Divergence Estimation in Message Passing algorithms

Divergence Estimation in Message Passing algorithms

Nikolajs Skuratovs, Michael Davies, *Fellow IEEE*

***Abstract*—Many modern imaging applications can be modeled as compressed sensing linear inverse problems. When the measurement operator involved in the inverse problem is sufficiently random, denoising Scalable Message Passing (SMP) algorithms have a potential to demonstrate high efficiency in recovering compressed data. One of the key components enabling SMP to achieve fast convergence, stability and predictable dynamics is the Onsager correction that must be updated at each iteration of the algorithm. This correction involves the denoiser's divergence that is traditionally estimated via the Black-Box Monte Carlo (BB-MC) method [1]. While the BB-MC method demonstrates satisfying accuracy of estimation, it requires heuristic tuning and executing the denoiser additional times at each iteration and might lead to a substantial increase in computational cost of the SMP algorithms. In this work we develop two Large System Limit models of the Onsager correction for denoisers operating within SMP algorithms and use these models to propose practical black-box methods for divergence estimation that require no additional executions of the denoiser and demonstrate similar correction compared to the BB-MC method.**

***Index Terms*—Message Passing, Divergence Estimation, Denoiser, Onsager Correction, Expectation Propagation**

I. Introduction

In this work we consider a particular sub-problem that arises in certain iterative methods designed to recover a signal $\mathbf{x} \in \mathbb{R}^N$ from a set of linear measurements

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{w} \tag{1}$$

where $\mathbf{y} \in \mathbb{R}^M$, $\mathbf{w} \in \mathbb{R}^M$ is a zero-mean i.i.d. Gaussian noise vector $\mathbf{w} \sim \mathcal{N}(0, v_w \mathbf{I}_M)$ and $\mathbf{A} \in \mathbb{R}^{M \times N}$ is a measurement matrix that is assumed to be available. We consider the large scale compressed sensing scenario $M < N$ with a subsampling factor $\delta = \frac{M}{N} = O(1)$.

While there are many first-order iterative methods for recovering $\mathbf{x}$ from the set of measurement (1) including [2]–[5] and many others, in this work we focus on the family of *Scalable Message Passing (SMP)* algorithms that includes Approximate Message Passing (AMP) [6], Orthogonal AMP (OAMP) [7], Vector AMP (VAMP) [8], Conjugate Gradient VAMP (CG-VAMP) [9]–[11], Warm-Started CG-VAMP (WS-CG-VAMP) [10], Convolutional AMP (CAMP) [12], Memory AMP (MAMP) and others. When the measurement operator $\mathbf{A}$ comes from a certain family of random matrices, which may be different for each example of SMP, these algorithms demonstrate high per-iteration improvement compared to other first-order methods and stable and predictable dynamics. Additionally, it is evidenced that SMP algorithms can recover complex signals like natural images by employing powerful Plug-and-Play (PnP) denoisers like BM3D [13], Non-Local Means [14], Denoising CNN [15] and others, and demonstrate State-of-The-Art performance for certain examples of $\mathbf{A}$ [16].

On a general level, an SMP algorithm is an iterative method with a linear step followed by a denoising step. It can be shown [12], [15], [17], [18] that one can be flexible with the choice of denoisers in SMP as long as the key ingredient, the divergence of the denoiser at each iteration, can be computed to form a so-called *Onsager Correction* for the denoiser. In the literature on SMP algorithms [10], [15], [16], [19]–[21] and others, the suggested method for computing the divergence of a PnP denoiser is the Black-Box Monte Carlo (BB-MC) method [1]. The BB-MC method computes an estimate of the divergence of a function $\boldsymbol{f}(\mathbf{x})$ that admits a well-defined second-order Taylor expansion by executing this function again at the points $\mathbf{x} + \epsilon\mathbf{n}$ with the scalar ϵ approaching zero and where $\mathbf{n}$ is a zero-mean i.i.d. random vector with a unit variance and finite higher order moments. Then one can show that the divergence $\frac{1}{N}\nabla_{\mathbf{x}} \cdot \boldsymbol{f}(\mathbf{x}) = \frac{1}{N}\sum_{i=1}^{N} \frac{\partial(\boldsymbol{f}(\mathbf{x}))_i}{\partial x_i}$ of $\boldsymbol{f}$ is equivalent to [1]

$$\frac{1}{N}\nabla_{\mathbf{x}} \cdot \boldsymbol{f}(\mathbf{x}) = \lim_{\epsilon \to 0} \mathbb{E}_{\mathbf{n}}\left[\mathbf{n}^T\left(\frac{\boldsymbol{f}(\mathbf{x}+\epsilon\mathbf{n}) - \boldsymbol{f}(\mathbf{x})}{\epsilon}\right)\right] \tag{2}$$

To approximate the expectation operator in (2), one can use MC trials and implement the inner product inside of the expectation multiple times and average the results. However, given that the function $\boldsymbol{f}$ is of the appropriate class and the dimension of $\mathbf{x}$ is sufficiently large, one can often obtain a satisfactory accuracy of divergence estimation with only a single trial.

While this approach provides a practical method for the divergence estimation and leads to stable dynamics of SMP algorithms, it has two drawbacks. First, it assumes that the chosen denoiser $\boldsymbol{f}$ admits a well-defined second-order Taylor expansion, which is not the case for denoisers like BM3D and for ReLU based CNNs [15] that involve non-linear operations like thresholding as subroutines. This violation might result in unsatisfactory accuracy of the estimation and lead to the necessity for additional MC trials. Additionally one can no longer use too small values of ϵ as in this case the estimator (2) becomes unstable [1], which leads to the necessity to tune this parameter very carefully and, to the best of our knowledge, there is no rigorous method for this. As a result, often one needs to empirically tune the scalar ϵ for each denoiser individually to ensure the stability of the estimator. For example, from our experiments, the value of ϵ for which BB-MC produces accurate divergence estimates varies by an order for BM3D and for a DnCNN.

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.

The second problem with the BB-MC method is that it requires executing the denoiser one or more additional times. When the dimension of the inverse problem is large, as in modern computational imaging tasks, executing powerful denoisers can be the dominant cost of the algorithm and it is desired to execute it as infrequently as possible.

In this work we study the dynamics of SMP algorithms and develop two rigorous asymptotic models for the divergence of a PnP denoiser used within the algorithm. These models lead to two divergence estimation techniques that do not either require additional executions of the denoisers, nor empirical tuning. The first method works in a complete black-box fashion and has a minimal computational cost dominated by two inner-products of N-dimensional vectors, but has inferior estimation accuracy compared to BB-MC and the second method. The second method uses only the information generated within any SMP algorithm, has a computational complexity dominated by one matrix-vector product with $\mathbf{A}$ and has a similar or superior accuracy of the divergence estimation compared to the hand-tuned BB-MC method. When an SMP algorithm incorporates a powerful denoiser such as BM3D, using the proposed methods for divergence estimation instead of BB-MC leads to almost halving the computational time of the algorithm. We numerically compare the proposed methods against the BB-MC method in the context of AMP, VAMP, CG-VAMP and WS-CG-VAMP used for recovering natural images from compressed measurements.

A. Notations

We use roman v for scalars, small boldface $\mathbf{v}$ for vectors and capital boldface $\mathbf{V}$ for matrices. We frequently use the identity matrix $\mathbf{I}_N$ with a subscript to define that this identity matrix is of dimension N or without a subscript where the dimensionality is clear from the context. We define $Tr\{\mathbf{M}\}$ to be the trace of a matrix $\mathbf{M}$, $\kappa(\mathbf{M})$ to be the condition number of $\mathbf{M}$ and use $\mathbf{M}^\dagger$ to be the left pseudo-inverse matrix of a full-rank matrix $\mathbf{M}$, $\mathbf{M}^\dagger = (\mathbf{M}^T\mathbf{M})^{-1}\mathbf{M}^T$. We use $||\cdot||_k$ to define the l_k norm and $||\cdot||$ specifically for l_2 norm. The divergence of a function $\boldsymbol{f}(\mathbf{x})$ with respect to the vector $\mathbf{x}$ is defined as $\nabla_{\mathbf{x}} \cdot f(\mathbf{x}) = \sum_{i=1}^{N} \frac{\partial_i}{\partial x_i} \boldsymbol{f}(\mathbf{x})$. By writing $q(x) = \mathcal{N}(\mathbf{m}, \mathbf{\Sigma})$ we mean that the density $q(x)$ is normal with mean vector $\mathbf{m}$ and covariance matrix $\mathbf{\Sigma}$. We reserve the letter t for the outer-loop iteration number of the EP- and VAMP-based algorithms. Lastly, we use the notation *i.i.d.* for a shorthand of *independent and identically distributed*.

