



THE UNIVERSITY *of* EDINBURGH

Edinburgh Research Explorer

Upscaling Vector Approximate Message Passing

Citation for published version:

Skuratovs, N & Davies, M 2020, Upscaling Vector Approximate Message Passing. in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* . vol. 2020, 19787434 , International Conference on Acoustics, Speech, and Signal Processing (ICASSP) , Institute of Electrical and Electronics Engineers, 2020 IEEE International Conference on Acoustics, Speech, and Signal Processing, Barcelona, Spain, 4/05/20. <https://doi.org/10.1109/ICASSP40776.2020.9053799>

Digital Object Identifier (DOI):

[10.1109/ICASSP40776.2020.9053799](https://doi.org/10.1109/ICASSP40776.2020.9053799)

Link:

[Link to publication record in Edinburgh Research Explorer](#)

Document Version:

Peer reviewed version

Published In:

ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)

General rights

Copyright for the publications made accessible via the Edinburgh Research Explorer is retained by the author(s) and / or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy

The University of Edinburgh has made every reasonable effort to ensure that Edinburgh Research Explorer content complies with UK legislation. If you believe that the public display of this file breaches copyright please contact openaccess@ed.ac.uk providing details, and we will remove access to the work immediately and investigate your claim.



UPSCALING VECTOR APPROXIMATE MESSAGE PASSING

Nikolajs Skuratovs, Michael Davies

School of Engineering and Electronics, The University of Edinburgh

ABSTRACT

In this paper we consider the problem of recovering a signal \mathbf{x} of size N from noisy and compressed measurements $\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{w}$ of size M , where the measurement matrix \mathbf{A} is right-orthogonally invariant (ROI). Vector Approximate Message Passing (VAMP) demonstrates great reconstruction results for even highly ill-conditioned matrices \mathbf{A} in relatively few iterations. However, performing each iteration is challenging due to either computational or memory point of view. On the other hand, a recently proposed Conjugate Gradient (CG) Expectation Propagation (CG-EP) framework is able to sacrifice some performance for efficiency, but requires access to exact singular spectrum of \mathbf{A} . In this work we develop a CG-VAMP algorithm that does not require such information, is feasible to implement and converges to the neighborhood of the original VAMP.

Index Terms— Compressed Sensing, Vector Approximate Message Passing, Expectation Propagation

1. INTRODUCTION

Consider the recovery of a random signal $\mathbf{x} \in \mathbb{R}^N$ from a general linear model

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{w} \quad (1)$$

where $\mathbf{y} \in \mathbb{R}^M$ is the measurement vector, $\mathbf{w} \in \mathbb{R}^M$ is a zero-mean i.i.d. Gaussian noise vector $\mathbf{w} \sim N(0, v_w \mathbf{I}_M)$ and $\mathbf{A} \in \mathbb{R}^{M \times N}$ is a measurement matrix. We consider the compressed sensing scenario $M \ll N$ with the ratio $\frac{M}{N} = \delta = O(1)$ fixed. In this work we assume that \mathbf{x} has finite 4th order moment and variance $v_x = \lim_{N \rightarrow \infty} \frac{1}{N} \mathbf{x}^T \mathbf{x}$. Additionally, following [6], we assume that in the SVD of $\mathbf{A} = \mathbf{U}\mathbf{S}\mathbf{V}^T$ the matrix \mathbf{V} is Haar distributed and therefore \mathbf{A} is a right-orthogonally invariant (ROI) matrix.

The Vector Approximate Message Passing (VAMP) algorithm [6], [2] is an iterative algorithm that estimates the signal \mathbf{x} from (1) and can be derived from the Expectation Propagation (EP) framework. VAMP can achieve Bayes-optimal reconstruction results [6] when \mathbf{A} is ROI and the two estimators involved in VAMP are chosen to be Bayes optimal. While the

first estimator, the *denoiser* \mathbf{g}_B , tries to estimate the signal \mathbf{x} from a zero-mean i.i.d. Gaussian channel, the second estimator, the *LMMSE estimator* \mathbf{g}_A , is supposed to estimate an intrinsic i.i.d. Gaussian signal \mathbf{x}_0 substituted in (1) from \mathbf{y} . The challenge in implementing VAMP is the design of these estimators and computation of their parameters, divergences and Mean Squared Error (MSE) estimates, that ensures correct and efficient operation of the algorithm.

