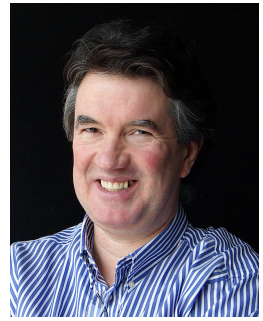
EURECOM
Sophia Antipolis



Alternating Constrained Minimization based Approximate Message Passing

Christo Kurisummoottil Thomas[1], **Dirk Slock**[2]

[1] NEWS Group, Virginia Tech, MD, USA

[2] Communication Systems Department, EURECOM, France

Outline

Outline

Introduction

- recovery of a sparse signal vector $\boldsymbol{x}$ can be formulated as,

$$\boldsymbol{y} = \mathbf{A}\boldsymbol{x} + \boldsymbol{v}$$

$\boldsymbol{y}$ = observations or data,
$\mathbf{A}$ = measurement or sensing matrix which is known and is of dimension $M \times N$ with typically $M < N$.

- In the sparse model case, $\boldsymbol{x}$ contains only K non-zero (or significant) entries, with $K < M < N$.
- In Bayesian inference, the Sparse Bayesian Learning (SBL) algorithm is based on a two or three layer hierarchical prior on the sparse coefficients $\boldsymbol{x}$. The priors for the hyperparameters (precision parameters) are chosen such that the marginal prior for $\boldsymbol{x}$ induces sparsity, allowing the majority of the coefficients to tend towards zero.
- [1] provides a detailed overview of the various sparse signal recovery algorithms which fall under l_1 or l_2 norm minimization approaches such as Basis Pursuit, LASSO etc and SBL methods. The authors justify the superior recovery performance of SBL compared to the above mentioned conventional methods.
- Nevertheless, the matrix inversion involved in the Linear Minimum Mean Squared Error (LMMSE) step in SBL at each iteration makes it computationally complex even for moderately large data sets. This complexity is the motivation behind approximate inference methods.

State of the Art

- Belief propagation (BP) is a message passing technique that allows to compute the posterior marginals. However, due to loops in the factor graph, loopy belief propagation may have convergences issues and is furthermore still relatively complex. BP based SBL algorithms [2] are computationally more efficient.
- The Approximate Message Passing (AMP) algorithm has been introduced to reduce the complexity of Belief Propagation, from $2MN$ to $M+N$ messages. Generalized AMP (GAMP) allows non-Gaussian priors and measurement processes.
- But convergence of (G)AMP can be problematic for some matrices $\mathbf{A}$. Existing converging AMP versions introduced so far:
  1) adding the Alternating Direction Method of Multipliers (ADMM) [3] leading to a higher complexity ADMM-GAMP,
  2) exploiting part of the singular value decomposition (SVD) of the measurement matrix in Vector AMP (VAMP) [4], [5] or esp. Unitarily Transformed UT-AMP [6] (but which do not allow to handle n.i.i.d. priors conveniently),
  3) introducing damping [7], but with typically difficult to determine damping requirements.

Contributions

- We propose a convergent version of GAMP, AMBGAMP, which applies alternating minimization to an augmented Lagrangian of a large system limit of the Bethe free Energy (BFE).
- AMBGAMP can be interpreted as applying a simplified ADMM to the BFE, with a constrained Lagrange multiplier parameterization for the mean constraint, and a quadratic optimization subproblem being solved by a gradient update with line search. The ADMM is complemented with a fixed point iteration for the variance constraint.
- We show that AMBGAMP converges to the LMMSE estimate in the Gaussian case.
- Furthermore, under the SBL setting, the posterior for inverse variance (precisions) hyper-parameter posteriors are obtained by minimizing the Kullback-Leibler divergence (KLD) between the approximate posterior of $\boldsymbol{x}$ (computed by the AMBGAMP-SBL iterations) and the true posterior. The resulting algorithm that involves optimization of the posteriors for $\boldsymbol{x}$ and the auxiliary variables (including the hyperparameters) is called variational AMBGAMP-SBL (VAMBGAMP-SBL).
- The simulation results validate the convergence to LMMSE solutions under different measurement matrices variants, including i.i.d and low-rank cases.

Generalized Linear Model

- The data model considered in GAMP is essentially a linear mixing model

$$\mathbf{z} = \mathbf{A}\,\boldsymbol{x}\ ,\ p_{\boldsymbol{x},\boldsymbol{\alpha}}(\boldsymbol{x},\boldsymbol{\alpha}) = \prod_{i=1}^{N} p_{x_i,\alpha_i}(x_i,\alpha_i)\ ,\ p_{\boldsymbol{y},\boldsymbol{\gamma}|\mathbf{z}}(\boldsymbol{y},\boldsymbol{\gamma}|\mathbf{z}) = \prod_{k=1}^{M} p_{y_k|z_k}(y_k|z_k,\gamma_k)p_{\gamma_k}(\gamma_k)$$

with (possibly) non identically independently distributed (n.i.i.d.) prior $p_{\boldsymbol{x},\boldsymbol{\alpha}}(\boldsymbol{x},\boldsymbol{\alpha})$ and n.i.i.d. measurements $p_{\boldsymbol{y},\boldsymbol{\gamma}|\mathbf{z}}(\boldsymbol{y},\boldsymbol{\gamma}|\mathbf{z})$. The noise precision vector (inverse variance hyper-parameter) is $\boldsymbol{\gamma} = [\gamma_1,\cdots,\gamma_M]^T$. In SBL, we parameterize the prior using the unknown precisions (inverse variances) $\alpha = [\alpha_1,\cdots\alpha_N]^T$. Hence, we write

$$p_{x_i,\alpha_i}(x_i,\alpha_i) = p_{x_i|\alpha_i}(x_i|\alpha_i)p_{\alpha_i}(\alpha_i) = \mathcal{N}(x_i;0,\alpha_i^{-1})\mathcal{G}(\alpha_i;a,b)$$

where the Gamma distribution for α_i, $\mathcal{G}(\alpha_i;a,b) = \frac{\alpha_i^{a-1}e^{-b\alpha_i}b^{\alpha_i}}{\Gamma(\alpha_i)}$. Note: case of an uninformative prior corresponds to $a=1$, $b=0$. Each γ_i is also assumed to have a Gamma prior $p_{\gamma_k}(\gamma_k) = \mathcal{G}(\gamma_k;c,d)$, with known c,d .