II. Background on SMP algorithms

In this section we briefly review the structure and the main properties of SMP algorithms to set up the context of the paper. For more details on a specific SMP algorithm, please refer to [7], [10], [12], [15]–[17], [22] and the references therein.

A. General Message Passing framework

In this work, we consider SMP algorithms that alternate between the following linear and denoising steps [17]

$$\mathbf{r}_t = \frac{1}{C_r}\Big(\mathbf{A}^T \boldsymbol{f}_t(\mathbf{S}_{t+1}, \mathbf{y}) - \mathbf{S}_{t+1}\boldsymbol{\gamma}_t\Big) \tag{3}$$

$$\mathbf{s}_{t+1} = \frac{1}{C_s}\Big(\mathbf{g}_t(\mathbf{r}_t) - \mathbf{r}_t\alpha_t\Big) \tag{4}$$

which is initialized with $\mathbf{s}_0 = \mathbf{0}$ and where $\mathbf{S}_{t+1} = (\mathbf{s}_0, ..., \mathbf{s}_t)$. The update of the denoising step $\mathbf{s}_{t+1}$ involves a denoiser $\mathbf{g}_t(\mathbf{r}_t)$ that acts on the intrinsic channel $\mathbf{r}_t = \mathbf{x} + \mathbf{h}_t$, where

$$\mathbf{h}_t = \mathbf{r}_t - \mathbf{x} \tag{5}$$

is modeled as an i.i.d. Gaussian noise vector independent of $\mathbf{x}$. The denoiser output is corrected with the *Onsager term* $\mathbf{r}_t\alpha_t$ that involves the divergence α_t of the denoiser $\mathbf{g}_t(\mathbf{r}_t)$

$$\alpha_t = \frac{1}{N}\nabla_{\mathbf{r}_t} \cdot \mathbf{g}_t(\mathbf{r}_t). \tag{6}$$

The main purpose of the Onsager term is to ensure that the input error $\mathbf{h}_t$ is orthogonal to the resulting output error

$$\mathbf{q}_{t+1} = \mathbf{s}_{t+1} - \mathbf{x}. \tag{7}$$

Ensuring the orthogonality between $\mathbf{q}_{t+1}$ and $\mathbf{h}_t$ leads to stable and efficient operation of the SMP algorithms. Lastly, except for the AMP case where $C_s = 1$, the normalization scalar C_s is usually chosen to be $C_s = 1 - \alpha_t$ [8].

Similarly, the linear step $\mathbf{r}_t$ involves a linear function $\boldsymbol{f}_t$ and the Onsager term $\mathbf{S}_{t+1}\boldsymbol{\gamma}_t$. The structure of $\boldsymbol{f}_t$ depends on the chosen SMP algorithm and how it processes the residual vector

$$\mathbf{z}_t = \mathbf{y} - \mathbf{A}\mathbf{s}_t. \tag{8}$$

From the fixed point perspective, under certain assumptions discussed below, assuming $\mathbf{A}$ is uniformly drawn from the set of orthogonal matrices and given that the replica prediction is correct for Haar matrices, the optimal choice of the function $\boldsymbol{f}_t$ is the LMMSE estimator

$$\boldsymbol{f}_t(\mathbf{S}_{t+1}, \mathbf{y}) = \mathbf{W}_t^{-1}\mathbf{z}_t \tag{9}$$

that is proposed to use within the VAMP algorithm [8]. Here the matrix $\mathbf{W}_t$ is

$$\mathbf{W}_t = (v_w\mathbf{I} + v_{q_t}\mathbf{A}\mathbf{A}^T)^{-1} \tag{10}$$

where v_{q_t} models the variance of the intrinsic error $\mathbf{q}_t$. The other SMP algorithms incorporate a suboptimal but scalable alternative to the optimal linear estimator (9). For example, MF-OAMP implements the naive approximation $\mathbf{W}_t^{-1} = \mathbf{I}$ so that $\boldsymbol{f}_t(\mathbf{S}_{t+1}, \mathbf{y}) = \mathbf{z}_t$, while in CG-VAMP [9], [10], the function $\boldsymbol{f}_t(\mathbf{S}_{t+1}, \mathbf{y})$ approximates the linear mapping (9) with i iterations of the zero-initialized Conjugate Gradient (CG) algorithm. In these three cases, the update of $\mathbf{r}_t$ is *single-memory* since the function $\boldsymbol{f}_t(\mathbf{S}_{t+1}, \mathbf{y})$ depends on only the last output of the denoising step so $\mathbf{S}_{t+1} = \mathbf{s}_t$. On the other hand, the algorithms like WS-CG-VAMP, MAMP and CAMP approximate the LMMSE estimator (9) using the whole history of vectors $\mathbf{S}_{t+1} = (\mathbf{s}_0, ..., \mathbf{s}_t)$ in order to achieve better accuracy. These types of algorithms will be referred as *long-memory* SMP algorithms. Lastly, we have the original AMP[1]

[1]Originally, AMP was formulated as a single-memory asymmetric algorithm [6], but in this work we will use the unified error framework from [17] that analyzes Message Passing algorithms that takes the symmetric form (3)-(4). In [17], the authors showed that one can reformulate AMP in a symmetric form with t-long memory update of $\mathbf{r}_t$, as in (3).

algorithm, which can also be mapped into the long-memory version following the structure (3)-(4) [17].

Similarly, to the denoising step, the update (3) involves an Onsager term $\mathbf{S}_{t+1}\boldsymbol{\gamma}_t$, where $\boldsymbol{\gamma}_t = (\gamma_t^0, \gamma_t^1, ..., \gamma_t^t)^T$ and

$$\gamma_t^\tau = \frac{1}{N}\nabla_{\mathbf{s}_\tau} \cdot \mathbf{A}^T \boldsymbol{f}_t(\mathbf{S}_{t+1}, \mathbf{y}) \tag{11}$$

is the divergence of $\mathbf{A}^T \boldsymbol{f}_t(\mathbf{S}_{t+1}, \mathbf{y})$ with respect to each vector $\mathbf{s}_\tau$ that $\boldsymbol{f}_t$ depends on. These Onsager terms perform a similar role to α_t for the denoising step $\mathbf{s}_{t+1}$ – ensuring the orthogonality between $\mathbf{h}_t$ and $\mathbf{s}_\tau$ for $\tau = 0, ..., t$. The closed-form solution to these scalars can be found in the works related to a specific SMP algorithm. Specifically, in [17] the authors showed that when AMP is mapped into the structure (3)-(4), we have $\boldsymbol{\gamma}_t = \mathbf{0}$ and there is no Onsager correction required for the linear step.

Lastly, the normalization scalar C_r in (3) is usually computed as

$$C_r = -\sum_{\tau=0}^{t} \gamma_t^\tau, \tag{12}$$

except for the AMP case, where $C_r = 1$.

B. Error dynamics of Message Passing algorithms

When an SMP algorithm follows the general structure (3)-(4), the dynamics of the error vector $\mathbf{h}_t$ and $\mathbf{q}_t$ from (5) and (7) can be rigorously defined under the following assumptions [12], [17]

Assumption 1: The dimensions of the signal model N and M approach infinity with a fixed ratio $\delta = \frac{M}{N} = O(1)$

Assumption 2: $\mathbf{A}$ is normalized so that $\frac{1}{N}Tr\{\mathbf{A}\mathbf{A}^T\} = 1$. Additionally,

1) For AMP: The measurement matrix $\mathbf{A}$ is orthogonally invariant, such that in the SVD of $\mathbf{A} = \mathbf{U}\mathbf{S}\mathbf{V}^T$, the matrices $\mathbf{U}$ and $\mathbf{V}$ are independent of other random terms and are uniformly distributed on the set of orthogonal matrices, while the matrix $\mathbf{S}^T\mathbf{S}$ has the limiting eigenvalue distribution with the first t moments equal to the first t moments of Marčhenko-Pastur distribution [23], where t is the maximum number of iterations of AMP.
2) For the rest of the algorithms mentioned at the end of Section II.A: The same condition on $\mathbf{V}$, while $\mathbf{U}$ is allowed to be any orthogonal matrix and the matrix $\mathbf{S}^T\mathbf{S}$ is allowed to have any Limiting Eigenvalue Distribution with a compact support. For those cases, we say $\mathbf{A}$ is right-orthogonally invariant (ROI).