Designing good denoisers \mathbf{g}_B and computing the corresponding parameters is a relatively studied problem (refer to [3] and [7] and references therein). In contrast there has been much less focus on efficiently implementing the LMMSE estimator (with the exception of [11] and [8]). The LMMSE estimator operating in the VAMP algorithm requires computing a linear mapping that involves inverting either an $N \times N$ [6] or $M \times M$ [10] matrix. Inversion of such a matrix has complexity $O(N^3)$ and for large-scale problems is not feasible. This problem persists even when there exist fast implementations of the measurement matrix, \mathbf{A} . In the original VAMP [6], this computation was addressed by precomputing the SVD of the measurement matrix \mathbf{A} , reducing the required computation to order $O(N^2)$. However, when both N and M are of order of hundred of thousands or more, as in CT imaging, storing the matrix of the singular vectors becomes not possible. In such applications even the measurement matrix \mathbf{A} might not be stored, but only computed on-the-fly.

The computational/memory problem was partially addressed in [11], where the authors suggested to approximate the LMMSE solution with the Conjugate Gradient (CG) algorithm, which only requires computation of matrix-vector products with $\mathbf{A}\mathbf{A}^T$, and proposed a rigorous method for estimating the key parameters of the algorithm. However, estimation of those parameters required the computation of the moments of the singular spectrum of $\mathbf{A}\mathbf{A}^T$ of order up to $2i + 2$, where i is the number of CG iterations. Even if these moments were available (which generally they are not), for highly ill-conditioned \mathbf{A} , the higher order moments of $\mathbf{A}\mathbf{A}^T$ are not feasible to manipulate with on devices with finite precision arithmetic. A final weakness of the CG approximation to the EP algorithm (CG-EP) is that it has worse convergence guarantees due to the LMMSE approximation which can result in inferior phase transition performance.

The contributions of this paper are twofold. First, we develop a practical method of estimating the key parameters for

This work was supported by the ERC project C-SENSE (ERC-ADG-2015-694888). MD is also supported by a Royal Society Wolfson Research Merit Award. We would like to thank James Hopgood and Alessandro Perelli for useful discussions.

the CG-EP algorithm without referring to the singular moments of \mathbf{A} . Second, the proposed method is used to construct the auto-tuned CG algorithm that computes its own divergence and the MSE of the produced estimate, and chooses such a number of iterations that ensures consistent progression of the overall CG-VAMP algorithm. The per-iteration cost of the proposed CG algorithm is only twice larger than of the CG algorithm in [11].

The paper is organized as follows. In the second section we discuss the key ingredients and limitations of the VAMP algorithm and consider the main constraints of its approximated version – CG-EP algorithm. In the third section it is introduced the main theoretical tool for the practical implementation of the CG-VAMP algorithm. In the next section the proposed tool is used to construct the auto-tuned CG algorithm that ensures certain per-iteration progression of the CG-VAMP algorithm. The last section is dedicated to numerical experiments of the proposed methods.

2. EXPECTATION PROPAGATION BASED ALGORITHMS

For the measurement system (1) with ROI \mathbf{A} , iterative recovery of the signal \mathbf{x} can be addressed by a family of algorithms derived from Expectation Propagation (EP). Vector Approximated Message Passing (VAMP) is one of the first EP-motivated inference algorithms designed for recovery of \mathbf{x} from measurements (1). Although next we concentrate on VAMP algorithm, we note that VAMP [6], the exact EP for ROI measurements [10] and the CG-EP [11] are all instances of a general EP-based framework shown in Algorithm 1.

The EP-based framework from Algorithm 1 represents iterative passing of the messages $\mathbf{x}_{A \rightarrow B}^t$ and $\mathbf{x}_{B \rightarrow A}^t$ between two blocks: Block A and Block B. Under some mild conditions on \mathbf{g}_A and \mathbf{g}_B [2] and for ROI matrices \mathbf{A} , it can be shown that in the LSL these messages are equivalent to the true signal \mathbf{x} corrupted by some additive noise

$$\mathbf{x}_{A \rightarrow B}^t = \mathbf{x} + \mathbf{h}_t \quad (2)$$

$$\mathbf{x}_{B \rightarrow A}^t = \mathbf{x} + \mathbf{q}_t \quad (3)$$

Notably, due to the Haar matrix \mathbf{V} and the choice of the scalars γ_A^t and γ_B^t , the noise vector \mathbf{h}_t can be shown to be equivalent to a zero-mean i.i.d. Gaussian vector that is independent of \mathbf{x} [10]. Variances $v_{A \rightarrow B}^t$ and $v_{B \rightarrow A}^t$ of both noise vectors \mathbf{h}_t and \mathbf{q}_t respectively define the current state of the algorithm. At each iteration t , the functions \mathbf{g}_A and \mathbf{g}_B attempt to reduce the variances of the noise terms.