- In Bayesian estimation, we are interested in the posterior

$$p_{\boldsymbol{x},\mathbf{z}|\boldsymbol{y}}(\boldsymbol{x},\mathbf{z},\boldsymbol{\alpha},\boldsymbol{\gamma}|\boldsymbol{y}) = \frac{e^{-\sum_{i=1}^{N} f_{x_i,\alpha_i}(x_i,\alpha_i) - \sum_{k=1}^{M} f_{z_k,\gamma_k}(z_k,\gamma_k)}}{Z(\boldsymbol{y})}\ \mathbb{1}_{\{\mathbf{z}=\mathbf{A}\boldsymbol{x}\}}, \tag{1}$$

where we define the negative log-likelihoods as $f_{x_i,\alpha_i}(x_i,\alpha_i) = -\ln p_{x_i,\alpha_i}(x_i,\alpha_i)$, $f_{z_k,\gamma_k}(z_k,\gamma_k) = -\ln p_{y_k,\gamma_k|z_k}(y_k,\gamma_k|z_k)$, (up to constants that may depend on $\boldsymbol{y}$, absorbed in the normalization $Z(\boldsymbol{y})$).

- Bayesian problem: computation of $Z(\boldsymbol{y})$, posterior means and variances.

Outline

GAMP

- GAMP is an approximate belief propagation technique which is motivated by asymptotic considerations in which the rows and columns of the measurement matrix $\mathbf{A}$ are considered as random and independent, in which case GAMP can actually produce the correct posterior marginals. In any case, GAMP computes a separable approximate posterior of the form

$$q_{\boldsymbol{x},\boldsymbol{\alpha},\mathbf{z},\boldsymbol{\gamma}}(\boldsymbol{x},\boldsymbol{\alpha},\mathbf{z},\boldsymbol{\gamma}) = q_{\boldsymbol{x}}(\boldsymbol{x})\, q_{\boldsymbol{\alpha}}(\boldsymbol{\alpha})\, q_{\mathbf{z}}(\mathbf{z}) q_{\boldsymbol{\gamma}}(\boldsymbol{\gamma}) = \prod_{i=1}^{N} q_{x_i}(x_i) q_{\alpha_i}(\alpha_i) \prod_{k=1}^{M} q_{z_k}(z_k) q_{\gamma_k}(\gamma_k), \tag{2}$$

  in which the dependence on $\boldsymbol{y}$ has been omitted.
- The GAMP algorithm [8], [7] appears in the table for Algorithm 1. We emphasize that, apart from the hyperparameter updates, the algorithms specified in the tables hold for general Generalized Linear Models, only the indicated hyperparameter updates are specific for SBL.
- We only consider here Sum-Product GAMP (for MMSE estimation, as opposed to Max-Sum GAMP for MAP estimation).

GAMP vs AMBGAMP

Algorithm 1 EM-GAMP-SBL

Require: $\boldsymbol{y}, \mathbf{A}, \mathbf{S} = \mathbf{A}.\mathbf{A}, f_{x,\boldsymbol{\alpha}}(x, \boldsymbol{\alpha}), f_{z,\boldsymbol{\gamma}}(z, \boldsymbol{\gamma})$

Initialize: $t = 0, \widehat{x}^t, \boldsymbol{\tau}_x^t, \mathbf{s}^{t-1} = \mathbf{0}, \widehat{\boldsymbol{\alpha}}^{t-1}, \widehat{\boldsymbol{\gamma}}^{t-1}$
repeat
  [Output node update]
  $\boldsymbol{\tau}_p^t = \mathbf{S}\,\boldsymbol{\tau}_x^t$
  $\boldsymbol{p}^t = \mathbf{A}\,\widehat{x}^t - \mathbf{s}^{t-1}.\boldsymbol{\tau}_p^t$
  $\widehat{\boldsymbol{z}}^t = \mathbb{E}(\boldsymbol{z}|\boldsymbol{p}^t, \boldsymbol{\tau}_p^t, \widehat{\boldsymbol{\gamma}}^{t-1})$
  $\boldsymbol{\tau}_z^t = \mathrm{var}(\boldsymbol{z}|\boldsymbol{p}^t, \boldsymbol{\tau}_p^t, \widehat{\boldsymbol{\gamma}}^{t-1})$
  $\mathbf{s}^t = (\widehat{z}^t - \boldsymbol{p}^t)./\boldsymbol{\tau}_p^t$
  $\boldsymbol{\tau}_s^t = (1 - \boldsymbol{\tau}_z^t./\boldsymbol{\tau}_p^t)./\boldsymbol{\tau}_p^t$
  [Input node update]
  $\boldsymbol{\tau}_r^t = 1./(\mathbf{S}^T\boldsymbol{\tau}_s^t)$
  $\mathbf{r}^t = \widehat{x}^t + \boldsymbol{\tau}_r^t.\mathbf{A}^T\mathbf{s}^t$
  $\widehat{\boldsymbol{x}}^{t+1} = \mathbb{E}(\boldsymbol{x}|\mathbf{r}^t, \boldsymbol{\tau}_r^t, \widehat{\boldsymbol{\alpha}}^{t-1})$
  $\boldsymbol{\tau}_{\boldsymbol{x}}^{t+1} = \mathrm{var}(\boldsymbol{x}|\mathbf{r}^t, \boldsymbol{\tau}_r^t, \widehat{\boldsymbol{\alpha}}^{t-1})$
  [Hyperparameters update]
  $\widehat{\alpha}_i^t = \frac{2a+1}{\mathbb{E}(x_i^2)+2b},\ \forall i.$
  $\widehat{\gamma}_k^t = \frac{2c+1}{\mathbb{E}(|y_k - z_k|^2)+2d},\ \forall k.$
until Convergence

Algorithm 2 VAMBGAMP(-SBL)

Require: $\boldsymbol{y}, \mathbf{A}, \mathbf{S} = \mathbf{A}.\mathbf{A}, f_{x,\boldsymbol{\alpha}}(x, \boldsymbol{\alpha}), f_{z,\boldsymbol{\gamma}}(z, \boldsymbol{\gamma})$