Assumption 3: The denoiser $\mathbf{g}_t$ is uniformly Lipschitz so that the sequence of functions $\mathbf{g}_t : \mathbb{R}^N \mapsto \mathbb{R}^N$ indexed by N are Lipschitz continuous with a Lipschitz constant $L_N < \infty$ as $N \to \infty$ [15], [24]. Additionally, we assume the following limits exist almost surely [15]

$$\lim_{N\to\infty} \frac{1}{N}\mathbf{g}_t(\mathbf{x}+\mathbf{d}_1)^T\mathbf{g}_t(\mathbf{x}+\mathbf{d}_2), \quad \lim_{N\to\infty} \frac{1}{N}\mathbf{x}^T\mathbf{g}_t(\mathbf{x}+\mathbf{d}_1),$$
$$\lim_{N\to\infty} \frac{1}{N}\mathbf{d}_1^T\mathbf{g}_t(\mathbf{x}+\mathbf{d}_2), \quad \lim_{N\to\infty} \frac{1}{N}\mathbf{x}^T\mathbf{d}_1, \quad \lim_{N\to\infty} \frac{1}{N}||\mathbf{x}||^2$$

and the Stein's Lemma [25] holds for $\mathbf{g}_t$

$$\lim_{N\to\infty} \alpha_t = \lim_{N\to\infty} \frac{1}{\mathbf{C}_{1,2}} \frac{1}{N}\mathbf{d}_2^T\mathbf{g}_t(\mathbf{x}+\mathbf{d}_1). \tag{13}$$

Here α_t is the divergence of $\mathbf{g}_t$ as in (6) and $\mathbf{d}_1, \mathbf{d}_2 \in \mathbb{R}^N$ with $(\mathbf{d}_{1,n}, \mathbf{d}_{2,n}) \sim \mathcal{N}(\mathbf{0}, \mathbf{C})$ for some positive definite $\mathbf{C} \in \mathbb{R}^2$.

In the work [15] it is confirmed that the above assumptions are satisfied by *group-based denoisers*, *convolutional denoisers*, *Convolutional Neural Networks* with a Lipschitz separable activation function, such as sigmoid or ReLU, and *singular-value thresholding denoisers*.

Under these assumptions, it is possible to establish the following theorem.

Theorem 1. *[15]: Let Assumptions 1-3 hold. For $\tau = 0, 1, ...$ and $\tau' = 0, 1, ..., \tau$ we have that*

1) $\mathbf{h}_\tau$ and $\mathbf{q}_{\tau'}$ are asymptotically orthogonal

$$\lim_{N\to\infty} \frac{1}{N}\mathbf{h}_\tau^T\mathbf{q}_{\tau'} \overset{a.s.}{=} 0 \tag{14}$$

and $\mathbf{q}_{\tau'}$ satisfies

$$\lim_{N\to\infty} \frac{1}{N}\mathbf{w}^T\mathbf{A}\mathbf{q}_{\tau'} \overset{a.s.}{=} 0 \tag{15}$$

2) In the limit $N \to \infty$, the matrices

$$\mathbf{H}_{t+1} = (\mathbf{h}_0, ..., \mathbf{h}_t) \tag{16}$$
$$\mathbf{Q}_{t+1} = (\mathbf{q}_0, ..., \mathbf{q}_t) \tag{17}$$

are full rank almost surely.

3) $\mathbf{h}_\tau$ and $\mathbf{b}_{\tau'} = \mathbf{V}^T\mathbf{q}_{\tau'}$ follow

$$\mathbf{h}_\tau = \breve{\boldsymbol{h}}_\tau + \boldsymbol{o}\big(\mathbf{H}_\tau, \mathbf{Q}_{\tau+1}\big) \tag{18}$$
$$\mathbf{b}_{\tau'} = \breve{\boldsymbol{b}}_{\tau'} + \boldsymbol{o}\big(\mathbf{H}_{\tau'}, \mathbf{Q}_{\tau'}\big) \tag{19}$$

where $\breve{\boldsymbol{h}}_\tau$ and $\breve{\boldsymbol{b}}_{\tau'}$ are zero-mean i.i.d. Gaussian vectors satisfying

$$\lim_{N\to\infty} \frac{1}{N}||\breve{\boldsymbol{h}}_\tau||^2 \overset{a.s.}{=} \lim_{N\to\infty} \frac{1}{N}||\mathbf{h}_\tau||^2 = v_{h_\tau} < \infty \tag{20}$$
$$\lim_{N\to\infty} \frac{1}{N}||\breve{\boldsymbol{b}}_{\tau'}||^2 \overset{a.s.}{=} \lim_{N\to\infty} \frac{1}{N}||\mathbf{b}_{\tau'}||^2 = v_{q_{\tau'}} < \infty \tag{21}$$

and the vectors $\boldsymbol{o}\big(\mathbf{H}_\tau, \mathbf{Q}_{\tau+1}\big) \in \mathbb{R}^N$ and $\boldsymbol{o}\big(\mathbf{H}_{\tau'}, \mathbf{Q}_{\tau'}\big) \in \mathbb{R}^N$ satisfy

$$\lim_{N\to\infty} \frac{1}{N}||\boldsymbol{o}\big(\mathbf{H}_\tau, \mathbf{Q}_{\tau+1}\big)||^2 \overset{a.s.}{=} 0 \tag{22}$$
$$\lim_{N\to\infty} \frac{1}{N}||\boldsymbol{o}\big(\mathbf{H}_{\tau'}, \mathbf{Q}_{\tau'}\big)||^2 \overset{a.s.}{=} 0. \tag{23}$$

The above theorem confirms the intuition that $\mathbf{g}_t$ should be designed as a denoiser. Indeed, since we initialize the MP algorithm with $\mathbf{s}_0 = \mathbf{0}$, from (7) we have that $\mathbf{x} = -\mathbf{q}_0$ and (14) implies

$$\lim_{N\to\infty} \frac{1}{N}\mathbf{h}_t^T\mathbf{x} \overset{a.s.}{=} 0 \tag{24}$$

This, and the fact that $\mathbf{h}_t$ asymptotically behaves as a zero-mean i.i.d. Gaussian vector, suggests to view $\mathbf{r}_t$ as a Gaussian channel and, therefore, to design the function $\mathbf{g}_t$ to be a denoiser.

Additionally, for most MP algorithms, it was shown that there exists a State Evolution (SE) that defines the dynamics of the magnitude of the error propagated in the SMP algorithms.

In particular, for single-memory algorithms such as VAMP, CG-VAMP etc one can establish the mapping

$$v_{h_{t+1}} = SE_{t+1}(v_{h_t}) \tag{25}$$

that defines how the intrinsic uncertainty evolves as the algorithm iterates [8]–[10], [26]. The form of the function SE_{t+1} depends on the chosen SMP algorithm, but is independent of particular realizations of $\mathbf{x}$, $\mathbf{w}$ and $\mathbf{A}$. A similar evolution can be defined for the long-memory algorithms mentioned before, although in these cases the new variance $v_{h_{t+1}}$ will depend on the variances of the whole history of vectors $\mathbf{h}_\tau$ for $\tau \leq t$ and their cross-correlations, i.e.

$$v_{h_{t+1}} = SE_{t+1}(\mathbf{C}_{h_{t+1}}) \tag{26}$$

with $(\mathbf{C}_{h_{t+1}})_{\tau,\tau'} = \lim_{N\to\infty} \frac{1}{N}\mathbf{h}_\tau^T\mathbf{h}_{\tau'}$. The SE provides the means of optimizing the functions $\boldsymbol{f}_t$ and $\mathbf{g}_t$ to obtain the optimal performance of the algorithm and provides a theoretical tool to study the stability and efficiency of SMP algorithms. In particular, the SE was used in [8], [12], [18], [22] to show that AMP, VAMP, CAMP and MAMP can achieve Bayes optimal reconstruction under Assumptions 1-3 given the denoiser $\mathbf{g}_t$ is Bayes optimal, the subsampling factor δ is above a certain threshold, and (except for AMP) conditioned on the validity of the replica prediction for right-orthogonally invariant matrices.

