv}(\mathbf{z} - \mathbf{A}\,\boldsymbol{u}|q_{\mathbf{z}})\\ &+ \tfrac{1}{2}\textstyle\sum_k \ln(\tau_{p_k}) + \mathbf{s}^T(\widehat{\mathbf{z}} - \mathbf{A}\,\widehat{\boldsymbol{x}}) - \tfrac{1}{2}\boldsymbol{\tau}_{\mathbf{s}}^T(\boldsymbol{\tau_p} - \mathbf{S}\,\text{msv}(\boldsymbol{x} - \boldsymbol{u}|q_{\boldsymbol{x}}).\end{aligned} \tag{5}$$

Here, we have introduced a new optimization variable $\boldsymbol{u}$ to fluidify the alternating optimization, and dual variables $\mathbf{s}$ and $\boldsymbol{\tau}_{\mathbf{s}}$. The proposed algorithm corresponds to alternating enforcing of the Karush–Kuhn–Tucker (KKT) conditions for (5). We define diagonal matrices $\mathbf{D_p} = \text{diag}(\boldsymbol{\tau_p})$, $\mathbf{D}_{\mathbf{r}} = \text{diag}(\boldsymbol{\tau}_{\mathbf{r}})$, and

$$\boldsymbol{\tau}_{\mathbf{r}} = \mathbf{1}./(\mathbf{S}^T\boldsymbol{\tau}_{\mathbf{s}})\,. \tag{6}$$

In (5), the weighting of $\text{msv}(\boldsymbol{x}-\boldsymbol{u}|q_{\boldsymbol{x}})$ can therefore be written as $\frac{1}{2}(\mathbf{1}./\boldsymbol{\tau}_{\mathbf{r}})^T$, similar to the weighting of $\text{msv}(\mathbf{z} - \mathbf{A}\,\boldsymbol{u}|q_{\mathbf{z}})$. In the following discussion, we use $(\mathbf{D_p}, \mathbf{D}_{\mathbf{r}})$ and $(\boldsymbol{\tau_p}, \boldsymbol{\tau}_{\mathbf{r}})$ interchangeably.

Proposition 0.1. *If the KKT conditions of* (5) *are satisfied, then* $\forall \boldsymbol{\tau_p}, \boldsymbol{\tau}_{\mathbf{r}}$

$$\begin{aligned}(\mathbf{1}./\boldsymbol{\tau_p})^T msv(\mathbf{z} - \mathbf{A}\,\boldsymbol{u}|q_{\mathbf{z}}) &= (\mathbf{1}./\boldsymbol{\tau_p})^T var(\mathbf{z}|q_{\mathbf{z}})\\ (\mathbf{1}./\boldsymbol{\tau}_{\mathbf{r}})^T msv(\boldsymbol{x} - \boldsymbol{u}|q_{\boldsymbol{x}}) &= (\mathbf{1}./\boldsymbol{\tau}_{\mathbf{r}})^T var(\boldsymbol{x}|q_{\boldsymbol{x}}).\end{aligned} \tag{7}$$

In the following, we use superscript $^+$ to denotes solutions $q^+_{\boldsymbol{x}}$, $q^+_{\mathbf{z}}$, and $\boldsymbol{u}^+$ satisfying the KKT conditions. The main idea of the proof is to show that the KKT conditions of (5) imply

$$\boldsymbol{u}^+ = \mathbb{E}_{q^+_{\boldsymbol{x}}}[\boldsymbol{x}];\ \mathbf{A}\boldsymbol{u}^+ = \mathbb{E}_{q^+_{\mathbf{z}}}[\mathbf{z}]. \tag{8}$$

Proof. We have in general

$$\begin{aligned}&(\mathbf{1}./\boldsymbol{\tau_p})^T \text{msv}(\mathbf{z} - \mathbf{A}\boldsymbol{u}|q_{\mathbf{z}}) = (\mathbf{1}./\boldsymbol{\tau_p})^T \text{var}(\mathbf{z}|q_{\mathbf{z}})\\ &+ (\widehat{\mathbf{z}} - \mathbf{A}\boldsymbol{u})^T\mathbf{D_p}^{-1}(\widehat{\mathbf{z}} - \mathbf{A}\boldsymbol{u}),\end{aligned} \tag{9}$$

$$\begin{aligned}&(\mathbf{1}./\boldsymbol{\tau}_{\mathbf{r}})^T \text{msv}(\boldsymbol{x} - \boldsymbol{u}|q_{\boldsymbol{x}}) = (\mathbf{1}./\boldsymbol{\tau}_{\mathbf{r}})^T \text{var}(\boldsymbol{x}|q_{\boldsymbol{x}})\\ &+ (\widehat{\boldsymbol{x}} - \boldsymbol{u})^T\mathbf{D}_{\mathbf{r}}^{-1}(\widehat{\boldsymbol{x}} - \boldsymbol{u}).\end{aligned} \tag{10}$$

Due to the mean constraint, we have

$$\widehat{\mathbf{z}}^+ = \mathbf{A}\widehat{\boldsymbol{x}}^+,\ \text{where } \widehat{\mathbf{z}}^+ = \mathbb{E}_{q^+_{\boldsymbol{y}|\mathbf{z}}}[\mathbf{z}];\ \widehat{\boldsymbol{x}}^+ = \mathbb{E}_{q^+_{\boldsymbol{x}}}[\boldsymbol{x}]. \tag{11}$$