Calculation of the two messages requires the computation of two pairs of scalars: γ_A^t and γ_B^t , and $\tilde{v}_{A \rightarrow B}^t$ and $\tilde{v}_{B \rightarrow A}^t$. Here, the scalars γ_A^t and γ_B^t correspond to (the inverse of the) divergence of the functions \mathbf{g}_A and \mathbf{g}_B and allows the progress of the algorithm to be described by a simple one dimensional State Evolution (SE). Additionally, the functions f_A and f_B produce estimates $\tilde{v}_{A \rightarrow B}^t$ and $\tilde{v}_{B \rightarrow A}^t$ of the true variances $v_{A \rightarrow B}^t$ and $v_{B \rightarrow A}^t$.

Algorithm 1: EP-based Algorithm

Initialization: $\mathbf{x}_{B \rightarrow A}^0 = 0, \tilde{v}_{B \rightarrow A}^0 = \tilde{v}_x, t = 0$

1 **while** $t < T_{max}$ **and** $\tilde{v}_{B \rightarrow A}^t \geq \epsilon$ **do**

2 **Block A**

$$\mu_A^t = \mathbf{g}_A(\mathbf{x}_{B \rightarrow A}^t, \tilde{v}_{B \rightarrow A}^t, \tilde{v}_w)$$

$$\frac{1}{\gamma_A^t} = \frac{1}{N} \nabla_{(\mathbf{x}_{B \rightarrow A}^t)} \cdot \left(\mathbf{A}^T \mathbf{g}_A(\mathbf{x}_{B \rightarrow A}^t, \tilde{v}_{B \rightarrow A}^t, \tilde{v}_w) \right)$$

$$\mathbf{x}_{A \rightarrow B}^t = \mathbf{x}_{B \rightarrow A}^t - \gamma_A^t \mathbf{A}^T \mu_A^t$$

$$\tilde{v}_{A \rightarrow B}^t = f_A(\mu_A^t, \mathbf{x}_{B \rightarrow A}^t, \tilde{v}_{B \rightarrow A}^t, \tilde{v}_w)$$

3 **Block B**

$$\mu_B^{t+1} = \mathbf{g}_B(\mathbf{x}_{A \rightarrow B}^t, \tilde{v}_{A \rightarrow B}^t)$$

$$\gamma_B^{t+1} = \frac{1}{N} \nabla_{(\mathbf{x}_{A \rightarrow B}^t)} \cdot \mathbf{g}_B(\mathbf{x}_{A \rightarrow B}^t, \tilde{v}_{A \rightarrow B}^t)$$

$$\mathbf{x}_{B \rightarrow A}^{t+1} = \frac{1}{1 - \gamma_B^{t+1}} \left(\mu_B^{t+1} - \gamma_B^{t+1} \mathbf{x}_{A \rightarrow B}^t \right)$$

$$\tilde{v}_{B \rightarrow A}^{t+1} = f_B(\mu_B^{t+1}, \mathbf{x}_{B \rightarrow A}^t, \tilde{v}_{A \rightarrow B}^t)$$

4 $t = t + 1$

Output: μ_B^t

In the VAMP version of Algorithm 1, the function \mathbf{g}_B is chosen to estimate \mathbf{x} from the message $\mathbf{x}_{A \rightarrow B}^t$, which from (2) allows \mathbf{g}_B to be interpreted as a *denoiser*. Today, there are numerous Plug-and-Play methods for denoising signal from a Gaussian channel, including BM3D [1], Convolutional Denoisers/Neural Networks [2] etc. Additionally, computing the divergence and MSE estimates for these denoisers can be done using Monte-Carlo SURE techniques [5] [9].