Initialize: $t = 0, \widehat{x}^0, \boldsymbol{\tau}_x^0, \boldsymbol{u}^0, \boldsymbol{\tau}_p^0, \mathbf{s}^0 = \mathbf{0}, \widehat{\boldsymbol{\gamma}}^0, \widehat{\boldsymbol{\alpha}}^0$
repeat (t=1,2,...)
  [Output node update]
  $\boldsymbol{u}^t = \boldsymbol{u}^{t-1} - \eta^t\,\mathbf{g}^t$, with $\mathbf{g}^t, \eta^t$ from (17), (18)
  $\boldsymbol{p}^t = \mathbf{A}\,\boldsymbol{u}^t - \mathbf{s}^{t-1}.\boldsymbol{\tau}_p^{t-1}$
  $\widehat{\boldsymbol{z}}^t = \mathbb{E}(\boldsymbol{z}|\boldsymbol{p}^t, \boldsymbol{\tau}_p^{t-1}, \widehat{\boldsymbol{\gamma}}^{t-1})$, Gaussian case: $\widehat{z}^t = \boldsymbol{\tau}_z^t.(\boldsymbol{y}.\widehat{\boldsymbol{\gamma}}^{t-1} + \boldsymbol{p}^t./\boldsymbol{\tau}_p^{t-1})$
  $\boldsymbol{\tau}_{\boldsymbol{z}}^t = \mathrm{var}(\boldsymbol{z}|\boldsymbol{p}^t, \boldsymbol{\tau}_p^{t-1}, \widehat{\boldsymbol{\gamma}}^{t-1})$, Gaussian case: $1./\boldsymbol{\tau}_z^t = 1./\boldsymbol{\tau}_p^{t-1} + \widehat{\boldsymbol{\gamma}}^{t-1}$
  $\mathbf{s}^t = \mathbf{s}^{t-1} + (z^t - \mathbf{A}\boldsymbol{u}^t)./\boldsymbol{\tau}_p^{t-1}$
  [Variance matching]
  $\boldsymbol{\tau}_s^t = (1 - \boldsymbol{\tau}_z^t./\boldsymbol{\tau}_p^{t-1})./\boldsymbol{\tau}_p^{t-1}$
  $\boldsymbol{\tau}_p^t = \mathbf{S}\,\boldsymbol{\tau}_x^{t-1}$
  $\boldsymbol{\tau}_r^t = 1./(\mathbf{S}^T\boldsymbol{\tau}_s^t)$
  [Input node update]
  $\mathbf{r}^t = \boldsymbol{u}^t + \boldsymbol{\tau}_r^t.\mathbf{A}^T\mathbf{s}^t$
  $\widehat{\boldsymbol{x}}^t = \mathbb{E}(\boldsymbol{x}|\mathbf{r}^t, \boldsymbol{\tau}_r^t, \widehat{\boldsymbol{\alpha}}^{t-1})$, Gaussian case: $\widehat{x}^t = \mathbf{r}^t./(1 + \widehat{\boldsymbol{\alpha}}^{t-1}.\boldsymbol{\tau}_r^t)$
  $\boldsymbol{\tau}_{\boldsymbol{x}}^t = \mathrm{var}(\boldsymbol{x}|\mathbf{r}^t, \boldsymbol{\tau}_r^t, \widehat{\boldsymbol{\alpha}}^{t-1})$, Gaussian case: $1./\boldsymbol{\tau}_x^t = 1./\boldsymbol{\tau}_r^t + \widehat{\boldsymbol{\alpha}}^{t-1}$
  [Hyperparameters update]
  $\widehat{\alpha}_i^t = \frac{2a+1}{\mathbb{E}_{q_{x_i}}(x_i^2)+2b},\ \forall i$
  $\widehat{\gamma}_k^t = \frac{2c+1}{\mathbb{E}_{q_{z_k}}(|y_k - z_k|^2)+2d},\ \forall k$
until Convergence

Outline

VAMBGAMP

- AMB is short for ACM-LSL-BFE: Alternating Constrained Minimization of the Large System Limit of the Bethe Free Energy.
- AMBGAMP uses most of the same updates as GAMP, but GAMP does not rigorously follow the principle of alternating minimization (block coordinate descent) esp. in the presence of constraints.
- any fixed point GAMP is a critical point of the following constrained minimization of a Large System Limit (LSL) of the Bethe Free Energy (BFE) (see [7] and references therein), here extended to include hyperparameters:

$$\begin{aligned} \min_{q_{\boldsymbol{x}}, q_{\mathbf{z}}, \boldsymbol{\tau}_p, q_{\alpha}, q_{\boldsymbol{\gamma}}} \quad & J_{LSL-BFE}(q_{\boldsymbol{x}}, q_{\mathbf{z}}, \boldsymbol{\tau}_p, q_{\boldsymbol{\alpha}}, q_{\boldsymbol{\gamma}}) \\ s.t. \quad & \mathbb{E}(\mathbf{z}|q_{\mathbf{z}}) = \mathbf{A}\, \mathbb{E}(\boldsymbol{x}|q_{\boldsymbol{x}}) \\ & \boldsymbol{\tau}_p = \mathbf{S}\, \mathsf{var}(\boldsymbol{x}|q_{\boldsymbol{x}}) \end{aligned} \tag{3}$$

where the LSL BFE is given by

$$J_{LBFE}(q_{\boldsymbol{x}}, q_{\mathbf{z}}, \boldsymbol{\tau}_p, q_{\boldsymbol{\alpha}}, q_{\boldsymbol{\gamma}}) = D(q_{\boldsymbol{x}} q_{\boldsymbol{\alpha}} || e^{-f_{\boldsymbol{x},\alpha}}) + D(q_{\mathbf{z}} q_{\boldsymbol{\gamma}} || e^{-f_{\mathbf{z},\gamma}}) + H_G(q_{\mathbf{z}}, \boldsymbol{\tau}_p),$$

with $H_G(q_{\mathbf{z}}, \boldsymbol{\tau}_p) = \frac{1}{2} \sum_{k=1}^{M} \left[\frac{\mathsf{var}(z_k|q_{z_k})}{\tau_{p_k}} + \ln(2\pi\, \tau_{p_k}) \right]$

and where $D(q||p) = \mathbb{E}(\ln(\frac{q}{p})\,|q)$ is the Kullback-Leibler distance (KLD) and $H_G({}_q\mathbf{z}, \boldsymbol{\tau}_p)$ is a sum of a KLD and an entropy of Gaussians with identical means but different variances.

VAMBGAMP Augmented Lagrangian

- The LSL BFE optimization problem (3) can be reformulated with the following augmented Lagrangian

$$\begin{aligned}
&\min_{q_{\boldsymbol{x}},q_{\mathbf{z}},q_{\boldsymbol{\alpha}},q_{\boldsymbol{\gamma}},\boldsymbol{\tau}_p,\boldsymbol{u}} \max_{\mathbf{s},\boldsymbol{\tau}_s} L(q_{\boldsymbol{x}},q_{\mathbf{z}},\boldsymbol{\tau}_p,\boldsymbol{u},\mathbf{s},\boldsymbol{\tau}_s,q_{\boldsymbol{\alpha}},q_{\boldsymbol{\gamma}}) \text{ with} \\
&L = D(q_{\boldsymbol{x}}q_{\boldsymbol{\alpha}}||e^{-f_{\boldsymbol{x},\alpha}})+D(q_{\mathbf{z}}q_{\boldsymbol{\gamma}}||e^{-f_{\mathbf{z},\gamma}})+H_G(q_{\mathbf{z}},\boldsymbol{\tau}_p) \\
&+\mathbf{s}^T(\mathbb{E}(\mathbf{z}|q_{\mathbf{z}}) - \mathbf{A}\,\mathbb{E}(\boldsymbol{x}|q_{\boldsymbol{x}})) - \tfrac{1}{2}\boldsymbol{\tau}_s^T(\boldsymbol{\tau}_p - \mathbf{S}\,\mathrm{var}(\boldsymbol{x}|q_{\boldsymbol{x}})) \\
&+\tfrac{1}{2}\|\,\mathbb{E}(\boldsymbol{x}|q_{\boldsymbol{x}}) - \boldsymbol{u}\|^2_{\boldsymbol{\tau}_r} + \tfrac{1}{2}\|\,\mathbb{E}(\mathbf{z}|q_{\mathbf{z}}) - \mathbf{A}\,\boldsymbol{u}\|^2_{\boldsymbol{\tau}_p},
\end{aligned} \tag{4}$$

  where $\mathbf{s}$, $\boldsymbol{\tau}_s$ are Lagrange multipliers, and $\boldsymbol{\tau}_r = \mathbf{1}./(\mathbf{S}^T\boldsymbol{\tau}_s)$ is just a short-hand notation for quantities that depend on $\boldsymbol{\tau}_s$.