As we can see, the quadratic terms in (9) and (10), and hence in (5) become zero and hence are minimized for $\boldsymbol{u}^+ = \widehat{\boldsymbol{x}}^+$. Thus, at the optimal points, we have

$$\begin{aligned}\mathbb{E}_{q^+_{\mathbf{z}}}[\|\mathbf{z} - \mathbf{A}\boldsymbol{u}^+\|^2_{\boldsymbol{\tau_p}}] &= (\mathbf{1}./\boldsymbol{\tau_p})^T \text{var}(\mathbf{z}|q^+_{\mathbf{z}})\\ \mathbb{E}_{q^+_{\boldsymbol{x}}}[\|\boldsymbol{x} - \boldsymbol{u}^+\|^2_{\boldsymbol{\tau}_{\mathbf{r}}}] &= (\mathbf{1}./\boldsymbol{\tau}_{\mathbf{r}})^T \text{var}(\boldsymbol{x}|q^+_{\boldsymbol{x}}).\end{aligned} \tag{12}$$

and hence (7). □

To satisfy the KKT conditions, we not only set the partial derivatives of (5) w.r.t. $q_{\boldsymbol{x}}(\boldsymbol{x})$, $q_{\mathbf{z}}(\mathbf{z})$, $\boldsymbol{\tau_p}$, and $\boldsymbol{u}$ to zero, but also satisfy the equality constraints. The KKT conditions are equivalent to the following system of equations [5]

$$\frac{\partial\{\ \|\widehat{\boldsymbol{x}} - \boldsymbol{u}\|^2_{\boldsymbol{\tau}_{\mathbf{r}}} + \|\widehat{\mathbf{z}} - \mathbf{A}\boldsymbol{u}\|^2_{\boldsymbol{\tau_p}}\}}{\partial \boldsymbol{u}} = \mathbf{0} \tag{13}$$

$$\widehat{\mathbf{z}}(\mathbf{s}) = \mathbf{A}\,\widehat{\boldsymbol{x}}(\mathbf{s}) \tag{14}$$

$$q_{\boldsymbol{x}}(\boldsymbol{x}) \propto p_{\boldsymbol{x}}(\boldsymbol{x})\,\mathcal{N}(\boldsymbol{x}|\mathbf{r}, \mathbf{D}_{\mathbf{r}}) \tag{15}$$

$$\boldsymbol{\tau_p} = \mathbf{S}\,\boldsymbol{\tau_x} \tag{16}$$

$$q_{\mathbf{z}}(\mathbf{z}) \propto p_{\boldsymbol{y}|\mathbf{z}}(\mathbf{z})\,\mathcal{N}(\mathbf{z}|\boldsymbol{p}, \mathbf{D_p}) \tag{17}$$

$$\boldsymbol{\tau}_{\mathbf{s}} = (\mathbf{1} - \boldsymbol{\tau_z}./\boldsymbol{\tau_p})./\boldsymbol{\tau_p} \tag{18}$$

where

$$\begin{aligned}&\widehat{\boldsymbol{x}} = \mathbb{E}_{q_{\boldsymbol{x}}}[\boldsymbol{x}];\ \widehat{\mathbf{z}} = \mathbb{E}_{q_{\mathbf{z}}}[\mathbf{z}];\ \boldsymbol{\tau_x} = \text{var}_{q_{\boldsymbol{x}}}[\boldsymbol{x}];\ \boldsymbol{\tau}_{\widehat{\mathbf{z}}} = \text{var}_{q_{\mathbf{z}}}[\mathbf{z}]\\ &\boldsymbol{p} = \mathbf{A}\,\boldsymbol{u} - \mathbf{D_p}\,\mathbf{s};\ \mathbf{r} = \boldsymbol{u} + \mathbf{D}_{\mathbf{r}}\,\mathbf{A}^T\mathbf{s};\ \boldsymbol{\tau}_{\mathbf{r}} = \mathbf{1}./(\mathbf{S}^T\boldsymbol{\tau}_{\mathbf{s}}).\end{aligned} \tag{19}$$

In (16) and (18), the variances $\boldsymbol{\tau_x}$, $\boldsymbol{\tau_z}$ should in principle be the msv terms from (5), but we can replace them with the $\text{var}(\cdot)$ expressions due to the KKT conditions mentioned in the Proposition. This simplification corresponds to ignoring the dependence of the ADMM quadratic moment weights on the optimization variables $\boldsymbol{\tau_p}$, $\boldsymbol{\tau}_{\mathbf{s}}$ in [5], but at least now we know what this simplification corresponds to. In (14), we explicitly write $\widehat{\mathbf{z}}(\mathbf{s})$, $\widehat{\boldsymbol{x}}(\mathbf{s})$ because this is the constraint we use to update the Lagrange multiplier $\mathbf{s}$. Based on the above discussion, we will propose a low complexity algorithm in the following section.

IV. Proposed KGAMP Algorithm

The overall idea is to enforce the KKT conditions (13)-(18) alternatingly. We update 3 groups: $\boldsymbol{u}$, $(q_{\boldsymbol{x}}, \boldsymbol{\tau_p}, q_{\mathbf{z}}, \mathbf{s})$, and $\boldsymbol{\tau}_{\mathbf{s}}$ in an alternating manner. We first update $\boldsymbol{u}$ based on (13). Then $(q_{\boldsymbol{x}}, \boldsymbol{\tau_p}, q_{\mathbf{z}}, \mathbf{s})$ are updated by satisfying (14)-(17) at the same time. Finally, $\boldsymbol{\tau}_{\mathbf{s}}$ is updated based on (18).