III. Efficient estimation of the divergence in SMP algorithms

In SMP algorithms, the key ingredients ensuring stable, efficient and predictable dynamics are the correction scalars $\{\gamma_t^\tau\}_{\tau=0}^t$ and α_t. For SMP algorithms discussed above, estimating $\{\gamma_t^\tau\}_{\tau=0}^t$ is a well studied problem because the linear function $\boldsymbol{f}_t(\mathbf{S}_{t+1}, \mathbf{y})$ has an explicit dependence on each input vector. This allowed the authors of each algorithm to derive the closed-form solution for every scalar γ_t^τ and use it to form the corresponding estimator. Unfortunately, the same strategy does not work for the denoising step (4) when there is a PnP denoiser $\mathbf{g}_t$. For this case, there is only one available black-box method for estimating the divergence α_t – BB-MC method [1]. However, this method requires a hand-tuning and additional executions of the denoiser in order to estimate α_t. In this section we develop two theoretical models for the divergence α_t within SMP and propose associated estimators that can be computed using only the observed data in the algorithm and do not require additional executions of the denoiser. We begin with an intuition behind the methods and then move to the formal results.

A. Intuition

In the center of the developed techniques is the following parametrized denoiser and its oracle error

$$\hat{\boldsymbol{s}}_{t+1}(\hat{\alpha}) = \mathbf{g}_t(\mathbf{r}_t) - \hat{\alpha}\mathbf{r}_t \tag{27}$$

$$\hat{\boldsymbol{q}}_{t+1}(\hat{\alpha}) = \hat{\boldsymbol{s}}_{t+1}(\hat{\alpha}) - \mathbf{x} \tag{28}$$

where $\hat{\alpha}$ is a scalar parameter. Note that when $\hat{\alpha} = \alpha_t$, (27) is an instance of (4) with the normalization $C_s = 1$. However, by using the fact that $\mathbf{x} = -\mathbf{q}_0$ and expanding the error $\mathbf{q}_{t+1}$ in (14), we find that

$$\begin{aligned}
\lim_{N\to\infty} \frac{1}{N}\mathbf{h}_t^T\mathbf{q}_{t+1} &\overset{a.s.}{=} 0\\
\lim_{N\to\infty} \frac{1}{N}\mathbf{h}_t^T\mathbf{s}_{t+1} - \frac{1}{N}\mathbf{h}_t^T\mathbf{x} &\overset{a.s.}{=} 0\\
\lim_{N\to\infty} \frac{1}{C_s}\frac{1}{N}\mathbf{h}_t^T\Big(\mathbf{g}_t(\mathbf{r}_t) - \alpha_t\mathbf{r}_t\Big) &\overset{a.s.}{=} 0
\end{aligned} \tag{29}$$

where in the last step we used (24). This result implies that the orthogonality result (14) holds for any finite $C_s \neq 0$, including $C_s = 1$. One can verify that when $\hat{\alpha} = \alpha_t$, the error vector $\hat{\boldsymbol{q}}_{t+1}(\alpha_t)$ also follows the other main asymptotic properties of the original vector $\mathbf{q}_{t+1}$ in SMP algorithms, including the fact that $\mathbf{V}^T\hat{\boldsymbol{q}}_{t+1}(\alpha_t)$ acts as a zero-mean i.i.d. Gaussian vector corrupted by an error that almost surely converges to zero.

The idea behind our method is to seek such a function $E(\hat{\alpha})$ that has a root at α_t and we could solve for it. As we just discussed, when $\hat{\alpha} = \alpha_t$, the error $\hat{\boldsymbol{q}}_{t+1}$ is asymptotically orthogonal to $\mathbf{h}_t$. Thus, a straightforward example of such a function would be

$$E(\hat{\alpha}) = \frac{1}{N}\mathbf{h}_t^T\hat{\boldsymbol{q}}_{t+1}(\hat{\alpha}) \tag{30}$$

Then, one could recover α_t by solving $E(\hat{\alpha}) = 0$. Unfortunately, this example of $E(\hat{\alpha})$ cannot be implemented in practice since it is explicitly formulated in terms of the error vectors that are not available. In this work, we use the observed quantities in the algorithm to construct two types of practical functions $E(\hat{\alpha})$ that equated to zero can produce an estimate $\tilde{\alpha}_t$ such that

$$\lim_{N\to\infty} \tilde{\alpha}_t \overset{a.s.}{=} \lim_{N\to\infty} \alpha_t \tag{31}$$

Next, we can adapt Corollary 2 from [24] to our form of SMP algorithms (3)-(4) to show that using such an estimate in SMP algorithms preserves the properties stated in Theorem 1

Lemma 1. *c.f. Corollary 2 [24]: Let Assumptions 1-3 hold. Consider an SMP algorithm* (3)-(4) *but where at every iteration t, the divergence α_t is replaced by a scalar $\tilde{\alpha}_t$ that satisfies* (31)*. Define these iterations as*

$$\tilde{\boldsymbol{r}}_t = \frac{1}{\tilde{C}_r}\Big(\mathbf{A}^T\boldsymbol{f}_t(\tilde{\boldsymbol{S}}_{t+1}, \mathbf{y}) - \tilde{\boldsymbol{S}}_{t+1}\boldsymbol{\gamma}_t\Big) \tag{32}$$

$$\tilde{\boldsymbol{s}}_{t+1} = \frac{1}{\tilde{C}_s}\Big(\mathbf{g}_t(\tilde{\boldsymbol{r}}_t) - \tilde{\boldsymbol{r}}_t\tilde{\alpha}_t\Big) \tag{33}$$

where the rest of the components are the same as in (3)-(4)*. Then, the results* (14)-(17) *from Theorem 1 hold when $\mathbf{h}_\tau$ and $\mathbf{q}_{\tau'}$ are replaced by the error vectors*

$$\tilde{\boldsymbol{h}}_t = \tilde{\boldsymbol{r}}_t - \mathbf{x}$$

$$\tilde{\mathbf{q}}_t = \tilde{\boldsymbol{s}}_t - \mathbf{x}$$

respectively.

Proof. See Appendix D. ∎

This lemma suggests that the main asymptotic properties of an SMP algorithm (3)-(4) are preserved when α_t is replaced by an estimate $\tilde{\alpha}_t$ that asymptotically converges to α_t at every iteration t. Thus, in the following we can focus only on designing a divergence estimator and showing that it is asymptotically consistent under the assumption that Theorem 1 holds up to iteration t.

B. Algebraic divergence estimator

The first class of estimators we propose is a practical extension of the naive and unavailable estimator (30). Its based on substituting the definitions $\hat{\boldsymbol{q}}_{t+1} = \hat{\boldsymbol{s}}_{t+1}(\hat{\alpha}) - \mathbf{x}$ and $\mathbf{h}_t = \mathbf{r}_t - \mathbf{x}$ into (30) and noting that

$$\lim_{N\to\infty} \frac{1}{N}(\mathbf{r}_t - \mathbf{x})^T(\hat{\boldsymbol{s}}_{t+1}(\hat{\alpha}) - \mathbf{x}) \overset{a.s.}{=} \lim_{N\to\infty} \frac{1}{N}(\mathbf{r}_t - \mathbf{x})^T \hat{\boldsymbol{s}}_{t+1}(\hat{\alpha})$$

where we used (24) to obtain $\lim_{N\to\infty} \frac{1}{N}(\mathbf{r}_t - \mathbf{x})^T\mathbf{x} \overset{a.s.}{=} 0$. Still, the above equation involves $\mathbf{x}$ explicitly, which can be resolved by considering the difference $\mathbf{r}_t - \mathbf{r}_{t-1}$ instead of $\mathbf{r}_t$ alone

$$\begin{aligned}
&\lim_{N\to\infty} \frac{1}{N}(\mathbf{r}_t - \mathbf{r}_{t-1})^T(\hat{\boldsymbol{s}}_{t+1}(\hat{\alpha}) - \mathbf{x}) \\
&= \lim_{N\to\infty} \frac{1}{N}(\mathbf{h}_t + \mathbf{x} - \mathbf{h}_{t-1} - \mathbf{x})^T(\hat{\boldsymbol{s}}_{t+1}(\hat{\alpha}) - \mathbf{x}) \\
&= \lim_{N\to\infty} \frac{1}{N}(\mathbf{h}_t - \mathbf{h}_{t-1})^T(\hat{\boldsymbol{s}}_{t+1}(\hat{\alpha}) - \mathbf{x}) \\
&\overset{a.s.}{=} \lim_{N\to\infty} \frac{1}{N}(\mathbf{h}_t - \mathbf{h}_{t-1})^T \hat{\boldsymbol{s}}_{t+1}(\hat{\alpha}) \qquad (34)
\end{aligned}$$

where, again, we used (24) to obtain $\lim_{N\to\infty} \frac{1}{N}(\mathbf{h}_t - \mathbf{h}_{t-1})^T\mathbf{x} \overset{a.s.}{=} 0$. This result suggests that if we define a scalar function

$$E_{t+1}(\hat{\alpha}) = (\mathbf{r}_t - \mathbf{r}_{t-1})^T \hat{\boldsymbol{s}}_{t+1}(\hat{\alpha}) \qquad (35)$$

and equate it to zero, then we can recover $\hat{\alpha}$ such that the orthogonality between $\hat{\boldsymbol{q}}_{t+1}(\hat{\alpha})$ and $\mathbf{h}_t$ and $\mathbf{h}_{t-1}$ is insured. The following theorem summarizes and generalizes this idea.