A completely different picture is observed in the Block A. In VAMP [6] and in exact EP for ROI matrices \mathbf{A} [10], the function \mathbf{g}_A is the linear mapping

$$\mathbf{g}_A(\mathbf{x}_{B \rightarrow A}^t) = \mathbf{W}_t^{-1} (\mathbf{y} - \mathbf{A} \mathbf{x}_{B \rightarrow A}^t) = \mathbf{z}_t \quad (4)$$

where the matrix \mathbf{W}_t is

$$\mathbf{W}_t = \tilde{v}_w \mathbf{I}_M + \tilde{v}_{B \rightarrow A}^t \mathbf{A} \mathbf{A}^T \quad (5)$$

and \tilde{v}_w is an estimate of the variance of the measurement noise \mathbf{w} in (1). For this choice of \mathbf{g}_A , the scalar γ_A^t corresponds to

$$(\gamma_A^t)^{-1} = (N)^{-1} \text{Tr} \{ \mathbf{A}^T \mathbf{W}_t^{-1} \mathbf{A} \} \quad (6)$$

Additionally, in the LSL, an unbiased estimate $\tilde{v}_{A \rightarrow B}^t$ of would have the following simple form [10]

$$\tilde{v}_{A \rightarrow B}^t = \gamma_A^t - \tilde{v}_{B \rightarrow A}^t$$

Despite the potential optimality of the resulting algorithm, computing the mapping (4) and the corresponding divergence γ_A^t (6) is costly from either a computational or a memory point of view. In [11] the authors addressed this by approximating (4) using the CG algorithm shown on Algorithm 2. The new CG-EP algorithm involved the function $\mathbf{g}_A^{i[t]}$ corresponding to $i[t]$ iterations of the CG algorithm that approximates the solution to

$$\mathbf{W}_t \mu_A^t = \mathbf{y} - \mathbf{A} \mathbf{x}_{B \rightarrow A}^t = \mathbf{z}_t \quad (7)$$

The authors further show that CG-EP has a new State Evolution (SE) ([11], Theorem 1) and derive the expressions for the divergence $\gamma_A^{t,i[t]}$ with $i[t]$ iterations and for the variance $v_{A \rightarrow B}^t$. Both of the expressions depend on moments

Algorithm 2: CG Algorithm for approximating $\mathbf{W}\boldsymbol{\mu} = \mathbf{z}$

Initialization: $\boldsymbol{\mu} = \mathbf{0}, \mathbf{r}^0 = \mathbf{p}^0 = \mathbf{z}, i = 0$

```

1 while  $i < i_{max}$  do
2    $\alpha^i = \frac{\|\mathbf{r}^i\|^2}{\|\mathbf{W}^{1/2}\mathbf{p}^i\|^2}$ 
3    $\boldsymbol{\mu}^{i+1} = \boldsymbol{\mu}^i + \alpha^i \mathbf{p}^i$ 
4    $\mathbf{r}^{i+1} = \mathbf{r}^i - \alpha^i \mathbf{W}\mathbf{p}^i$ 
5    $\beta^i = \frac{\|\mathbf{r}^{i+1}\|^2}{\|\mathbf{r}^i\|^2}$ 
6    $\mathbf{p}^{i+1} = \mathbf{r}^{i+1} + \beta^i \mathbf{p}^i = \mathbf{z} - \mathbf{W}\boldsymbol{\mu}^{i+1} + \beta^i \mathbf{p}^i$ 
7    $i = i + 1$ 

```

Output: $\boldsymbol{\mu}^{i+1}$

$\xi_j = \int \lambda^j \rho(\lambda)$, where $\rho(\lambda)$ is the LSL density function of eigenvalues of $\mathbf{A}\mathbf{A}^T$ [11]. More precisely, it was shown that calculation of the two scalars requires the moments ξ_j of order up to $j = 2i + 2$. In practice, the suggested method for computing $\gamma_A^{t,i[t]}$ and $v_{A \rightarrow B}^t$ might be impossible for the reasons discussed before.

3. MONTE-CARLO-BASED ESTIMATES

We consider estimation of $\gamma_A^{t,i[t]}$ and $v_{A \rightarrow B}^t$ in the LSL, where we expect to observe high accuracy of the Strong Law of Large Numbers (for details, see Theorem 1 in [10]). Although, in general CG iterations constitute nonlinear mappings, as shown in [9], in the LSL, the CG approximation of (7) with $i[t]$ iterations $\mathbf{g}_A^{i[t]}$ is the following linear mapping