A. Update of $\boldsymbol{u}$ by Adaptive Accelerated Gradient Descent

To avoid having to solve the linear system of equations in (13), consider at iteration t the quadratic cost function representing the terms in the Lagrangian (5) depending on $\boldsymbol{u}$:

$$F^t(\boldsymbol{u}) = \frac{1}{2}\|\widehat{\mathbf{z}}^{t-1} - \mathbf{A}\boldsymbol{u}\|^2_{\boldsymbol{\tau_p}^{t-1}} + \frac{1}{2}\|\widehat{\boldsymbol{x}}^{t-1} - \boldsymbol{u}\|^2_{\boldsymbol{\tau}_{\mathbf{r}}^{t-1}}. \tag{20}$$

We introduce a momentum auxiliary variable $\boldsymbol{w}$. The general form for (Nesterov) Accelerated Gradient Descent (AGD) is [16]

$$\begin{aligned}\boldsymbol{u}^t &= \boldsymbol{w}^{t-1} - \alpha\nabla F^t(\boldsymbol{w}^{t-1})\\ \boldsymbol{w}^t &= \boldsymbol{u}^t + \beta(\boldsymbol{u}^t - \boldsymbol{u}^{t-1}),\end{aligned} \tag{21}$$

where the stepsize α and momentum gain β will be made adaptive (AAGD) by alternating line search. Define

$$\begin{aligned}\mathcal{H}^t &= \mathbf{D}(\mathbf{1}./\boldsymbol{\tau}_{\mathbf{r}}^{t-1}) + \mathbf{A}^T\mathbf{D}(\mathbf{1}./\boldsymbol{\tau}_{\boldsymbol{p}}^{t-1})\mathbf{A}\\ \mathbf{g}^t(\boldsymbol{u}) &= \nabla F^t(\boldsymbol{u}) = \mathbf{g}^t(\mathbf{0}) + \mathcal{H}^t\,\boldsymbol{u}\\ &= -\mathbf{A}^T((\widehat{\mathbf{z}}^{t-1} - \mathbf{A}\boldsymbol{u})./\boldsymbol{\tau}_{\boldsymbol{p}}^{t-1}) - (\widehat{\boldsymbol{x}}^{t-1} - \boldsymbol{u})./\boldsymbol{\tau}_{\mathbf{r}}^{t-1}\end{aligned} \tag{22}$$

The quadratic cost function in (20) can be rewritten as

$$\begin{aligned}F^t(\boldsymbol{u}) &= \tfrac{1}{2}\boldsymbol{u}^T\mathcal{H}^t\boldsymbol{u} + \boldsymbol{u}^T\mathbf{g}^t(\mathbf{0}) + const.\\ \mathbf{g}^t(\boldsymbol{w}^{t-1}) &= \nabla F^t(\boldsymbol{w}^{t-1}) = \mathcal{H}^t\boldsymbol{w}^{t-1} + \mathbf{g}^t(\mathbf{0}).\end{aligned} \tag{23}$$

Due to the linearity of ∇F, (21) can be reformulated into a second-order recurrence for $\boldsymbol{u}$:

$$\begin{aligned}\boldsymbol{u}^t =& \boldsymbol{u}^{t-1} - \alpha\,(\mathcal{H}^t\boldsymbol{u}^{t-1} + \mathbf{g}^t(\mathbf{0})) + \beta\,(\boldsymbol{u}^{t-1} - \boldsymbol{u}^{t-2})\\ &+ \alpha\beta\,\mathcal{H}^t(\boldsymbol{u}^{t-1} - \boldsymbol{u}^{t-2})\,.\end{aligned} \tag{24}$$

α and β should be adapted by minimizing $F^t(\boldsymbol{u}^t)$ after substituting (24) into $F^t(\boldsymbol{u})$ in (23). However, to avoid high-order equations, stepsize α and momentum gain β are optimized by alternating line search with a small inner loop. We have actually tried introducing an extra variable $\gamma = \alpha\beta$ and minimizing the resulting quadratic criterion in α, β, γ, but this does not appear to work as well.

1) Adaptive Accelerated Gradient Descent Inner Loop: We use t' to denote the inner loop iterations. Based on (24), define

$$\begin{aligned}\mathbf{d}_1^t(\beta) &= \boldsymbol{u}^{t-1} + \beta\,(\boldsymbol{u}^{t-1} - \boldsymbol{u}^{t-2})\\ \mathbf{d}_2^t(\beta) &= (\mathcal{H}^t\boldsymbol{u}^{t-1} + \mathbf{g}^t(\mathbf{0})) + \beta\,\mathcal{H}^t(\boldsymbol{u}^{t-1} - \boldsymbol{u}^{t-2})\\ \mathbf{d}_3^t(\alpha) &= \boldsymbol{u}^{t-1} - \alpha\,(\mathcal{H}^t\boldsymbol{u}^{t-1} + \mathbf{g}^t(\mathbf{0}))\\ \mathbf{d}_4^t(\alpha) &= (\boldsymbol{u}^{t-1} - \boldsymbol{u}^{t-2}) - \alpha\,\mathcal{H}^t(\boldsymbol{u}^{t-1} - \boldsymbol{u}^{t-2}).\end{aligned} \tag{25}$$

Therefore, (24) becomes

$$\boldsymbol{u}^t = \mathbf{d}_1^t(\beta) - \alpha\mathbf{d}_2^t(\beta) = \mathbf{d}_3^t(\alpha) + \beta\mathbf{d}_4^t(\alpha)\,. \tag{26}$$

By setting the derivative of $F^t(\boldsymbol{u}^t|\alpha^{t'}, \beta^{t'-1})$ with respect to $\alpha^{t'}$ to zero, we obtain

$$\alpha^{t'} = \frac{\mathbf{d}_2^t(\beta^{t'-1})^T(\mathcal{H}^t\mathbf{d}_1^t(\beta^{t'-1}) + \mathbf{g}^t(\mathbf{0}))}{\mathbf{d}_2^t(\beta^{t'-1})^T\mathcal{H}^t\mathbf{d}_2^t(\beta^{t'-1})}. \tag{27}$$

Next, we set the derivative of $F^t(\boldsymbol{u}^t|\alpha^{t'}, \beta^{t'})$ w.r.t. $\beta^{t'}$ to zero, and we obtain

$$\beta^{t'} = -\frac{\mathbf{d}_4^t(\alpha^{t'})^T(\mathcal{H}^t\mathbf{d}_3^t(\alpha^{t'}) + \mathbf{g}^t(\mathbf{0}))}{\mathbf{d}_4^t(\alpha^{t'})^T\mathcal{H}^t\mathbf{d}_4^t(\alpha^{t'})}. \tag{28}$$

We propose to include an inner loop to find α and β iteratively based on (27) and (28). During the first sweep of the inner loop for AAGD, the complexity is $O(MN)$ due to the matrix-vector multiplication. However, after that, each consecutive sweep of the inner loop within one outer iteration has a complexity of $O(1)$.