- Last term = ADMM style quadratic version of mean constraint, with auxiliary variable $\boldsymbol{u}$, requiring addition of second to last term. Weights in quadratic terms are judiciously chosen.
- We also use the notations: $\|\boldsymbol{u}\|^2_{\boldsymbol{\tau}} = \sum_i u_i^2/\tau_i$, element-wise multiplication as in $\mathbf{s}.\boldsymbol{\tau}$ and element-wise division as in $\mathbf{1}./\boldsymbol{\tau}$.
- Earlier attempt [9], [10]: We interpret the constraints as follows: $\mathbb{E}(\mathbf{z}|q_{\mathbf{z}}) = \mathbf{A}\,\mathbb{E}(\boldsymbol{x}|q_{\boldsymbol{x}})$ is interpreted as a constraint on $\mathbb{E}(\mathbf{z}|q_{\mathbf{z}})$, and $\boldsymbol{\tau}_p = \mathbf{S}\,\mathrm{var}(\boldsymbol{x}|q_{\boldsymbol{x}})$ (which is a vector of the individual variances) is interpreted as a constraint on $\boldsymbol{\tau}_p$.

From ADMM-GAMP to AMBGAMP

- In [9], [10], a careful updating schedule was considered with partial optimization steps on subsets of primal and dual variables. However, that approach is not guaranteed to converge in general.
- In [11] we continued to consider an alternating optimization approach in which the schedule is less critical and some of the optimizations are reduced to gradient updates. The resulting algorithm can be considered an extended and generalized version of the ADMM algorithm (extended: there are more than two primal variable groups, generalized: the quadratic augmentation term does not exactly correspond to the linear (mean) constraint).
- However, there is an alternative point of view, based on [3], where a double mean constraint was introduced leading to the ADMM-GAMP augmented Lagrangian

$$\min_{q_{\boldsymbol{x}},q_{\mathbf{z}},\boldsymbol{\tau}_p,\boldsymbol{u}} \max_{\mathbf{q},\mathbf{s},\boldsymbol{\tau}_s} L_A(q_{\boldsymbol{x}},q_{\mathbf{z}},\boldsymbol{\tau}_p,\boldsymbol{u},\mathbf{q},\mathbf{s},\boldsymbol{\tau}_s) \text{ with}$$

$$\begin{aligned} L_A =& D(q_{\boldsymbol{x}}||e^{-f_{\boldsymbol{x}}}) - \tfrac{1}{2}\boldsymbol{\tau}_s^T(\boldsymbol{\tau}_p - \mathbf{S}\,\mathrm{var}(\boldsymbol{x}|q_{\boldsymbol{x}})) + D(q_{\mathbf{z}}||e^{-f_{\mathbf{z}}}) \\ &+ H_G(q_{\mathbf{z}},\boldsymbol{\tau}_p) + \mathbf{q}^T(\mathbb{E}(\boldsymbol{x}|q_{\boldsymbol{x}}) - \boldsymbol{u})) + \mathbf{s}^T(\mathbb{E}(\mathbf{z}|q_{\mathbf{z}}) - \mathbf{A}\,\boldsymbol{u})) \\ &+ \tfrac{1}{2}\|\,\mathbb{E}(\boldsymbol{x}|q_{\boldsymbol{x}}) - \boldsymbol{u}\|^2_{\boldsymbol{\tau}_r} + \tfrac{1}{2}\|\,\mathbb{E}(\mathbf{z}|q_{\mathbf{z}}) - \mathbf{A}\,\boldsymbol{u}\|^2_{\boldsymbol{\tau}_p}, \end{aligned}$$

ADMM-GAMP

- ADMM-GAMP augmented Lagrangian

$$\min_{q_{\boldsymbol{x}},q_{\mathbf{z}},\boldsymbol{\tau}_p,\boldsymbol{u}} \max_{\mathbf{q},\mathbf{s},\boldsymbol{\tau}_s} L_A(q_{\boldsymbol{x}},q_{\mathbf{z}},\boldsymbol{\tau}_p,\boldsymbol{u},\mathbf{q},\mathbf{s},\boldsymbol{\tau}_s) \text{ with}$$

$$\begin{aligned} L_A &= D(q_{\boldsymbol{x}}||e^{-f_{\boldsymbol{x}}}) - \tfrac{1}{2}\boldsymbol{\tau}_s^T(\boldsymbol{\tau}_p - \mathbf{S}\,\mathrm{var}(\boldsymbol{x}|q_{\boldsymbol{x}})) + D(q_{\mathbf{z}}||e^{-f_{\mathbf{z}}}) \\ &+ H_G(q_{\mathbf{z}},\boldsymbol{\tau}_p) + \mathbf{q}^T(\mathbb{E}(\boldsymbol{x}|q_{\boldsymbol{x}}) - \boldsymbol{u}) + \mathbf{s}^T(\mathbb{E}(\mathbf{z}|q_{\mathbf{z}}) - \mathbf{A}\,\boldsymbol{u}) \\ &+ \tfrac{1}{2}\|\,\mathbb{E}(\boldsymbol{x}|q_{\boldsymbol{x}}) - \boldsymbol{u}\|^2_{\boldsymbol{\tau}_r} + \tfrac{1}{2}\|\,\mathbb{E}(\mathbf{z}|q_{\mathbf{z}}) - \mathbf{A}\,\boldsymbol{u}\|^2_{\boldsymbol{\tau}_p}, \end{aligned}$$

  For ADMM, the first two terms are the cost function for $q_{\boldsymbol{x}}$, the next two terms constitute the cost function for $q_{\mathbf{z}}$. The two groups of primal variables are $\{q_{\boldsymbol{x}}, q_{\mathbf{z}}\}$ and $\boldsymbol{u}$ (and the optimization of L_A is decoupled between $q_{\boldsymbol{x}}$, $q_{\mathbf{z}}$).
- The two linear constraints together constitute a single extended set of linear constraints with extended Lagrange multiplier $[\mathbf{q}^T\,\mathbf{s}^T]^T$. The appropriately weighted quadratic augmentation terms correspond exactly to the set of linear constraints. The optimization in [3] is organized with the *usual ADMM algorithm alternating* between minimizations over the two groups of primal variables, followed by the ADMM specific Lagrange multiplier update.
- The optimization over the remaining variable τ_p, τ_s is then performed in an **outer loop**.