$$\mathbf{g}_A^{i[t]} = \mathbf{U}\mathbf{H}_t^{i[t]}\mathbf{U}^T \quad (8)$$

where \mathbf{U} is the matrix of left-singular vectors of \mathbf{A} and the matrix \mathbf{H}_t is a diagonal matrix with entries depending only on singular spectrum of \mathbf{A} , v_w and $v_{B \rightarrow A}^t$. Note that (8) also does not depend on the vector being mapped ($\mathbf{z}_t = \mathbf{y} - \mathbf{A}\mathbf{x}_{B \rightarrow A}^t$), but only depends on the variances of vectors \mathbf{w} and \mathbf{q}_t involved in \mathbf{y} and $\mathbf{x}_{B \rightarrow A}^t$ respectively. Additionally, by using the definition of vector \mathbf{y} from (1) and the message $\mathbf{x}_{B \rightarrow A}^t$ from (3), the vector \mathbf{z}_t can be rewritten as

$$\mathbf{z}_t = \mathbf{y} - \mathbf{A}\mathbf{x}_{B \rightarrow A}^t = \mathbf{w} - \mathbf{A}\mathbf{q}_t = \mathbf{w} - \mathbf{U}\Sigma\mathbf{V}^T\mathbf{q}_t \quad (9)$$

which is independent of the signal \mathbf{x} . Furthermore, it was proved in [6] and [10] that the vector

$$\mathbf{V}^T\mathbf{q}_t = \mathbf{b}_t \sim N(0, v_{B \rightarrow A}^t \mathbf{I}_N) \quad (10)$$

and is independent of \mathbf{w} , \mathbf{U} and Σ . By using (8), (9) and (10) we conclude that the output of the CG algorithm

$$\boldsymbol{\mu}_A^{t,i[t]} = \mathbf{U}\mathbf{H}_t^{i[t]}\mathbf{U}^T(\mathbf{w} - \mathbf{U}\Sigma\mathbf{b}_t)$$

corresponds to a linear mapping of a Gaussian vector. We therefore propose the following method for estimating $\gamma_A^{t,i[t]}$ and $v_{A \rightarrow B}^t$. The first step is to generate two *synthetic* vectors

$$\dot{\mathbf{y}} = \mathbf{0} + \dot{\mathbf{w}} \sim N(\mathbf{0}, \mathbf{v}_w \mathbf{I}_M)$$

$$\dot{\mathbf{x}}_{B \rightarrow A}^t = \mathbf{0} + \dot{\mathbf{q}}_t \sim N(\mathbf{0}, \mathbf{v}_{B \rightarrow A}^t \mathbf{I}_N)$$

where the $\mathbf{0}$ highlights that we have effectively set the synthetic signal $\dot{\mathbf{x}} = \mathbf{0}$. We then form the corresponding synthetic vector $\dot{\mathbf{z}}_t$

$$\dot{\mathbf{z}}_t = \dot{\mathbf{y}} - \mathbf{A}\dot{\mathbf{x}}_{B \rightarrow A}^t = \dot{\mathbf{w}} - \mathbf{A}\dot{\mathbf{q}}_t$$

and execute the CG algorithm with the original matrix \mathbf{W}_t on the synthesized vector $\dot{\mathbf{z}}_t$

$$\dot{\boldsymbol{\mu}}_A^{t,i[t]} = \mathbf{g}_A^{i[t]}(\dot{\mathbf{z}}_t) = \mathbf{g}_A^{i[t]}(\dot{\mathbf{w}} - \mathbf{U}\Sigma\mathbf{V}^T\dot{\mathbf{q}}_t)$$

Lastly, we show that

$$\begin{aligned} \lim_{N \rightarrow \infty} \frac{1}{N} \dot{\mathbf{q}}_t^T \mathbf{A}^T \dot{\boldsymbol{\mu}}_A^{t,i[t]} &\stackrel{(a)}{=} \lim_{N \rightarrow \infty} \frac{1}{N} E[\dot{\mathbf{q}}_t^T \mathbf{A}^T \dot{\boldsymbol{\mu}}_A^{t,i[t]}] \\ &= \lim_{N \rightarrow \infty} \frac{1}{N} \text{Tr} \left\{ \text{Cov} \left[\mathbf{V}^T \dot{\mathbf{q}}_t, \mathbf{g}_A^{i[t]}(\dot{\mathbf{z}}) \right] \mathbf{U} \Sigma \right\} \\ &\stackrel{(b)}{=} \lim_{N \rightarrow \infty} \frac{1}{N} \text{Tr} \left\{ \text{Cov} \left[\mathbf{V}^T \dot{\mathbf{q}}_t, \mathbf{V}^T \dot{\mathbf{q}}_t \right] \nabla_{(\mathbf{V}^T \dot{\mathbf{q}}_t)} \mathbf{g}_A^{i[t]}(\dot{\mathbf{z}}) \mathbf{U} \Sigma \right\} \\ &\stackrel{(c)}{=} \lim_{N \rightarrow \infty} v_{B \rightarrow A}^t \frac{1}{N} \text{Tr} \left\{ \Sigma^T \mathbf{H}_t^{i[t]} \mathbf{U}^T \mathbf{U} \Sigma \right\} = v_{B \rightarrow A}^t \frac{1}{\gamma_A^{t,i[t]}} \end{aligned}$$