B. Update of $q_{\boldsymbol{x}}, \boldsymbol{\tau}_{\boldsymbol{p}}, q_{\mathbf{z}}, \mathbf{s}$

By treating $\boldsymbol{u} = \boldsymbol{u}^t$ as given in (19), we have

$$\boldsymbol{\tau}_{\boldsymbol{p}}^t = \mathbf{S}\,\mathrm{var}[\boldsymbol{x}|q_{\boldsymbol{x}}^t];\ \boldsymbol{p}^t = \mathbf{A}\boldsymbol{u}^t - \mathbf{D}_{\boldsymbol{p}}^t\mathbf{s}^t;\ \mathbf{r}^t = \boldsymbol{u}^t + \mathbf{D}_{\mathbf{r}}^{t-1}\mathbf{A}^T\mathbf{s}^t \tag{29}$$

Therefore, the updates for beliefs $q_{\boldsymbol{x}}^t(\boldsymbol{x})$ and $q_{\mathbf{z}}^t(\mathbf{z})$ in (15) and (17) are uniquely determined by the Lagrange multiplier $\mathbf{s}^t$. We find $\mathbf{s}^t$ by solving for $\mathbf{s}^t$ from the mean constraint

$$\mathbb{E}_{q_{\mathbf{z}}^t|\mathbf{s}^t,\boldsymbol{u}^t}[\mathbf{z}] = \mathbf{A}\,\mathbb{E}_{q_{\boldsymbol{x}}^t|\mathbf{s}^t,\boldsymbol{u}^t}[\boldsymbol{x}]. \tag{30}$$

However, solving (30) directly may lead to high complexity. We can also update $\mathbf{s}^t$ by line-search-aided gradient descent on a quadratic version of (30).

After that, we obtain the belief mean and variance by substituting $\boldsymbol{u}^t$ and $\mathbf{s}^t$ into $q_{\boldsymbol{x}}$ and $q_{\mathbf{z}}$.

$$\begin{aligned}q_{\boldsymbol{x}}^t(\boldsymbol{x}) &= p_{\boldsymbol{x}}(\boldsymbol{x})\mathcal{N}(\boldsymbol{x}|\mathbf{r}^t, \mathbf{D}_{\mathbf{r}}^{t-1})\\ q_{\mathbf{z}}^t(\mathbf{z}) &= p_{\boldsymbol{y}|\mathbf{z}}(\mathbf{z})\mathcal{N}(\mathbf{z}|\boldsymbol{p}^t, \mathbf{D}_{\boldsymbol{p}}^t)\end{aligned} \tag{31}$$

Therefore, we have

$$\begin{aligned}\widehat{\boldsymbol{x}}^t &= \mathbb{E}_{q_{\boldsymbol{x}}^t}[\boldsymbol{x}];\ \boldsymbol{\tau}_{\boldsymbol{x}}^t = \mathrm{var}_{q_{\boldsymbol{x}}^t}[\boldsymbol{x}]\\ \widehat{\mathbf{z}}^t &= \mathbb{E}_{q_{\mathbf{z}}^t}[\mathbf{z}];\ \boldsymbol{\tau}_{\mathbf{z}}^t = \mathrm{var}_{q_{\mathbf{z}}^t}[\mathbf{z}]\end{aligned} \tag{32}$$

1) Updating $\mathbf{s}$ in the Gaussian Case: Define diagonal matrices $\boldsymbol{\Sigma}_{\boldsymbol{x}}$ and $\boldsymbol{\Sigma}_{\boldsymbol{v}}$ of prior variances. With Gaussian $p_{\boldsymbol{x}}$ and $p_{\boldsymbol{y}|\mathbf{z}}$

$$p_{\boldsymbol{x}}(\boldsymbol{x}) = \mathcal{N}(\boldsymbol{x}|\mathbf{m}_{\boldsymbol{x}}, \boldsymbol{\Sigma}_{\boldsymbol{x}});\ p_{\boldsymbol{y}|\mathbf{z}}(\mathbf{z}) = \mathcal{N}(\mathbf{z}|\boldsymbol{y}, \boldsymbol{\Sigma}_{\boldsymbol{v}}). \tag{33}$$

In the Gaussian case, the update of $\boldsymbol{\tau}_p$ by (16) is independent of the value of $\mathbf{s}$. Therefore, we have

$$\begin{aligned}\boldsymbol{\tau}_{\boldsymbol{x}}^t &= \mathrm{diag}[(\boldsymbol{\Sigma}_{\boldsymbol{x}}^{-1} + (\mathbf{D}_{\mathbf{r}}^{t-1})^{-1})^{-1}]\\ \boldsymbol{\tau}_{\boldsymbol{p}}^t &= \mathbf{S}\boldsymbol{\tau}_{\boldsymbol{x}}^t;\quad \mathbf{D}_{\boldsymbol{p}}^t = \mathrm{Diag}(\boldsymbol{\tau}_{\boldsymbol{p}}^t)\\ \boldsymbol{\tau}_{\mathbf{z}}^t &= \mathrm{diag}[(\boldsymbol{\Sigma}_{\boldsymbol{v}}^{-1} + (\mathbf{D}_{\boldsymbol{p}}^t)^{-1})^{-1}]\end{aligned} \tag{34}$$