From ADMM-GAMP to AMBGAMP (2)

- We show here (by the variance subsystem convergence analysis) that this organization in two levels is not necessary.
- Furthermore, there is a redundancy between the linear and quadratic constraint terms in ADMM Augmented Lagrangian. Indeed, if we impose the constrained Lagrange multiplier structure $\mathbf{q}^T = -\mathbf{s}^T \mathbf{A}$, then we obtain the AMBGAMP Augmented Lagrangian.

$$\mathbf{q}^T(\mathbb{E}(\boldsymbol{x}|q_{\boldsymbol{x}}) - \boldsymbol{u}) + \mathbf{s}^T(\mathbb{E}(\mathbf{z}|q_{\mathbf{z}}) - \mathbf{A}\,\boldsymbol{u}) = \mathbf{s}^T(\mathbb{E}(\mathbf{z}|q_{\mathbf{z}}) - \mathbf{A}\;\mathbb{E}(\boldsymbol{x}|q_{\boldsymbol{x}}))$$

- This is constrained enough since the Lagrange multiplier $\mathbf{s}$ will lead to $\mathbb{E}(\mathbf{z}|q_{\mathbf{z}}) = \mathbf{A}\;\mathbb{E}(\boldsymbol{x}|q_{\boldsymbol{x}})$, in which case the quadratic augmentation terms are minimized by $\boldsymbol{u} = \mathbb{E}(\boldsymbol{x}|q_{\boldsymbol{x}})$ and disappear.
- However, constraining $\mathbf{q}^T = -\mathbf{s}^T \mathbf{A}$ leads to a deviation from the strict ADMM structure and requires separate convergence analysis, which we provide here.

$$\begin{bmatrix} \mathbf{q} \\ \mathbf{s} \end{bmatrix} = \begin{bmatrix} -\mathbf{A}^T \\ \mathbf{I} \end{bmatrix} \mathbf{s}$$

Updating Schedule

- At iteration t we propose the following updating sequence

$$\{\boldsymbol{u}^t\} = \arg\min_{\boldsymbol{u}} L(q_{\boldsymbol{x}}^{t-1}, q_{\mathbf{z}}^t, \boldsymbol{\tau}_p^{t-1}, \boldsymbol{u}, \mathbf{s}^{t-1}, \boldsymbol{\tau}_s^{t-1}, \boldsymbol{\tau}_s^t, q_{\boldsymbol{\alpha}}^{t-1}, q_{\boldsymbol{\gamma}}^{t-1}) \tag{5}$$

$$\{q_{\mathbf{z}}^t\} = \arg\min_{q_{\mathbf{z}}} L(q_{\boldsymbol{x}}^{t-1}, q_{\mathbf{z}}, \boldsymbol{\tau}_p^{t-1}, \boldsymbol{u}^t, \mathbf{s}^{t-1}, \boldsymbol{\tau}_s^{t-1}, \boldsymbol{\tau}_s^t, q_{\boldsymbol{\alpha}}^{t-1}, q_{\boldsymbol{\gamma}}^{t-1}) \tag{6}$$

$$\{\mathbf{s}^t\} = \arg\max_{\mathbf{s}} L(q_{\boldsymbol{x}}^{t-1}, q_{\mathbf{z}}^t, \boldsymbol{\tau}_p^{t-1}, \boldsymbol{u}^t, \mathbf{s}, \boldsymbol{\tau}_s^{t-1}, \boldsymbol{\tau}_s^t, q_{\boldsymbol{\alpha}}^{t-1}, q_{\boldsymbol{\gamma}}^{t-1}) \tag{7}$$

$$\{\boldsymbol{\tau}_p^t, \boldsymbol{\tau}_s^t\} = \arg\min_{\boldsymbol{\tau}_p} \max_{\boldsymbol{\tau}_s} L(q_{\boldsymbol{x}}^{t-1}, q_{\mathbf{z}}^t, \boldsymbol{\tau}_p, \boldsymbol{u}^t, \mathbf{s}^t, \boldsymbol{\tau}_s, \boldsymbol{\tau}_s^t, q_{\boldsymbol{\alpha}}^{t-1}, q_{\boldsymbol{\gamma}}^{t-1}) \tag{8}$$

$$\{q_{\boldsymbol{x}}^t\} = \arg\min\nolimits_{q_{\boldsymbol{x}}} L(q_{\boldsymbol{x}}, q_{\mathbf{z}}^t, \boldsymbol{\tau}_p^t, \boldsymbol{u}^t, \mathbf{s}^t, \boldsymbol{\tau}_s^t, \boldsymbol{\tau}_s^t, q_{\boldsymbol{\alpha}}^{t-1}, q_{\boldsymbol{\gamma}}^{t-1}) \tag{9}$$

$$\{q_{\boldsymbol{\alpha}}^t, q_{\boldsymbol{\gamma}}^t\} = \arg\min\nolimits_{q_{\boldsymbol{\alpha}}, q_{\boldsymbol{\gamma}}} L(q_{\boldsymbol{x}}^t, q_{\mathbf{z}}^t, \boldsymbol{\tau}_p^t, \boldsymbol{u}^t, \mathbf{s}^t, \boldsymbol{\tau}_s^t, q_{\boldsymbol{\alpha}}, q_{\boldsymbol{\gamma}}). \tag{10}$$

Outline

Update of $\boldsymbol{u}$

- To update $\boldsymbol{u}$, we use a gradient descent method with line search to optimize the step-size. From (4), (5), we get

$$\begin{aligned} &L(q_{\boldsymbol{x}}^{t-1}, q_{\mathbf{z}}^{t-1}, \boldsymbol{\tau}_p^{t-1}, \boldsymbol{u}, \mathbf{s}^{t-1}, \boldsymbol{\tau}_s^{t-1}) \\ &= \tfrac{1}{2}\|\widehat{\boldsymbol{x}}^{t-1} - \boldsymbol{u}\|^2_{\boldsymbol{\tau}_r^{t-1}} + \tfrac{1}{2}\|\widehat{\mathbf{z}}^{t-1} - \mathbf{A}\,\boldsymbol{u}\|^2_{\boldsymbol{\tau}_p^{t-1}} + const. \end{aligned} \tag{11}$$

where $const.$ denotes constants w.r.t. $\boldsymbol{u}$. The minimizing update can be obtained as

$$\boldsymbol{u}^t = \boldsymbol{u}^{t-1} - \eta^t\,\mathbf{g}^t \tag{12}$$

with gradient $\mathbf{g}^t = \mathbf{g}^t(\boldsymbol{u}^{t-1})$ where

$$\begin{aligned} &\mathbf{g}^t(\boldsymbol{u}) = \nabla_{\boldsymbol{u}} L(q_{\boldsymbol{x}}^{t-1}, q_{\mathbf{z}}^{t-1}, \boldsymbol{\tau}_p^{t-1}, \boldsymbol{u}, \mathbf{s}^{t-1}, \boldsymbol{\tau}_s^{t-1}) \\ &= -\mathbf{A}^T((\widehat{\mathbf{z}}^{t-1} - \mathbf{A}\boldsymbol{u})./\boldsymbol{\tau}_p^{t-1}) - (\widehat{\boldsymbol{x}}^{t-1} - \boldsymbol{u})./\boldsymbol{\tau}_r^{t-1} \\ &= \mathbf{g}^t(\mathbf{0}) + \mathcal{H}^t\,\boldsymbol{u},\ \mathcal{H}^t = \mathbf{D}(\mathbf{1}./\boldsymbol{\tau}_r^{t-1}) + \mathbf{A}^T\mathbf{D}(\mathbf{1}./\boldsymbol{\tau}_p^{t-1})\mathbf{A} \end{aligned} \tag{13}$$