where in (a) we used the Strong Law of Large Numbers, in (b) we used (10) and the Stein's Lemma and in (c) we used (8). Since the divergence $\dot{\gamma}_A^{t,i[t]}$ for the synthesized data is only a function of \mathbf{A} , v_w and $v_{B \rightarrow A}^t$, in the limit, the two terms will be the same

$$\lim_{N \rightarrow \infty} \dot{\gamma}_A^{t,i[t]} = \gamma_A^{t,i[t]}$$

Therefore we propose the following estimator of $(\gamma_A^{t,i[t]})^{-1}$

$$\frac{1}{\dot{\gamma}_A^{t,i[t]}} = \frac{\frac{1}{N} \dot{\mathbf{q}}_t^T \mathbf{A}^T \mathbf{g}_A^{i[t]}(\dot{\mathbf{z}}_t)}{\tilde{v}_{B \rightarrow A}^t} \quad (11)$$

Calculation of an estimate of the variance $v_{A \rightarrow B}^t$ is done in a similar way. By definition of $v_{A \rightarrow B}^t$ we have

$$\lim_{N \rightarrow \infty} v_{A \rightarrow B}^t = \lim_{N \rightarrow \infty} \frac{1}{N} \|\mathbf{x}_{A \rightarrow B}^t - x\|^2 \quad (12)$$

which depends only on the singular spectrum of \mathbf{A} , v_w and $v_{B \rightarrow A}^t$. Therefore, substituting the synthesized data in (12) and calculating the resulting norm will produce the same result in the LSL. Thus we propose to use the following estimator of the variance $v_{A \rightarrow B}^t$

$$\tilde{v}_{A \rightarrow B}^t = \frac{1}{N} \|\dot{\mathbf{q}}_t + \tilde{\gamma}_A^{t,i[t]} \mathbf{A}^T \dot{\boldsymbol{\mu}}_A^{t,i[t]}\|^2 \quad (13)$$

4. AUTO-TUNED CG

By approximating the exact LMMSE solution using the CG algorithm, the resulting CG-VAMP sacrifices convergence rates and guarantees. As proved for VAMP [6] and CG-EP in [11], the progression and the fixed points of the two algorithms can be defined through the 1D State Evolution (SE) $v_{t+1} = SE(v_t)$. Because the LMMSE estimator is the optimal g_A [6] and the SE is 1D, then we have [4]

$$SE_{LMMSE}(v_t) \leq SE_{CG(i[t])}(v_t)$$

Moreover, by choosing insufficient number of CG iterations, the variance at iteration $t + 1$ may not reduce below the variance at iteration t , i.e. $v_{t+1} \geq v_t$, which is equivalent to the algorithm converging to a new spurious fixed point.

To ensure progression of Block A, the proposed CG algorithm uses the following simple rule

$$\tilde{v}_{A \rightarrow B}^t(i) \leq c \tilde{v}_{A \rightarrow B}^{t-1} \quad (14)$$

where a scalar c represents a bound on the expected variance drop in VAMP (until it is in the neighborhood of the VAMP

fixed point) after each iteration t . Once the inequality (14) is met or after a maximum number of iterations, the CG algorithm is terminated.

In the proposed CG algorithm, the variance $v_{A \rightarrow B}^t(i)$ is estimated using (13), which requires calculation of $\tilde{\gamma}_A^{t,i[t]}$ first. Using (11) and noting that the matrix-vector product $\mathbf{A}\hat{\mathbf{q}}_t$ needs to be computed only once, the calculation of $\tilde{\gamma}_A^{t,i[t]}$ requires only $O(N)$ additional computations per iteration i . On the other hand, direct evaluation of (13) requires computing $\mathbf{A}^T \hat{\boldsymbol{\mu}}_t^i$ and would lead to a doubling the complexity. Instead, by expanding the norm and using the definition of $\tilde{\gamma}_A^{t,i[t]}$, one can show that (13) is equal to

$$\tilde{v}_{A \rightarrow B}^t = (N)^{-1} (\tilde{\gamma}_A^{t,i[t]})^2 (\hat{\boldsymbol{\mu}}_t^{i[t]})^T \mathbf{A} \mathbf{A}^T \hat{\boldsymbol{\mu}}_t^{i[t]} - v_{B \rightarrow A}^t \quad (15)$$