Equation (30) is equivalent to

$$\boldsymbol{Q}\,\mathbf{s} = \mathbf{b}, \tag{35}$$

where

$$\begin{aligned}(\boldsymbol{\Sigma}_p^t)^{-1} &= (\boldsymbol{\Sigma}_{\boldsymbol{v}} + \mathbf{D}_{\boldsymbol{p}}^t)^{-1};\ \mathbf{D}_{\mathbf{z}}^t = \boldsymbol{\Sigma}_{\boldsymbol{v}}(\boldsymbol{\Sigma}_{\boldsymbol{v}} + \mathbf{D}_{\boldsymbol{p}}^t)^{-1}\mathbf{D}_{\boldsymbol{p}}^t\\ (\boldsymbol{\Sigma}_r^t)^{-1} &= (\boldsymbol{\Sigma}_{\boldsymbol{x}} + \mathbf{D}_{\mathbf{r}}^{t-1})^{-1};\ \mathbf{D}_{\boldsymbol{x}}^t = \mathbf{D}_{\mathbf{r}}^{t-1}(\boldsymbol{\Sigma}_{\boldsymbol{x}} + \mathbf{D}_{\mathbf{r}}^{t-1})^{-1}\boldsymbol{\Sigma}_{\boldsymbol{x}}\\ \mathbf{b}^t &= \mathbf{D}_{\boldsymbol{p}}^t(\boldsymbol{\Sigma}_p^t)^{-1}\boldsymbol{y} + \boldsymbol{\Sigma}_{\boldsymbol{v}}(\boldsymbol{\Sigma}_p^t)^{-1}\mathbf{A}\boldsymbol{u}^t\\ &\quad - \mathbf{A}\mathbf{D}_{\mathbf{r}}^{t-1}(\boldsymbol{\Sigma}_r^t)^{-1}\mathbf{m}_{\boldsymbol{x}} - \mathbf{A}\boldsymbol{\Sigma}_{\boldsymbol{x}}(\boldsymbol{\Sigma}_r^t)^{-1}\boldsymbol{u}^t\\ \boldsymbol{Q}^t &= \mathbf{A}\mathbf{D}_{\boldsymbol{x}}^t\mathbf{A}^T + \mathbf{D}_{\mathbf{z}}^t.\end{aligned} \tag{36}$$

To solve (35) in a low complexity manner, we adopt line search (AGD) to the quadratic problem for which (35) can be considered the KKT condition

$$\arg\min_{\mathbf{s}} \frac{1}{2}\mathbf{s}^T\boldsymbol{Q}^t\mathbf{s} - \mathbf{s}^T\mathbf{b}^t. \tag{37}$$

Thus, by line search, $\mathbf{s}^t$ is updated by

$$\mathbf{s}^t = \mathbf{s}^{t-1} - \frac{(\mathbf{h}^t)^T\mathbf{h}^t}{(\mathbf{h}^t)^T\boldsymbol{Q}^t\mathbf{h}^t}\,\mathbf{h}^t \tag{38}$$

where $\mathbf{h}^t = \boldsymbol{Q}^t\mathbf{s}^{t-1} - \mathbf{b}^t$.

2) Updating $\mathbf{s}$ *in the General Case:* In this case we can construct equivalent Gaussian priors by the Expectation Propagation rule, by dividing the Gaussian posterior models constructed from the MMSE estimate and associated MSE, by the Gaussian extrinsics:

$$\tilde{p}_{\boldsymbol{x}}(\boldsymbol{x}) = \mathcal{N}(\boldsymbol{x}|\mathbf{m}_{\boldsymbol{x}}, \boldsymbol{\Sigma}_{\boldsymbol{x}}) \propto \frac{\mathcal{N}(\boldsymbol{x}|\widehat{\boldsymbol{x}}, \mathbf{D}_{\boldsymbol{x}})}{\mathcal{N}(\boldsymbol{x}|\mathbf{r}, \mathbf{D}_{\mathbf{r}})} \tag{39}$$

$$\tilde{p}_{\mathbf{z}}(\mathbf{z}) = \mathcal{N}(\mathbf{z}|\mathbf{m}_{\mathbf{z}}, \boldsymbol{\Sigma}_{\mathbf{z}}) \propto \frac{\mathcal{N}(\mathbf{z}|\widehat{\mathbf{z}}, \mathbf{D}_{\mathbf{z}})}{\mathcal{N}(\mathbf{z}|\boldsymbol{p}, \mathbf{D}_{\boldsymbol{p}})} \tag{40}$$

C. Update of $\boldsymbol{\tau}_{\mathbf{s}}$

Finally, we update $\boldsymbol{\tau}_{\mathbf{s}}$ from $\partial L/\partial \boldsymbol{\tau}_{\boldsymbol{p}} = 0$, leading to (18):

$$\boldsymbol{\tau}_{\mathbf{s}}^t = (\mathbf{1} - \boldsymbol{\tau}_{\mathbf{z}}^t ./ \boldsymbol{\tau}_{\boldsymbol{p}}^t) ./ \boldsymbol{\tau}_{\boldsymbol{p}}^t. \tag{41}$$

By definition in (19), $\boldsymbol{\tau}_{\mathbf{r}}^t = \mathbf{1}./(\mathbf{S}^T \boldsymbol{\tau}_{\mathbf{s}}^t)$, which actually follows from asymptotic Belief Propagation [15].

V. Convergence Analysis

We consider a relaxed algorithm in which,

- Both $p_{\boldsymbol{x}}$ and $p_{\boldsymbol{y}|\mathbf{z}}$ are Gaussian, as in (33),
- $\boldsymbol{u}^t$ is obtained by zeroing the gradient of (20):

$$\boldsymbol{u}^t = (\mathcal{H}^t)^{-1} \mathbf{g}(\mathbf{0}), \tag{42}$$

- $\mathbf{s}^t$ is obtained by solving (30), and hence (35).

It has been shown in [5] that the variance subsystem (which is independent of the mean subsystem in the Gaussian prior case) is guaranteed to converge. In [19], it is shown that under the large system assumption, $\boldsymbol{\tau}_{\boldsymbol{x}}^{\infty}$ converges to the LMMSE posterior variances:

$$\boldsymbol{\tau}_{\boldsymbol{x}}^{\infty} = \text{diag}[(\mathbf{A}^T \boldsymbol{\Sigma}_{\boldsymbol{v}}^{-1} \mathbf{A} + \boldsymbol{\Sigma}_{\boldsymbol{x}}^{-1})^{-1}]. \tag{43}$$

In the following, we will show that the mean subsystem of the relaxed version of KGAMP also converges.