Theorem 2. *Consider an SMP algorithm following* (3)-(4). *Define a vector*

$$\overline{\mathbf{r}}_t = \sum_{\tau=0}^{t-1} k_\tau^t \mathbf{r}_\tau \qquad (36)$$

with scalar weights $\sum_{\tau=0}^{t-1} k_\tau^t = 1$. *Then, under Assumptions 1-3, the inner-product* $(\mathbf{r}_t - \overline{\mathbf{r}}_t)^T\mathbf{r}_t$ *is asymptotically non-zero and*

$$\hat{\alpha}_t = \frac{(\mathbf{r}_t - \overline{\mathbf{r}}_t)^T \mathbf{g}_t(\mathbf{r}_t)}{(\mathbf{r}_t - \overline{\mathbf{r}}_t)^T\mathbf{r}_t} \qquad (37)$$

almost surely converges to the divergence α_t *of the denoiser* $\mathbf{g}_t(\mathbf{r}_t)$,

$$\lim_{N\to\infty} \hat{\alpha}_t \overset{a.s.}{=} \lim_{N\to\infty} \alpha_t. \qquad (38)$$

Proof. First, due to the normalization of the weights k_τ^t, we have that

$$\overline{\mathbf{r}}_t - \mathbf{x} = \sum_{\tau=0}^{t-1} k_\tau^t \mathbf{r}_\tau - \sum_{\tau=0}^{t-1} k_\tau^t \mathbf{x} = \sum_{\tau=0}^{t-1} k_\tau^t \mathbf{h}_\tau =: \overline{\boldsymbol{h}}_t \qquad (39)$$

where we defined a weighted error vector $\overline{\boldsymbol{h}}_t$. From (18) we have that this error vector is equivalent to

$$\begin{aligned}
\overline{\boldsymbol{h}}_t &= \sum_{\tau=0}^{t-1} k_\tau^t \breve{\boldsymbol{h}}_\tau + \sum_{\tau=0}^{t-1} k_\tau^t \boldsymbol{o}\big(\mathbf{H}_\tau, \mathbf{Q}_{\tau+1}\big) \qquad (40) \\
&=: \sum_{\tau=0}^{t-1} k_\tau^t \breve{\boldsymbol{h}}_\tau + \overline{\boldsymbol{o}}\big(\mathbf{H}_t, \mathbf{Q}_{t+1}\big) \qquad (41)
\end{aligned}$$

where we defined a vector

$$\overline{\boldsymbol{o}}\big(\mathbf{H}_t, \mathbf{Q}_{t+1}\big) := \sum_{\tau=0}^{t-1} k_\tau^t \boldsymbol{o}\big(\mathbf{H}_\tau, \mathbf{Q}_{\tau+1}\big) \qquad (42)$$

Note that because each $\breve{\boldsymbol{h}}_\tau$ is Gaussian, the first sum in (40) is Gaussian as well, while the second sum represents a vector whose magnitude almost surely converges to zero as follows from (22). Then, because of the assumption about $\mathbf{g}_t$ being uniformly Lipschitz continuous, in the limit we have that

$$\begin{aligned}
&\lim_{N\to\infty} \frac{1}{N}||\mathbf{g}_t(\mathbf{x} + \mathbf{h}_t) - \mathbf{g}_t(\mathbf{x} + \breve{\boldsymbol{h}}_t)||^2 \\
&= \lim_{N\to\infty} \frac{1}{N}||\mathbf{g}_t\Big(\mathbf{x} + \breve{\boldsymbol{h}}_t + \overline{\boldsymbol{o}}\big(\mathbf{H}_t, \mathbf{Q}_{t+1}\big)\Big) - \mathbf{g}_t(\mathbf{x} + \breve{\boldsymbol{h}}_t)||^2 \\
&\leq \lim_{N\to\infty} \frac{1}{N}\left|\left|\overline{\boldsymbol{o}}\big(\mathbf{H}_t, \mathbf{Q}_{t+1}\big)\right|\right|^2 \overset{a.s.}{=} 0 \qquad (43)
\end{aligned}$$

Then, since $\mathbf{r}_t - \overline{\mathbf{r}}_t = \mathbf{h}_t - \overline{\boldsymbol{h}}_t$ as was seen in (34), we can use (13) to obtain

$$\begin{aligned}
&\lim_{N\to\infty} \frac{1}{N}(\mathbf{r}_t - \overline{\mathbf{r}}_t)^T \mathbf{g}_t(\mathbf{r}_t) = \lim_{N\to\infty} \frac{1}{N}(\mathbf{h}_t - \overline{\boldsymbol{h}}_t)^T \mathbf{g}_t(\mathbf{x} + \mathbf{h}_t) \\
&\overset{a.s.}{=} \lim_{N\to\infty} \frac{1}{N}\left(\breve{\boldsymbol{h}}_t - \sum_{\tau=0}^{t-1} k_\tau^t \breve{\boldsymbol{h}}_\tau\right)^T \mathbf{g}_t(\mathbf{x} + \breve{\boldsymbol{h}}_t) \\
&\overset{a.s.}{=} \lim_{N\to\infty} \frac{1}{N}\left(\breve{\boldsymbol{h}}_t - \sum_{\tau=0}^{t-1} k_\tau^t \breve{\boldsymbol{h}}_\tau\right)^T \breve{\boldsymbol{h}}_t \alpha_t \qquad (44) \\
&\overset{a.s.}{=} \lim_{N\to\infty} \frac{1}{N}(\mathbf{h}_t - \overline{\boldsymbol{h}}_t)^T \mathbf{h}_t \alpha_t \qquad (45)
\end{aligned}$$

Here, the step (44) follows from the fact that for each $\tau = 0, ..., t$ we have

$$\lim_{N\to\infty} \frac{1}{N} \breve{\boldsymbol{h}}_\tau^T \mathbf{g}_t(\mathbf{x} + \breve{\boldsymbol{h}}_t) \overset{a.s.}{=} \lim_{N\to\infty} \frac{1}{N} \breve{\boldsymbol{h}}_\tau^T \breve{\boldsymbol{h}}_t \alpha_t \qquad (46)$$

since $\breve{\boldsymbol{h}}_\tau$ for $\tau = 0, ..., t$ is i.i.d. Gaussian, and due to the Stein's identity (13). Next, we can use (24) to show

$$\begin{aligned}
\lim_{N\to\infty} \frac{1}{N}(\mathbf{r}_t - \overline{\mathbf{r}}_t)^T \mathbf{r}_t &= \lim_{N\to\infty} \frac{1}{N}(\mathbf{h}_t - \overline{\boldsymbol{h}}_t)^T(\mathbf{x} + \mathbf{h}_t) \\
&\overset{a.s.}{=} \lim_{N\to\infty} \frac{1}{N}(\mathbf{h}_t - \overline{\boldsymbol{h}}_t)^T \mathbf{h}_t \qquad (47)
\end{aligned}$$

Combining the results (45) and (47) gives

$$\begin{aligned}
&\lim_{N\to\infty} \frac{\frac{1}{N}(\mathbf{r}_t - \overline{\mathbf{r}}_t)^T \mathbf{g}_t(\mathbf{r}_t)}{\frac{1}{N}(\mathbf{r}_t - \overline{\mathbf{r}}_t)^T \mathbf{r}_t} \\
&\overset{a.s.}{=} \lim_{N\to\infty} \frac{\frac{1}{N}(\mathbf{h}_t - \overline{\boldsymbol{h}}_t)^T \mathbf{h}_t \alpha_t}{\frac{1}{N}(\mathbf{h}_t - \overline{\boldsymbol{h}}_t)^T \mathbf{h}_t} = \alpha_t \qquad (48)
\end{aligned}$$