where $\mathbf{D}(\boldsymbol{\tau})$ denotes a diagonal matrix with diagonal elements $\boldsymbol{\tau}$. The step-size η^t gets optimized for maximum descent :

$$\begin{aligned} &\frac{\partial L(q_{\boldsymbol{x}}^{t-1}, q_{\mathbf{z}}^{t-1}, \boldsymbol{\tau}_p^{t-1}, \boldsymbol{u}^t, \mathbf{s}^{t-1}, \boldsymbol{\tau}_s^{t-1})}{\partial \eta^t} = 0 \\ &\Rightarrow\ \eta^t = \|\mathbf{g}^t\|^2 / \mathbf{g}^{t\,T}\mathcal{H}^t\mathbf{g}^t\,. \end{aligned} \tag{14}$$

Update of $q_{\mathbf{z}}$

- Consider the relevant Lagrangian terms

$$
\begin{aligned}
&L(q_{\boldsymbol{x}}^{t-1}, q_{\mathbf{z}}, \boldsymbol{\tau}_p^{t-1}, \boldsymbol{u}^t, \mathbf{s}^{t-1}, \boldsymbol{\tau}_s^{t-1}, q_{\boldsymbol{\alpha}}^{t-1}, q_{\boldsymbol{\gamma}}^{t-1}) \\
&= D(q_{\mathbf{z}} q_{\boldsymbol{\gamma}} || e^{-f_{\mathbf{z},\gamma}}) + \tfrac{1}{2} \mathsf{var}(\mathbf{z}|q_{\mathbf{z}}) ./ \boldsymbol{\tau}_p^{t-1} \\
&+ \mathbf{s}^{t-1\,T}\, \mathbb{E}(\mathbf{z}|q_{\mathbf{z}}) + \tfrac{1}{2} \| \mathbb{E}(\mathbf{z}|q_{\mathbf{z}}) - \mathbf{A}\,\boldsymbol{u}^t \|^2_{\boldsymbol{\tau}_p^{t-1}} + const. \\
&= D(q_{\mathbf{z}} q_{\boldsymbol{\gamma}} || e^{-f_{\mathbf{z},\gamma}}) + \tfrac{1}{2}\, \mathbb{E}(\mathbf{z}^T \mathbf{z}|q_{\mathbf{z}}) ./ \boldsymbol{\tau}_p^{t-1} \\
&- (\mathbb{E}(\mathbf{z}|q_{\mathbf{z}}))^T ((\mathbf{A}\,\boldsymbol{u}^t) ./ \boldsymbol{\tau}_p^{t-1} - \mathbf{s}^{t-1}) + const. \\
&= D(q_{\mathbf{z}} q_{\boldsymbol{\gamma}} || e^{-f_{\mathbf{z},\gamma}}) + \tfrac{1}{2}\, \mathbb{E}(\|\mathbf{z} - \boldsymbol{p}^t\|^2_{\boldsymbol{\tau}_p^{t-1}} | q_{\mathbf{z}}) + const.
\end{aligned}
\tag{15}
$$

where $const.$ denotes constants w.r.t. $\mathbf{z}$ and

$$
\boldsymbol{p}^t = \mathbf{A}\,\boldsymbol{u}^{t-1} - \mathbf{s}^{t-1} . \boldsymbol{\tau}_p^{t-1} . \tag{16}
$$

This cost function is separable. We get per component

$$
\begin{aligned}
&\min_{q_{z_k}} D(q_{z_k} q_{\gamma_k} || g_{z_k}^t / Z_{z_k}^t) \;\Rightarrow\; q_{z_k}^t = \tilde{g}_{z_k}^t / Z_{z_k}^t \text{ with} \\
&\tilde{g}_{z_k}^t = e^{\mathbb{E}_{q_{\gamma_k}^{t-1}} \ln g_{z_k}^t}, \; Z_{z_k}^t = \int \tilde{g}_{z_k}^t \, dz_k \,, \; -\ln \tilde{g}_{z_k}^t \\
&= \mathbb{E}_{q_{\gamma_k}^{t-1}} f_{z_k,\gamma_k}(z_k, \gamma_k) + \frac{1}{2\tau_{p_k}^{t-1}} [(z_k - p_k^t)^2 - (p_k^t)^2]
\end{aligned}
\tag{17}
$$

Cumulant Generating Function

- Note that the partition function $Z_{z_k}^t$ acts as cumulant generating function:

$$\begin{aligned}
-\frac{\partial \ln Z_{z_k}^t}{\partial s_k} &= \mathbb{E}(z_k|q_{z_k}^t) = \mathbb{E}(z_k|p_k^t, \tau_{p_k}^{t-1}, \widehat{\gamma}_k^{t-1}) = \widehat{z}_k^t \\
\frac{\partial^2 \ln Z_{z_k}^t}{\partial s_k^2} &= \mathsf{var}(z_k|p_k^t, \tau_{p_k}^{t-1}, \widehat{\gamma}_k^{t-1}) = \tau_{z_k}^t \\
-\frac{\partial^3 \ln Z_{z_k}^t}{\partial s_k^3} &= \mathbb{E}((z_k - \mathbb{E}\, z_k)^3|q_{z_k}^t).
\end{aligned} \tag{18}$$

- The case of Gaussian noise leads again to a Gaussian posterior $q_{\mathbf{z}}$ with

$$\begin{aligned}
\mathbf{1}./\boldsymbol{\tau}_z^t &= \mathbf{1}./\boldsymbol{\tau}_p^{t-1} + \boldsymbol{\gamma}^{t-1}, \\
\widehat{\mathbf{z}}^t &= \boldsymbol{\tau}_z^t.(\boldsymbol{y}.\boldsymbol{\gamma}^{t-1} + \boldsymbol{p}^t./\boldsymbol{\tau}_p^{t-1}).
\end{aligned} \tag{19}$$

Update of $\mathbf{s}$ (ADMM style)