Next, we expand the matrix-vector product $\mathbf{W}_t \hat{\boldsymbol{\mu}}_t^{i+1}$ as

$$\mathbf{W}_t \hat{\boldsymbol{\mu}}_t^{i+1} = \sigma_w \hat{\boldsymbol{\mu}}_t^{i+1} + v_{B \rightarrow A}^t \mathbf{A} \mathbf{A}^T \hat{\boldsymbol{\mu}}_t^{i+1}$$

Next, for CG with synthesized data we use the scalars α_t^j and β_t^j from the CG with the real data, and exclude computing the residual vector r^{i+1} from Algorithm 2 by using the second option of calculating the vector p^{i+1} on line 6. Now, since the matrix-vector product $\mathbf{A} \mathbf{A}^T \hat{\boldsymbol{\mu}}_t^{i+1}$ is a part of the CG algorithm, computing (15) costs only $O(N)$ computations.

5. SIMULATION RESULTS

In this section we empirically investigate the quality of the estimate $\tilde{v}_{A \rightarrow B}^t$ suggested in (15), and the performance of the auto-tuned CG-VAMP in recovering \mathbf{x} from a highly ill-conditioned measurement system (1). We consider a Bernoulli-Gaussian signal \mathbf{x} of size 2^{14} in the same settings as in the original VAMP paper [6]. We consider the measurement matrix \mathbf{A} with geometric singular values, which is one of the worst cases, and condition number 10000. The compression rate is chosen to be $\delta = 0.5$. The auto-tuned CG algorithm iterates until the inequality (14) is met with the scalar $c = 0.9$. The total number of outer iterations t was chosen to be 50 and we average the results over 100 realizations.

Figure 1(a) plots the change of $\tilde{v}_{A \rightarrow B}^t$ over iterations t . The main line demonstrates the mean, while the shaded region shows the standard deviation of the variance over 100 realizations. From the plot we observe the consistent reduction of the variance after each iteration t as it was suggested by the inequality (14).

The Figure 1(b) illustrates the mean and the standard deviation of normalized squared difference of the estimated using (15) and true variance $v_{A \rightarrow B}^t$. As we see, even for moderately large systems, the quality of the estimator (15) is satisfying and only gets better as the algorithm progresses.

In the Figure 2 it is shown the average and the standard deviation of the number of CG iterations needed to satisfy the inequality (14) at each iteration of the CG-VAMP. As we see, the required number of CG iterations varies as CG-VAMP progresses, which confirms the necessity in having a tool for adaptively choosing the number of CG iterations.

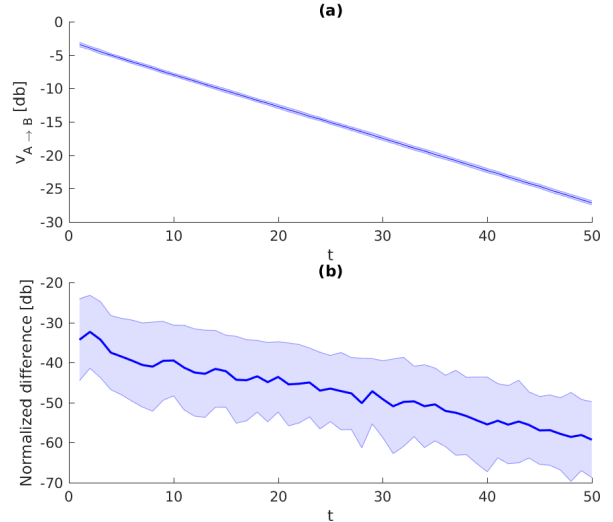


Fig. 1: Mean and standard deviation of $v_{A \rightarrow B}^t$ in (a) and of its difference from $\tilde{v}_{A \rightarrow B}^t$ in (b) versus iteration number t