Although this is not essential for the convergence analysis, as the variance susbsystem converges fairly fast, let's assume that it has converged already: $\forall t$, $\boldsymbol{\tau}_{\boldsymbol{p}}^t = \boldsymbol{\tau}_{\boldsymbol{p}}$, and $\boldsymbol{\tau}_{\mathbf{r}}^t = \boldsymbol{\tau}_{\mathbf{r}}$, and hence also $\boldsymbol{Q}^t := \boldsymbol{Q}$. At the end of each iteration, due to the second and third relaxation assumptions, and (30), we have

$$\widehat{\mathbf{z}}^t = \mathbf{A}\, \widehat{\boldsymbol{x}}^t;\ \boldsymbol{u}^t = \widehat{\boldsymbol{x}}^{t-1}. \tag{44}$$

Furthermore, with the second relaxation assumption, we have

$$\mathbf{s}^t = \boldsymbol{Q}^{-1} \mathbf{b}^t, \tag{45}$$

On the other hand, with fixed $\boldsymbol{\tau}_{\boldsymbol{p}}$, $\boldsymbol{\tau}_{\mathbf{r}}$, and the Gaussian prior, we have

$$\widehat{\boldsymbol{x}}^t = \mathbf{D}_{\mathbf{r}}(\boldsymbol{\Sigma}_{\boldsymbol{x}} + \mathbf{D}_{\mathbf{r}})^{-1} \mathbf{m}_{\boldsymbol{x}} + \boldsymbol{\Sigma}_{\boldsymbol{x}}(\boldsymbol{\Sigma}_{\boldsymbol{x}} + \mathbf{D}_{\mathbf{r}})^{-1} \mathbf{r}^t, \tag{46}$$

where $\mathbf{r}^t$ is computed in (29). Substituting (44) and (45) into (29) and substituting the resulting (29) into (46), we have

$$\widehat{\boldsymbol{x}}^t = \boldsymbol{\Phi} \widehat{\boldsymbol{x}}^{t-1} + const, \tag{47}$$

where

$$\begin{aligned} \boldsymbol{\Phi} &= (\mathbf{A}^T \mathbf{D}_{\mathbf{z}}^{-1} \mathbf{A} + \mathbf{D}_{\boldsymbol{x}}^{-1})^{-1} (\mathbf{A}^T \mathbf{D}_{\boldsymbol{p}}^{-1} \mathbf{A} + \mathbf{D}_{\mathbf{z}}^{-1}), \\ \mathbf{D}_{\boldsymbol{x}} &= (\mathbf{D}_{\mathbf{r}}^{-1} + \boldsymbol{\Sigma}_{\boldsymbol{x}}^{-1})^{-1}; \\ \mathbf{D}_{\mathbf{z}} &= (\mathbf{D}_{\boldsymbol{p}}^{-1} + \boldsymbol{\Sigma}_{\boldsymbol{v}}^{-1})^{-1}. \end{aligned} \tag{48}$$

We use $\boldsymbol{\Theta}_1 \preccurlyeq \boldsymbol{\Theta}_2$ to denote that $\boldsymbol{\Theta}_2 - \boldsymbol{\Theta}_1$ is positive semi-definite. From (48), we can verify that $\mathbf{D}_{\mathbf{z}} \preccurlyeq \mathbf{D}_{\boldsymbol{p}}$ and $\mathbf{D}_{\boldsymbol{x}} \preccurlyeq \mathbf{D}_{\mathbf{r}}$. Therefore, we have $\boldsymbol{\Phi} \preccurlyeq \mathbf{I}$. Hence, the update from $\widehat{\boldsymbol{x}}^{t-1}$ to $\widehat{\boldsymbol{x}}^t$ is a contraction and converges.

VI. Simulations

A. Gaussian Case

We consider an $M \times N = 5 \times 10$ measurement model. The input signal follows an exponential decay $x_n \sim \mathcal{N}(0, \sigma_{x_n}^2)$, where $\sigma_{x_n}^2 = 0.01^{(n-1)}$. The elements of $\mathbf{A}$ are drawn i.i.d. $\mathcal{N}(0, \frac{1}{N})$. We consider white Gaussian noise with power $\boldsymbol{\sigma}_v^2 = 10^{-4}$. We randomly generate 1000 realizations of $\boldsymbol{x}$, $\boldsymbol{v}$ and $\mathbf{A}$. For AAGD, we adopt an inner loop of 50 iterations. Since the novelty of KGAMP (w.r,t, AMBGAMP) is in the mean subsystem and we limit the simulations to the Gaussian case, we first obtain $\boldsymbol{\tau}_{\mathbf{r}}$ and $\boldsymbol{\tau}_{\boldsymbol{p}}$ by running the variance updates for 500 iterations. After that, we keep them fixed during the mean subsystem iterations. Furthermore, the same set of $\boldsymbol{\tau}_{\mathbf{r}}$ and $\boldsymbol{\tau}_{\boldsymbol{p}}$ is used in different methods to be compared.

In addition to the benchmark methods ADMM [20] and AMBGAMP [5], we also compare different low complexity methods for updating $\boldsymbol{u}$ in (31).