- Although the quadratic augmentation terms in the Lagrangian do not correspond exactly to a weighted quadratic version of the linear mean constraint, due to the introduction of the auxiliary variable $\boldsymbol{u}$ which streamlines the derivation of the updates of $q_{\boldsymbol{x}}$ and $q_{\mathbf{z}}$, nevertheless an ADMM style update of the mean constraint Lagrange multiplier $\mathbf{s}$ is possible. Indeed, the terms in (15) that contains $\mathbf{s}$ or $\widehat{\mathbf{z}}$ are

$$\widehat{\mathbf{z}}^T((\frac{1}{2}\widehat{\mathbf{z}} - \boldsymbol{p}^t)./\boldsymbol{\tau}_p^{t-1}) = \widehat{\mathbf{z}}^T(\mathbf{s}^{t-1} + (\frac{1}{2}\widehat{\mathbf{z}} - \mathbf{A}\boldsymbol{u}^t)./\boldsymbol{\tau}_p^{t-1}) \tag{20}$$

Taking the gradient w.r.t. $\widehat{\mathbf{z}}$ (as part of the optimization over $q_{\mathbf{z}}$) leads to the RHS of

$$\mathbf{s}^t = \mathbf{s}^{t-1} + (\widehat{\mathbf{z}}^t - \mathbf{A}\boldsymbol{u}^t)./\boldsymbol{\tau}_p^{t-1}\,. \tag{21}$$

Hence, if we use this update for $\mathbf{s}$, then (20) reduces to $\widehat{\mathbf{z}}^T\mathbf{s}^t$, as if the quadratic augmentation terms have disappeared!
This is the main characteristic of the Lagrange multiplier update in ADMM, which corresponds to a gradient ascent with a particular choice of (diagonal) step-size.

Update of $\{\boldsymbol{\tau}_p, \boldsymbol{\tau}_s\}$

- In [9], [10], the carefully chosen updating schedule made the quadratic augmentation terms inactive when updating $\{\boldsymbol{\tau}_p, \boldsymbol{\tau}_s\}$. Here these terms only become inactive at convergence. Nevertheless, these terms only play an active role for the means and not for the variances. Hence, we shall ignore them here.
- Hence the terms of interest in (5) are

$$\begin{aligned} &L(q_{\boldsymbol{x}}^{t-1}, q_{\mathbf{z}}^{t}, \boldsymbol{\tau}_p, \boldsymbol{u}^t, \mathbf{s}^t, \boldsymbol{\tau}_s) \\ &= H_G(q_{\mathbf{z}}^{t}, \boldsymbol{\tau}_p) - \tfrac{1}{2}\boldsymbol{\tau}_s^T(\boldsymbol{\tau}_p - \mathbf{S}\,\boldsymbol{\tau}_x^{t-1}) + const. = const.+ \\ &\tfrac{1}{2}\sum_{k=1}^{M}\left[\frac{\tau_{z_k}^t}{\tau_{p_k}} + \ln(2\pi\,\tau_{p_k})\right] - \frac{1}{2}\sum_{k=1}^{M}\tau_{s_k}(\tau_{p_k} - \mathbf{S}_{k,:}\,\boldsymbol{\tau}_x^{t-1}) \end{aligned} \tag{22}$$

where $const.$ denotes constants w.r.t. $\{\boldsymbol{\tau}_p, \boldsymbol{\tau}_s\}$. Deriving w.r.t. $\{\boldsymbol{\tau}_p, \boldsymbol{\tau}_s\}$ yields the feasibility conditions

$$\frac{\partial L}{\partial \tau_{s_k}} = 0 \qquad \Rightarrow \quad \tau_{p_k}^t = \mathbf{S}_{k,:}\,\boldsymbol{\tau}_x^{t-1} \tag{23}$$

$$\frac{\partial L}{\partial \tau_{p_k}} = \frac{1}{2}\left(-\frac{\tau_{z_k}^t}{\tau_{p_k}^2} + \frac{1}{\tau_{p_k}} - \tau_{s_k}\right) = 0 \quad \Rightarrow \quad \tau_{s_k}^t = \frac{1}{\tau_{p_k}^t}\left(1 - \frac{\tau_{z_k}^t}{\tau_{p_k}^t}\right) \tag{24}$$

which we run as a fixed-point sub-algorithm.
The position of these updates in the updating schedule is less important. Nevertheless we update $\{\boldsymbol{\tau}_p, \boldsymbol{\tau}_s\}$ as soon as the quantities on which they depend have been updated.

Update of Hyperparameter $\boldsymbol{\alpha}$

- The relevant part of the augmented Lagrangian simply leads to a KLD term

$$q_{\boldsymbol{\alpha}}^t = \arg\min_{q_{\boldsymbol{\alpha}}} D(q_{\boldsymbol{x}}^t q_{\boldsymbol{\alpha}} || e^{-f_{\boldsymbol{x},\boldsymbol{\alpha}}}) + const. \tag{25}$$

  which gets minimized alternatingly as in VB [12]

$$\ln q_{\alpha_i}^t = (\frac{1}{2} + a - 1) \ln \alpha_i - (\frac{1}{2} \mathbb{E}_{q_{x_i}^t}(x_i^2) + b) \alpha_i + const. \tag{26}$$

- This means that the posterior of α_i is a Gamma distribution: $q_{\alpha_i}^t = \mathcal{G}(\alpha_i; \widehat{a}, \widehat{b}^t)$, with $\widehat{a} = a + 1/2$ and $\widehat{b}^t = \mathbb{E}_{q_{x_i}^t}(x_i^2)/2 + b$, with mean $\frac{\widehat{a}}{\widehat{b}^t}$ or hence:

$$\widehat{\alpha}_i^t = \frac{2a+1}{\mathbb{E}_{q_{x_i}^t}(x_i^2) + 2b}. \tag{27}$$

Update of Hyperparameter $\boldsymbol{\gamma}$

- The relevant part of the augmented Lagrangian is again a KLD term:

$$q_{\boldsymbol{\gamma}}^t = \arg \min_{q_{\boldsymbol{\gamma}}(\boldsymbol{\gamma})} D(q_{\mathbf{z}}^t q_{\boldsymbol{\gamma}} || e^{-f_{\mathbf{z},\gamma}}) + const. \tag{28}$$

  which gets minimized alternatingly as in VB [12]

$$\ln q_{\gamma_k}^t = (\frac{1}{2} + c - 1) \ln \gamma_i - (\frac{\mathbb{E}_{q_{z_k}^t} (y_k - z_k)^2}{2} + d)\gamma_i + const. \tag{29}$$

  which means again that the posterior of γ_k is a Gamma distribution $q_{\gamma_k}^t = \mathcal{G}(\gamma_k; \widehat{c}, \widehat{d}^t)$, with $\widehat{c} = c + 1/2$ and $\widehat{d}^t = \mathbb{E}_{q_{\gamma_k}^t} (y_k - z_k)^2/2 + d$, and with mean $\frac{\widehat{c}}{\widehat{d}^t}$ or:

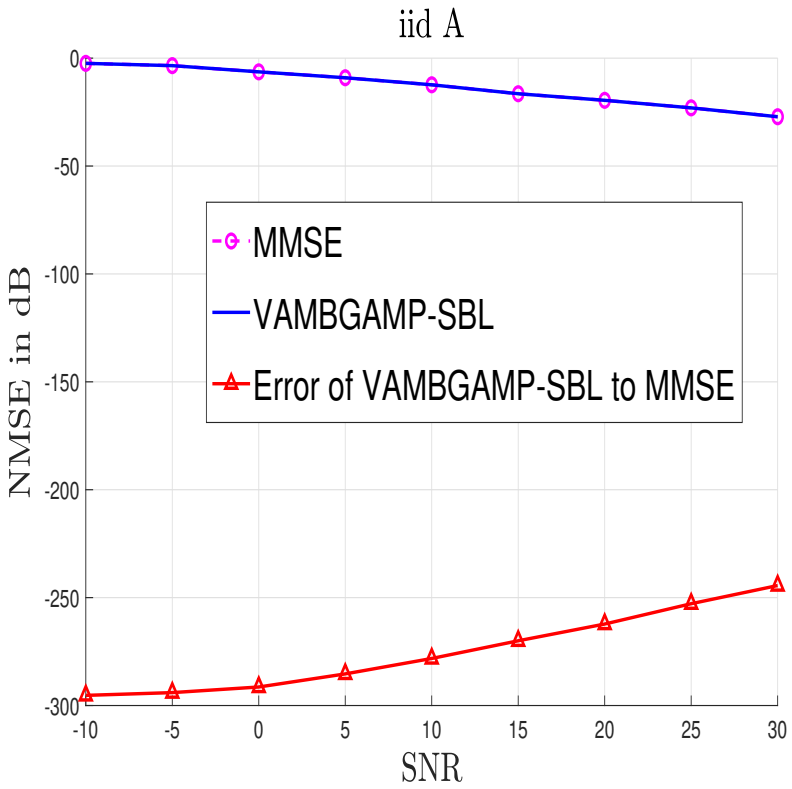
$$\gamma_k^t = \frac{2c+1}{\mathbb{E}_{q_{z_k}^t} (y_k - z_k)^2 + 2d}. \tag{30}$$

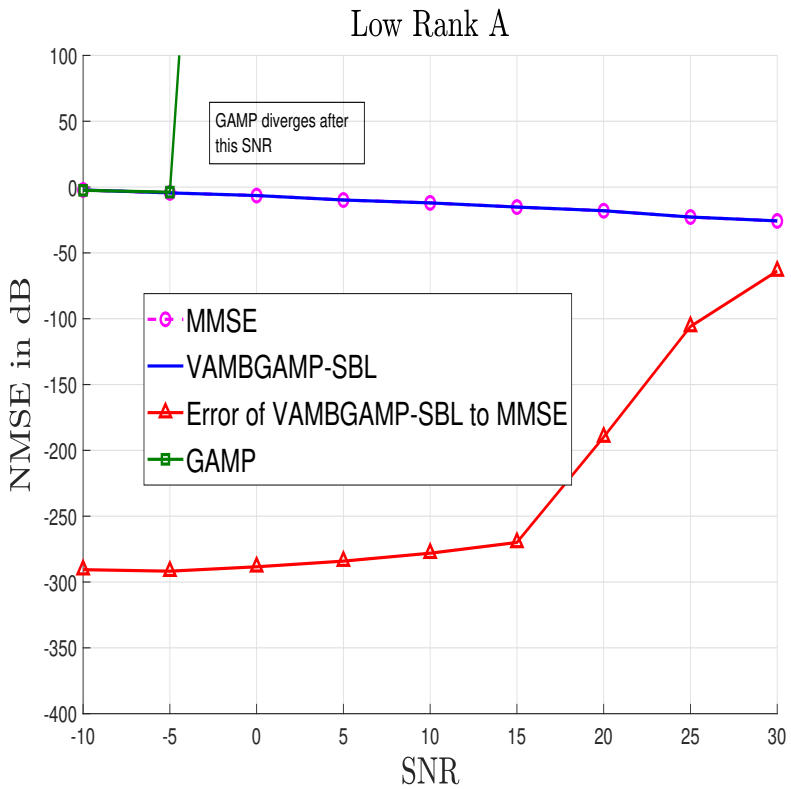
- For the case in which all noise variances are assumed to be equal, the update of $\gamma_0 = \gamma_k, \forall k$ can be shown to be [12]:

$$\gamma_0^t = \frac{2c+M}{\mathbb{E}_{q_{\mathbf{z}}^t} (\|\boldsymbol{y} - \mathbf{z}\|^2) + 2d}. \tag{31}$$

Outline

The figures show Normalized MSE $\boldsymbol{\tau}_x$ (2 top curves) and normalized MSE difference between $\boldsymbol{x}$ estimated by AMBAMP and LMMSE (2 bottom curves) with variance profile $\sigma^2_{x_i} = 0.93^{i-1}$, $i = 1, \ldots, N$, for $M = 150, N = 250$. Left is for i.i.d. Gaussian $\mathbf{A}$ whereas right is for low rank $\mathbf{A}$ in which the smallest half of the singular values in an i.i.d. $\mathbf{A}$ are set to zero. These simulations show that the VAMBGAMP-SBL algorithm continues to workin an unrealistically severe scenario, in which AMP diverges.





A Few References I

[1] R. Giri and Bhaskar D. Rao,
"Type I and Type II Bayesian Methods for Sparse Signal Recovery using Scale Mixtures,"
IEEE Trans. on Sig Process., 2018.

[2] X. Tan and J. Li,
"Computationally Efficient Sparse Bayesian Learning via Belief Propagation ,"
IEEE Trans. Sig. Proc., Apr. 2013.

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

[4] S. Rangan, P. Schniter, and A. K. Fletcher,
"Vector Approximate Message Passing,"
IEEE Trans. On Info. Theo., Oct. 2019.

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

[6] Q. Guo and J. Xi,
"Approximate message passing with unitary transformation,"
arxiv.org/abs/1504.04799, 2015.

[7] 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.

[8] S. Rangan,
"Generalized Approximate Message Passing for Estimation with Random Linear Mixing,"
in *Proc. IEEE Int. Symp. Inf. Theory*, Saint Petersburg, Russia, 2011,
extended version: arxiv1010.5141.

A Few References II

[9] D. Slock",
"Convergent Approximate Message Passing by Alternating Constrained Minimization of Bethe Free Energy,"
in *IEEE 7th Forum on Research and Technologies for Society and Industry Innov. (RTSI)*, Paris, France, Aug. 2022.

[10] D. Slock",
"Convergent Approximate Message Passing,"
in *IEEE Int'l Mediterr. Conf. Comm's and Netw'ing (MeditCom)*, Athens, Greece, Sep. 2022.

[11] C.K. Thomas and D. Slock,
"Alternating Constrained Minimization based Approximate Message Passing,"
in *IEEE Int'l Conf. Acoustics, Speech and Sig. Proc. (ICASSP)*, 2023.

[12] C. K. Thomas and D. Slock,
"SAVE - space alternating variational estimation for sparse Bayesian learning,"
in *Data Science Workshop*, Jun. 2018.