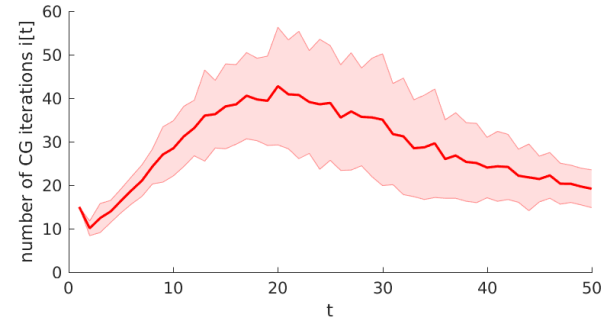


Fig. 2: Mean and standard deviation of CG iterations versus iterations number t

6. CONCLUSIONS

In this work, we developed a practical implementation of approximated VAMP that requires neither inversion of large matrices nor precomputing SVD of \mathbf{A} , nor information about the singular spectrum of \mathbf{A} . The CG-VAMP algorithm is guaranteed to consistently reduce the MSE of the intrinsic measurements given that the exact VAMP is capable of doing so.

The proposed algorithm relies on the strong statistical properties of the Haar matrix \mathbf{V} and with more practical measurement operators \mathbf{A} like FFT matrices, CT operators etc, one might employ damping to increase the stability as was suggested in [8]. On the other hand, these types of matrices have fast implementation w.r.t. matrix-vector products and therefore the per-iteration complexity of the CG-VAMP will reduce significantly. Analysis of the performance of the proposed CG-VAMP algorithm with structured \mathbf{A} is left as a further work.

7. REFERENCES

References

- [1] K. Dabov et al. “Image Denoising by Sparse 3-D Transform-Domain Collaborative Filtering”. In: *IEEE Transactions on Image Processing* 16.8 (Aug. 2007), pp. 2080–2095. DOI: 10.1109/TIP.2007.901238.
- [2] Alyson K. Fletcher et al. “Plug-in Estimation in High-Dimensional Linear Inverse Problems: A Rigorous Analysis”. In: *CoRR* abs/1806.10466 (2018). arXiv: 1806.10466. URL: <http://arxiv.org/abs/1806.10466>.
- [3] Chunli Guo and Mike E. Davies. “Near optimal compressed sensing without priors: Parametric SURE Approximate Message Passing”. In: *CoRR* abs/1409.0440 (2014). arXiv: 1409.0440. URL: <http://arxiv.org/abs/1409.0440>.
- [4] Christopher A. Metzler, Arian Maleki, and Richard G. Baraniuk. “From Denoising to Compressed Sensing”. In: *CoRR* abs/1406.4175 (2014). arXiv: 1406.4175. URL: <http://arxiv.org/abs/1406.4175>.
- [5] Sathish Ramani, Thierry Blu, and Michael Unser. “Monte-Carlo SURE: A Black-Box Optimization of Regularization Parameters for General Denoising Algorithms”. In: *IEEE transactions on image processing : a publication of the IEEE Signal Processing Society* 17 (Oct. 2008), pp. 1540–54. DOI: 10.1109/TIP.2008.2001404.
- [6] Sundeep Rangan, Philip Schniter, and Alyson K. Fletcher. “Vector Approximate Message Passing”. In: *CoRR* abs/1610.03082 (2016). arXiv: 1610.03082. URL: <http://arxiv.org/abs/1610.03082>.
- [7] Philip Schniter, Sundeep Rangan, and Alyson K. Fletcher. “Denoising based Vector Approximate Message Passing”. In: *CoRR* abs/1611.01376 (2016). arXiv: 1611.01376. URL: <http://arxiv.org/abs/1611.01376>.
- [8] Philip Schniter, Sundeep Rangan, and Alyson K. Fletcher. *Plug-and-play Image Recovery using Vector AMP*. 2017. URL: http://www2.ece.ohio-state.edu/~schniter/pdf/basp17_poster.pdf.
- [9] Charles M. Stein. “Estimation of the Mean of a Multivariate Normal Distribution”. In: *Ann. Statist.* 9.6 (Nov. 1981), pp. 1135–1151. DOI: 10.1214/aos/1176345632. URL: <https://doi.org/10.1214/aos/1176345632>.
- [10] Keigo Takeuchi. “Rigorous Dynamics of Expectation-Propagation-Based Signal Recovery from Unitarily Invariant Measurements”. In: *CoRR* abs/1701.05284 (2017). arXiv: 1701.05284. URL: <http://arxiv.org/abs/1701.05284>.
- [11] Keigo Takeuchi and Chao-Kai Wen. “Rigorous dynamics of expectation-propagation signal detection via the conjugate gradient method”. In: (July 2017), pp. 1–5. DOI: 10.1109/SPAWC.2017.8227654.