Lastly, from Theorem 1 we know that in the limit, the matrix $\mathbf{H}_{t+1} = (\mathbf{h}_0, ..., \mathbf{h}_t)$ is full rank so the inner-product

$$\frac{1}{N}(\mathbf{h}_t - \overline{\boldsymbol{h}}_t)^T \mathbf{h}_t = \frac{1}{N}\left(\mathbf{h}_t - \sum_{\tau=0}^{t-1} k_\tau^t \mathbf{h}_\tau\right)^T \mathbf{h}_t \qquad (49)$$

is non zero almost surely. Again, noting that $\lim_{N\to\infty} \frac{1}{N}(\mathbf{r}_t - \overline{\mathbf{r}}_t)^T\mathbf{r}_t \overset{a.s.}{=} \lim_{N\to\infty} \frac{1}{N}(\mathbf{h}_t - \overline{\boldsymbol{h}}_t)^T\mathbf{h}_t$ completes the proof. ■

In the following, we refer to the estimator based on (37) as an *algebraic estimator*. By equating (35) to zero and solving for $\hat{\alpha}$, one can show that the function E_{t+1} leads to the algebraic estimator with $\overline{\mathbf{r}}_t = \mathbf{r}_{t-1}$. While in the limit (37)

holds for any set of weights k_τ^t as long as the normalization is satisfied, in the finite dimensional case the asymptotic identities used to derive Theorem 2 are no longer exact and an additional error emerges. This error might be substantial in the case if, for example, we use $\overline{\mathbf{r}}_t = \mathbf{r}_{t-1}$. In this case, the term $\frac{1}{N}(\mathbf{h}_t - \overline{\boldsymbol{h}}_t)^T\mathbf{x}$ (assumed to be equal to zero in (47)) might have considerable magnitude due to the fact that the magnitude of $\mathbf{x}$ remains the same throughout the algorithm and might significantly exceed the magnitude of $\mathbf{h}_t$ and of $\overline{\boldsymbol{h}}_t$. Then, any small alignment of these error vectors with $\mathbf{x}$ would result in a substantial quantity that affects the accuracy of the LSL approximation (47).

On the other hand, the finite dimensional model of $\mathbf{h}_t$ deviates from the asymptotic one and these deviations accumulate as the algorithm progresses. One of the effects of this error is that the asymptotic identity

$$\lim_{N\to\infty}\frac{1}{N}\mathbf{h}_\tau^T\mathbf{g}_t(\mathbf{r}_t) \overset{a.s.}{=} \lim_{N\to\infty}\alpha_t\frac{1}{N}\mathbf{h}_\tau^T\mathbf{h}_t \tag{50}$$

used to prove Theorem 2 becomes less accurate for finite N as the difference between t and τ increases. For this reason we might observe poor quality of the divergence estimates if we use $\overline{\mathbf{r}}_t = \mathbf{r}_0$. The detailed analysis of the optimal choice of weights k_τ^t is left for further study, while in this work we consider the cases $\overline{\mathbf{r}}_t = \mathbf{r}_{t-1}$ and $\overline{\mathbf{r}}_t = \mathbf{r}_0$. The important advantage of these two options is that the computational cost of the resulting algebraic estimator is dominated by the cost of two inner-products of N-dimensional vectors. Such a low cost allows one to efficiently tune a parametrized denoiser using the SURE technique [25] to optimize the performance of the denoising block. In particular, let the denoiser $\mathbf{g}_t(\mathbf{r}_t, \boldsymbol{\theta})$ be dependent on some free parameter vector $\boldsymbol{\theta}$, which, in the context of BM3D denoiser, could be the patch size, window size, distance between patches etc. Then, one could optimize $\boldsymbol{\theta}$ with respect to the estimate $\hat{v}(\boldsymbol{\theta})$ of the MSE $v(\boldsymbol{\theta}) = \frac{1}{N}||\mathbf{g}_t(\mathbf{r}_t, \boldsymbol{\theta}) - \mathbf{x}||^2$ as [1]

$$\begin{aligned}\boldsymbol{\theta}_{opt} &= \underset{\boldsymbol{\theta}}{\operatorname{argmin}}\ \hat{v}(\theta) \\ &= \underset{\boldsymbol{\theta}}{\operatorname{argmin}}\ \frac{1}{N}||\mathbf{r}_t - \mathbf{g}_t(\mathbf{r}_t, \boldsymbol{\theta})||^2 - v_{h_t} + 2v_{h_t}\hat{\alpha}_t(\mathbf{r}_t, \boldsymbol{\theta})\end{aligned} \tag{51}$$

where $\hat{\alpha}_t(\mathbf{r}_t, \boldsymbol{\theta})$ is an estimate of the divergence $\alpha_t(\mathbf{r}_t, \boldsymbol{\theta})$ of $\mathbf{g}_t(\mathbf{r}_t, \boldsymbol{\theta})$. Then, using the algebraic divergence estimator makes the resulting cost of evaluating (51) negligible compared to executing most of plug-and-play denoisers.

Yet, as it will be demonstrated in the simulation section, these two special cases of the algebraic estimator are sensitive to finiteness of N and M, and demonstrate satisfactory accuracy only for inverse problems of dimension of order 10^6 and larger. While a rough estimate of the divergence might be acceptable for tuning the denoiser, in order to ensure stable performance of an SMP, we require more robust alternatives. In the next section we present such an alternative.

C. Polynomial divergence estimator

In this section we present another way of constructing a practical function $E(\hat{\alpha})$ that has a root almost surely converging to the divergence α_t of the denoiser $\mathbf{g}_t(\mathbf{r}_t)$. For this, we switch the parametrized denoising step $\hat{\boldsymbol{s}}_{t+1}(\hat{\alpha})$ from (27) to the following more general form

$$\overline{\mathbf{s}}_{t+1}(\hat{\alpha}, \tau) = \mathbf{g}_t(\mathbf{r}_t) - \hat{\alpha}(\mathbf{r}_t - \mathbf{s}_\tau) \tag{52}$$

which is associated with the corresponding error vector

$$\overline{\mathbf{q}}_{t+1}(\hat{\alpha}, \tau) = \overline{\mathbf{s}}_{t+1}(\hat{\alpha}, \tau) - \mathbf{x} \tag{53}$$

Note that $\hat{\boldsymbol{s}}_{t+1}(\hat{\alpha})$ is a special case of $\overline{\mathbf{s}}_{t+1}(\hat{\alpha}, \tau)$ when the parameter τ is set to 0. Another important property of (52) is that the asymptotic orthogonality of $\overline{\mathbf{q}}_{t+1}(\hat{\alpha}, \tau)$ and $\mathbf{h}_t$ implies the orthogonality of $\hat{\boldsymbol{q}}_{t+1}(\hat{\alpha})$ and $\mathbf{h}_t$ and vice versa, since

$$\begin{aligned}\lim_{N\to\infty}\frac{1}{N}\mathbf{h}_t^T\overline{\mathbf{q}}_{t+1}(\hat{\alpha}, \tau) &= \lim_{N\to\infty}\frac{1}{N}\mathbf{h}_t^T\big(\hat{\boldsymbol{q}}_{t+1}(\hat{\alpha}) + \hat{\alpha}\mathbf{s}_\tau\big) \\ &\overset{a.s.}{=} \lim_{N\to\infty}\frac{1}{N}\mathbf{h}_t^T\hat{\boldsymbol{q}}_{t+1}(\hat{\alpha})\end{aligned} \tag{54}$$

and this result is invariant with respect to τ. Here we used (14) and the fact that $\mathbf{x} = -\mathbf{q}_0$ to show that

$$\lim_{N\to\infty}\frac{1}{N}\mathbf{h}_t^T\mathbf{s}_\tau = \lim_{N\to\infty}\frac{1}{N}\mathbf{h}_t^T(\mathbf{x} + \mathbf{q}_\tau) \overset{a.s.}{=} 0 \tag{55}$$

Thus, one can equivalently use (52) to derive estimators for α_t, while the more general structure of $\overline{\mathbf{s}}_{t+1}(\hat{\alpha}, \tau)$ provides additional flexibility that will be useful next.