In Fig. 1, we plot the normalized mean squared error (NMSE) with respect to the ground-truth $\boldsymbol{x}$. Whereas in Fig. 2, we plot the NMSE to the MMSE estimate

$$\widehat{\boldsymbol{x}}_{MMSE} = (\mathbf{A}^T \boldsymbol{\Sigma}_{\boldsymbol{v}}^{-1} \mathbf{A} + \boldsymbol{\Sigma}_{\boldsymbol{x}}^{-1})^{-1} \mathbf{A}^T \boldsymbol{\Sigma}_{\boldsymbol{v}}^{-1} \boldsymbol{y}. \tag{49}$$

In all "KGAMP-" based algorithms, we only update $\boldsymbol{u}$ once per iteration. The proposed method is "KGAMP" in the figures, short for "KGAMP-AAGD". In "KGAMP-Nesterov" and "KGAMP-GD", we use a fixed stepsize $1/L$, where $L = \max[eig(\boldsymbol{Q}^t)]$. "KGAMP-Nesterov" uses standard Nesterov accelerated Gradient descent for updating $\boldsymbol{u}$, while "KGAMP-GD" uses the standard gradient descent algorithm for updating $\boldsymbol{u}$. "KGAMP-GD-line-search" (which is 'KGAMP-AGD" in fact) uses gradient descent with line search stepsize for updating $\boldsymbol{u}$ [5]. The dashed curves are methods that require matrix inversion. "KGAMP Least Squares" uses (45) to update $\mathbf{s}$ and hence $\boldsymbol{u}^t = \widehat{\boldsymbol{x}}^{t-1}$ to update $\boldsymbol{u}$.

"KGAMP Least Squares" provides an indication of how fast the algorithm can converge, as only the alternating optimization component remains to be further accelerated. ADMM-GAMP converges at a comparable rate; however, more realistic gradient-based implementations exhibit slower convergence than the other algorithms. The simulations also demonstrate that optimizing the stepsizes in Nesterov's method leads to substantial additional acceleration.

B. Gaussian Mixture Model Prior

We consider a $\mathbf{A} \in \mathbb{R}^{5\times 10}$ system. In this case the input elements are independent and follow a decaying Gaussian mixture model (GMM) distribution:

$$\begin{aligned} x_n = w_{n,1}\, &\mathcal{N}(0.5^n m_{n,1}, [0.5\, \sigma_{n,1}]^{2n}) \\ &+ w_{n,2}\, \mathcal{N}(0.5^n m_{n,2}, [0.5\, \sigma_{n,2}]^{2n}), \end{aligned} \tag{50}$$

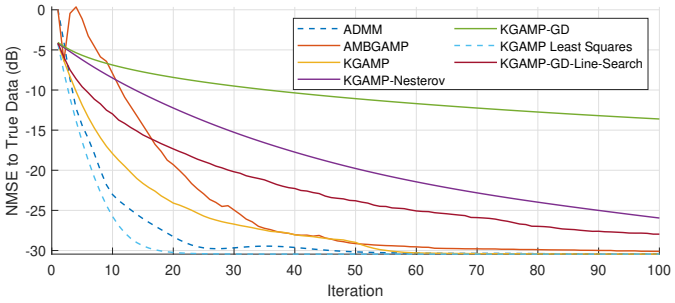

Fig. 1. NMSE-to-$\boldsymbol{x}$ vs Iterations

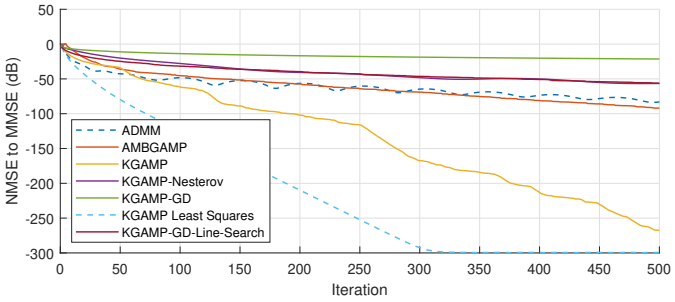

Fig. 2. NMSE-to-$\widehat{\boldsymbol{x}}_{MMSE}$ vs Iterations

where $\forall n, i$, the $w_{n,i}$ are drawn from a uniform distribution between 0 and 1. The sum $w_{n,1} + w_{n,2}$ gets normalized after drawing the realizations. The Gaussian parameters $m_{n,i}$, $\sigma_{n,i}$ are drawn from a standard Gaussian distribution. All the parameters are known during the signal recovery process.

The measurement matrix $\mathbf{A}$ follows the same model as in the Gaussian case above, and the noise is again white Gaussian but with variance adjuste to attain SNR=$17dB$.

We perform 400 different realizations with different prior distribution, measurement matrix and noise. To evaluate the accuracy, we compare the normalized mean squared error to MMSE estimates (which can be computed analytically for a GMM).

In Fig. 3 we compare the proposed KGAMP algorithm to the ADMM-GAMP algorithm proposed in [6], in which the order of some variance updates (involving $\tau_{\boldsymbol{p}}$) needs to be modified (as in [5]) for the algorithm to become convergent in our simulations, and this with a single loop algorithm (unlike the double loop in [6]). In both algorithms, $\boldsymbol{u}$ gets updated by solving the least-squares problem. The simulation shows that KGAMP converges faster and appears more stable after convergence.

VII. Conclusions

We propose KGAMP based on alternatingly enforcing the KKT conditions of the LSL-BFE. To avoid solving the systems of linear equations through matrix inversion, we introduce Adaptive (Accelerated) Gradient Descent (A(A)GD). With Gaussian or non-Gaussian priors and measurements, the proposed algorithm converges (very fast) when the KKT conditions in each alternating step are satisfied exactly. In low

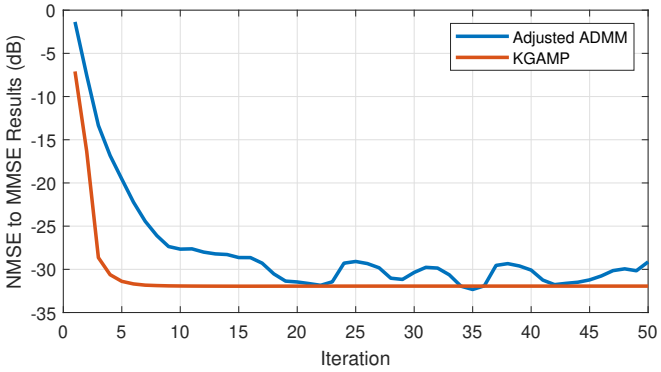

Fig. 3. NMSE-to-$\widehat{\boldsymbol{x}}_{MMSE}$ vs Iterations

complexity GAMP versions, convergence can be slowed down due to (i) GD for updating $\boldsymbol{u}$, (ii) (GD) for updating $\mathbf{s}$, (iii) or a GD-style update for $\mathbf{s}$ in ADMM algorithm versions. It appears that (i) is the dominating slow down factor. Finally, there is also the alternating optimization nature of GAMP which may pose the ultimate convergence speed challenge.