To derive a new estimator of α_t, consider the MSE of the parametrized denoiser $\overline{\mathbf{s}}_{t+1}(\hat{\alpha}, \tau)$ and recall that $\mathbf{r}_t = \mathbf{x} + \mathbf{h}_t$ and $\overline{\mathbf{s}}_{t+1}(\hat{\alpha}, \tau) = \mathbf{x} + \overline{\mathbf{q}}_{t+1}(\hat{\alpha}, \tau)$. Then, after simple algebraic manipulations we obtain

$$\begin{aligned}&\frac{1}{N}||\overline{\mathbf{s}}_{t+1}(\hat{\alpha}, \tau) - \mathbf{x}||^2 = \frac{1}{N}||\overline{\mathbf{s}}_{t+1}(\hat{\alpha}, \tau) - \mathbf{x} - \mathbf{h}_t + \mathbf{h}_t||^2 \\ &= \frac{1}{N}||\overline{\mathbf{s}}_{t+1}(\hat{\alpha}, \tau) - \mathbf{r}_t + \mathbf{h}_t||^2 \\ &= \frac{1}{N}\Big(||\overline{\mathbf{s}}_{t+1}(\hat{\alpha}, \tau) - \mathbf{r}_t||^2 + 2\mathbf{h}_t^T(\overline{\mathbf{s}}_{t+1}(\hat{\alpha}, \tau) - \mathbf{r}_t) + ||\mathbf{h}_t||^2\Big) \\ &= \frac{1}{N}\Big(||\overline{\mathbf{s}}_{t+1}(\hat{\alpha}, \tau) - \mathbf{r}_t||^2 + 2\mathbf{h}_t^T(\overline{\mathbf{q}}_{t+1}(\hat{\alpha}, \tau) - \mathbf{h}_t) + ||\mathbf{h}_t||^2\Big) \\ &= \frac{1}{N}\Big(||\overline{\mathbf{s}}_{t+1}(\hat{\alpha}, \tau) - \mathbf{r}_t||^2 - ||\mathbf{h}_t||^2 + 2\mathbf{h}_t^T\overline{\mathbf{q}}_{t+1}(\hat{\alpha}, \tau)\Big)\end{aligned}$$

Then, if we define a function

$$J_{t+1}^1(\hat{\alpha}, \tau) = \frac{1}{N}\Big(||\overline{\mathbf{s}}_{t+1}(\hat{\alpha}, \tau) - \mathbf{r}_t||^2 - ||\mathbf{h}_t||^2\Big), \tag{56}$$

this function would be equivalent to the MSE of parametrized intrinsic measurement $\overline{\mathbf{s}}_{t+1}(\hat{\alpha}, \tau)$ plus an error

$$J_{t+1}^1(\hat{\alpha}, \tau) = \frac{1}{N}||\overline{\mathbf{s}}_{t+1}(\hat{\alpha}, \tau) - \mathbf{x}||^2 + e_{t+1}^1(\hat{\alpha}) \tag{57}$$

where

$$e_{t+1}^1(\hat{\alpha}) = -2\mathbf{h}_t^T\overline{\mathbf{q}}_{t+1}(\hat{\alpha}, \tau) \tag{58}$$

As was discussed before, when $\hat{\alpha} = \alpha_t$, the error vector $\overline{\mathbf{q}}_{t+1}(\hat{\alpha}, \tau)$ is asymptotically orthogonal to $\mathbf{h}_t$. This implies that in the limit $J_{t+1}^1(\hat{\boldsymbol{s}}_{t+1}(\alpha_t))$ converges to the MSE of $\overline{\mathbf{s}}_{t+1}(\alpha_t, \tau)$ when $\hat{\alpha} = \alpha_t$, and is additionally corrupted by the error $e_{t+1}^1(\hat{\alpha})$ when $\hat{\alpha} \neq \alpha_t$.

At the same time, the same MSE can be observed in a different way. Using the definition of the vector $\mathbf{y}$ and that $\overline{\mathbf{s}}_{t+1}(\hat{\alpha}, \tau) = \mathbf{x} + \overline{\mathbf{q}}_{t+1}(\hat{\alpha}, \tau)$, we can show that

$$\lim_{N\to\infty}\frac{1}{N}||\mathbf{y}-\mathbf{A}\bar{\mathbf{s}}_{t+1}(\hat{\alpha},\tau)||^2$$
$$=\lim_{N\to\infty}\frac{1}{N}||\mathbf{w}-\mathbf{A}\bar{\mathbf{q}}_{t+1}(\hat{\alpha},\tau)||^2$$
$$\overset{a.s.}{=}\delta v_w+\lim_{N\to\infty}\frac{1}{N}||\mathbf{A}\bar{\mathbf{q}}_{t+1}(\hat{\alpha},\tau)||^2-\frac{2}{N}\mathbf{w}^T\mathbf{A}\bar{\mathbf{q}}_{t+1}(\hat{\alpha},\tau)$$

Next, we can use the conditioning technique [8], [17], [18] for the random matrix $\mathbf{A}$ to study the interaction between $\bar{\mathbf{q}}_{t+1}(\hat{\alpha},\tau)$ and $\mathbf{A}$. In Appendix B we show that

$$\lim_{N\to\infty}\frac{1}{N}||\mathbf{A}\bar{\mathbf{q}}_{t+1}(\hat{\alpha},\tau)||^2\overset{a.s.}{=}\lim_{N\to\infty}\frac{1}{N}||\bar{\mathbf{q}}_{t+1}(\hat{\alpha},\tau)||^2+\zeta_{t+1}(\hat{\alpha})$$

where $\zeta_{t+1}(\hat{\alpha})$ depends on the whole history of vectors $(\mathbf{q}_0,...,\mathbf{q}_t)$ when $\hat{\alpha}\neq\alpha_t$ and almost surely converges to zero for $\hat{\alpha}=\alpha_t$. Similarly, in Appendix B we show that $\frac{1}{N}\mathbf{w}^T\mathbf{A}\bar{\mathbf{q}}_{t+1}(\hat{\alpha},\tau)$ almost surely converges to $v_w(\alpha_t-\hat{\alpha})$, which becomes zero for $\hat{\alpha}=\alpha_t$. Therefore one can define another MSE estimator

$$J^2_{t+1}(\hat{\alpha},\tau)=\frac{1}{N}||\mathbf{y}-\mathbf{A}\bar{\mathbf{s}}_{t+1}(\hat{\alpha},\tau)||^2-\delta v_w \tag{59}$$

which can be represented as

$$J^2_{t+1}(\hat{\alpha},\tau)=\frac{1}{N}||\bar{\mathbf{q}}_{t+1}(\hat{\alpha},\tau)||^2+e^2_{t+1}(\hat{\alpha}) \tag{60}$$

where $e^2_{t+1}(\hat{\alpha})$ almost surely converges to zero for $\hat{\alpha}=\alpha_t$.

The important observation, which we theoretically confirm in Appendix B, about $J^1_{t+1}(\hat{\alpha},\tau)$ and $J^2_{t+1}(\hat{\alpha},\tau)$ is that their errors, $e^1_{t+1}(\hat{\alpha})$ and $e^2_{t+1}(\hat{\alpha})$, behave differently for $\hat{\alpha}\neq\alpha_t$ and both almost surely converge to zero for $\hat{\alpha}=\alpha_t$. Then, by defining a new function

$$E_{t+1}(\hat{\alpha},\tau)=J^1_{t+1}(\hat{\alpha},\tau)-J^2_{t+1}(\hat{\alpha},\tau) \tag{61}$$

one could recover α_t by finding the appropriate root to $E_{t+1}(\hat{\alpha},\tau)$. The following theorem shows that (61) corresponds to a particular quadratic polynomial

Lemma 2. *Consider an SMP algorithm following* (3)-(4) *and let* $v_{h_t}=\frac{1}{N}||\mathbf{h}_t||^2$. *Then* $E_{t+1}(\hat{\alpha},\tau)$ *from* (61) *is the following quadratic polynomial*

$$E_{t+1}(\hat{\alpha},\tau)=u_0+u_1(\tau)\hat{\alpha}+u_2(\tau)\hat{\alpha}^2 \tag{62}$$

where the scalar coefficients are defined as

$$u_0=\frac{1}{N}\Big(||\mathbf{r}_t-\mathbf{g}_t||^2-v_{h_t}-||\mathbf{y}-\mathbf{A}\mathbf{g}_t||^2+\delta v_w\Big)$$
$$u_1(\tau)=\frac{2}{N}(\mathbf{r}_t-\mathbf{s}_\tau)^T\big(\mathbf{r}_t-\mathbf{g}_t-\mathbf{A}^T(\mathbf{y}-\mathbf{A}\mathbf{g}_t)\big)$$
$$u_2(\tau)=\frac{1}{N}\big(||\mathbf{r}_t-\mathbf{s}_\tau||^2-||\mathbf{A}(\mathbf{r}_t-\mathbf{s}_\tau)||^2\big)$$

with $\mathbf{g}_t$ *used as a shorthand for* $\mathbf{g}_t(\mathbf{r}_t)$.