Tproposed more classical approach to

Before pursuing the approach presented in this paper, we actually tried accelerating the convergence of the AMBGAMP algorithm in [5] by pursuing accelerated ADMM algorithms, which have been introduced over the last decade. However, in spite of the Lagrange multiplier update in ADMM corresponding to a gradient ascent step, the stepsize has to be chosen very precisely for ADMM. This makes any deviation in order to introduce acceleration quite delicate. On the other hand, ADMM is used very often in distributed algorithms in which local versions of the variables are updated, among which then equality is imposed via ADMM. The more classical Lagrangian approach proposed in this paper may open new avenues for developing distributed algorithms with accelerated convergence.

Acknowledgements

EURECOM's research is partially supported by its industrial members: ORANGE, BMW, SAP, iABG, Norton LifeLock, by the French PEPR-5G projects PERSEUS and YACARI, the EU H2030 project CONVERGE, and by a Huawei France funded Chair towards Future Wireless Networks.

References

[1] M. E. Tipping, "Sparse Bayesian Learning and the Relevance Vector Machine," *J. Mach. Learn. Res.*, vol. 1, 2001.

[2] D. P. Wipf and B. D. Rao, "Sparse Bayesian Learning for Basis Selection ," *IEEE Trans. on Sig. Proc.*, vol. 52, no. 8, Aug. 2004.

[3] C. K. Thomas and D. Slock, "Low Complexity Static and Dynamic Sparse Bayesian Learning Combining BP, VB and EP Message Passing," in *Asilomar Conf. on Sig., Sys., and Comp.*, CA, USA, 2019.

[4] Z. Zhao and D. Slock, "Extrinsics and Linearized Component-Wise Conditionally Unbiased MMSE Estimation in Approximate Message Passing," in *Int'l Conf. Computing, Networking and Communications (ICNC)*, 2025.

[5] C. K. Thomas, Z. Zhao, and D. Slock, "Towards Convergent Approximate Message Passing by Alternating Constrained Minimization of Bethe Free Energy," in *IEEE Information Theory Workshop (ITW)*, 2023.

[6] S. Rangan, A. Fletcher, P. Schniter, and U. Kamilov, "Inference for Generalized Linear Models via Alternating Directions and Bethe Free Energy Minimization," *IEEE Trans. Info. Theory*, Jan. 2017.

[7] S. Rangan, P. Schniter, and A. K. Fletcher, "Vector Approximate Message Passing," *IEEE Trans. On Info. Theo.*, vol. 65, no. 10, Oct. 2019.

[8] P. Schniter, S. Rangan, and A. K. Fletcher, "Vector approximate Message Passing for the Generalized Linear Model," in *In IEEE 50th Asilomar Conference on Signals, Systems and Computers*, 2016.

[9] Q. Guo and J. Xi, "Approximate Message Passing with Unitary Transformation," 2015. [Online]. Available: https://arxiv.org/abs/1504.04799

[10] S. Rangan, P. Schniter, A. Fletcher, and V. Cevher, "Fixed Points of Generalized Approximate Message Passing with Arbitrary Matrices," *IEEE Trans. Info. Theory*, Dec. 2016.

[11] Z. Zhao and D. Slock, "Bethe Free Energy and Extrinsics in Approximate Message Passing," in *Asilomar Conf. Signals, Systems, and Computers*, 2023.

[12] Z. Zhao, F. Xiao, and D. Slock, "Approximate Message Passing for Not So Large niid Generalized Linear Models," in *IEEE Int'l Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, 2023.

[13] Z. Zhao, F. Xiao, C. K. Thomas, and D. Slock, "Reconciling AMP Algorithms derived from Belief Propagation or the Large System Limit Bethe Free Energy," in *IEEE Int'l Conf. Acoustics, Speech and Signal Processing (ICASSP)*, 2025.

[14] Z. Zhao, F. Xiao, and D. Slock, "Vector Approximate Message Passing for Not So Large N.I.I.D. Generalized I/O Linear Models," in *IEEE Int'l Conf. Acoustics, Speech and Signal Processing (ICASSP)*, 2024.

[15] ——, "Extrinsics and Linearized Component-Wise Conditionally Unbiased MMSE Estimation as in GAMP," in *Asilomar Conf. Signals, Systems, and Computers*, 2024.

[16] ——, " Accelerated Alternating Optimization of MIMO Transceivers," in *Asilomar Conf. Signals, Systems, and Computers*, 2025.

[17] ——, "Expectations in Expectation Propagation," 2026. [Online]. Available: https://arxiv.org/abs/2512.08034

[18] Z. Zhao and D. Slock, "Approximating Univariate Factored Distributions via Message-Passing Algorithms," 2026. [Online]. Available: https://arxiv.org/abs/2602.01377

[19] ——, "Variance Predictions in VAMP/UAMP with Right Rotationally Invariant Measurement Matrices for niid Generalized Linear Models," in *European Signal Processing Conf. (EUSIPCO)*, 2023.

[20] S. Rangan, A. K. Fletcher, P. Schniter, and U. S. Kamilov, "Inference for Generalized Linear Models via Alternating Directions and Bethe Free Energy Minimization," *IEEE Transactions on Information Theory*, vol. 63, no. 1, pp. 676–697, 2017.