Proof. See Appendix A. ■

Note that the coefficients of the equation (62) are formed only from the data that is naturally circulated in any SMP algorithm. Computational cost of evaluating u_0, u_1 and u_2 is dominated by implementing two matrix-vector products $\mathbf{A}\mathbf{g}_t(\mathbf{r}_t)$ and $\mathbf{A}\mathbf{r}_t$. However, this cost can be reduced by reusing the calculations to form the next update. All the algorithms mentioned at the end of section II.A compute the vector

$$\mathbf{z}_{t+1}=\mathbf{y}-\mathbf{A}\mathbf{s}_{t+1} \tag{63}$$

as part of the function $\boldsymbol{f}_t$ in (3). Using the definition of $\mathbf{s}_{t+1}$, this vector can be equivalently represented as

$$\mathbf{z}_{t+1}=\mathbf{y}-\mathbf{A}\mathbf{s}_{t+1}=\mathbf{y}-\mathbf{A}\mathbf{g}_t(\mathbf{r}_t)-\alpha_t\mathbf{A}\mathbf{r}_t$$

Thus, one can reuse the results of the matrix-vector products $\mathbf{A}\mathbf{g}_t(\mathbf{r}_t)$ and $\mathbf{A}\mathbf{r}_t$ to update $\mathbf{z}_{t+1}$ and, consequently, reduce the number of additional matrix-vector products down to 1. Additionally, one can store the m-dimensional vector $\mathbf{A}\mathbf{s}_\tau$ to reuse this result in the implementation of u_2 and u_3.

Lemma 2 becomes useful in the light of the following theorem that establishes the asymptotic behaviour of the roots of $E_{t+1}(\hat{\alpha},\tau)$

Theorem 3. *Consider an SMP algorithm following* (3)-(4). *Let* $\bar{\mathbf{s}}_{t+1}(\hat{\alpha},\tau)$ *be defined as in* (52) *with* $\tau\leq t$. *Additionally, let* $\mathbf{Q}_{t+1}=(\mathbf{q}_0,...,\mathbf{q}_t)$ *and define a vector*

$$\boldsymbol{\beta}_{t+1}=\lim_{N\to\infty}\mathbf{Q}^\dagger_{t+1}\big(\mathbf{g}_t(\mathbf{r}_t)-\mathbf{x}\big) \tag{64}$$

Then, under Assumptions 1-3, the function $E_{t+1}(\hat{\alpha},\tau)$ *from* (61) *has two roots* $\hat{\alpha}_1$ *and* $\hat{\alpha}_2$ *that satisfy*

$$\lim_{N\to\infty}\hat{\alpha}_1\overset{a.s.}{=}\alpha_t \tag{65}$$

$$\lim_{N\to\infty}\hat{\alpha}_2(\tau)\overset{a.s.}{=}\lim_{N\to\infty}\frac{c^0_t+c^2_t\alpha_t}{c^2_t-c^1_t(\tau)} \tag{66}$$

where

$$c^0_t=-2\big(v_{h_t}-v_w+\frac{1}{N}\boldsymbol{\beta}^T_{t+1}\mathbf{Q}^T_{t+1}\mathbf{A}^T\mathbf{A}\mathbf{h}_t\big) \tag{67}$$

$$c^1_t(\tau)=-2\frac{1}{N}\mathbf{q}^T_\tau\mathbf{A}^T\mathbf{A}\mathbf{h}_t \tag{68}$$

$$c^2_t=v_{h_t}-\frac{1}{N}||\mathbf{A}\mathbf{h}_t||^2 \tag{69}$$

Proof. See Appendix B. ■

Then, one way to estimate α_t is by computing the roots to (62) and identifying which of the two roots is the correct one. In the following, we refer to this estimator as a *polynomial estimator*. As it will be demonstrated in the simulation section, SMP algorithms with the polynomial estimator demonstrate stable dynamics similar to the BB-MC estimator even for N and M of order 10^4. To combine the advantages of the algebraic and the polynomial estimators, one could use the algebraic estimator to tune the denoiser via SURE and use the polynomial estimator to compute the final correction scalar α_t. This approach combines the advantages of both methods and results in a fast and efficient way of updating $\mathbf{s}_{t+1}$.

D. Root identification for the polynomial estimator

While Theorem 3 relates the correction scalar α_t to one of the roots $\hat{\alpha}_1$ and $\hat{\alpha}_2$ of (62), it is still required to identify which of the two roots is the right one. In this subsection, we propose a method for assigning α_t to either $\hat{\alpha}_1$ or $\hat{\alpha}_2$.

The idea is based on forming a pair of polynomials, $P_1(\hat{\alpha})$ and $P_2(\hat{\alpha})$, that share one root at α_t and the other two roots would be different by a substantial amount. Let the pair $\hat{\alpha}_1$ and $\hat{\alpha}_2$ and the pair $\hat{\alpha}_3$ and $\hat{\alpha}_4$ be the roots of $P_1(\hat{\alpha})$ and $P_2(\hat{\alpha})$ respectively. Additionally, assume that two roots from

different pairs are the same. Then, one way to form an estimate $\hat{\alpha}_t$ of the common root α_t would be

$$\begin{aligned}\hat{\alpha}_t &= \frac{\hat{\alpha}_{k^*} + \hat{\alpha}_{s^*}}{2} \\ (k^*, s^*) &= \underset{k\in\{1,2\}, s\in\{3,4\}}{\operatorname{argmin}} |\hat{\alpha}_k - \hat{\alpha}_s|,\end{aligned} \tag{70}$$

i.e. take the average of two roots that are from two different polynomials and that are the closest ones.

In our context, the polynomial $E_{t+1}(\hat{\alpha}, \tau)$ from (61) is a function of τ. From (65) we know that the first root, $\hat{\alpha}_1$, is invariant with respect to τ, but its not the case for the second root $\hat{\alpha}_2$. Given that there is a pair of indices $\tau \neq \tau'$ such that $\tau, \tau' \leq t$ and satisfies

$$\lim_{N\to\infty} \frac{1}{N} \mathbf{q}_\tau^T \mathbf{A}^T \mathbf{A} \mathbf{h}_t \neq \lim_{N\to\infty} \frac{1}{N} \mathbf{q}_{\tau'}^T \mathbf{A}^T \mathbf{A} \mathbf{h}_t \tag{71}$$

we can generate a pair of polynomials $E_{t+1}(\hat{\alpha}, \tau)$ and $E_{t+1}(\hat{\alpha}, \tau')$ that share one root at α_t and have distinct second roots. While identifying when the condition (71) holds for a general SMP framework (3)-(4) is a challenging theoretical task that we leave for further work, we can test this condition online. In Appendix D we show that finding a pair of indices (τ, τ') that follows (71) is asymptotically equivalent to finding the pair that ensures

$$\lim_{N\to\infty} \frac{1}{N} \mathbf{z}_\tau^T (\mathbf{y} - \mathbf{A}\mathbf{r}_t) \neq \lim_{N\to\infty} \frac{1}{N} \mathbf{z}_{\tau'}^T (\mathbf{y} - \mathbf{A}\mathbf{r}_t) \tag{72}$$

Note that all the elements involved in this condition are available so one can test the condition at a negligible computational cost.

Motivated by the above idea, we make the following proposal

Proposal 1. *Consider an SMP algorithm following* (3)-(4)*. Let $\tau, \tau' \leq t$ be a pair of indices that follow the condition* (71)*. Generate the pair of polynomials $E_{t+1}(\hat{\alpha}, \tau)$ and $E_{t+1}(\hat{\alpha}, \tau')$ associated with the pair (τ, τ'). Let $\hat{\alpha}_1$ and $\hat{\alpha}_2$ define the roots of the first polynomials and $\hat{\alpha}_3$ and $\hat{\alpha}_4$ define the roots of the second polynomial, respectively. Then, choose an estimate $\hat{\alpha}_t$ of the divergence α_t of the denoiser $\mathbf{g}_t(\mathbf{r}_t)$ as in* (70)*.*

E. Implementation details of the polynomial estimator

As discussed in Section III.B, when we implement SMP algorithms in practice, the finite dimensional model deviates from the asymptotic one, which results in the emergence of additional stochastic components in the algorithm. From our experiments, we observed that sometimes after a few iterations, the polynomial constructed from (62) might end up having complex roots. However, since we assume $\mathbf{g}_t$ is a real-valued function, the divergence α_t must be a real value. Therefore, when the roots $\hat{\alpha}_1$ and $\hat{\alpha}_2$ of the polynomial $E_{t+1}(\hat{\alpha}, \tau)$ associated with the index τ are complex, we set them to the stationary point of the quadratic function

$$\hat{\alpha}_1(\tau) = \hat{\alpha}_2(\tau) = -\frac{u_2(\tau)}{2u_3(\tau)} \tag{73}$$

where $u_2(\tau)$ and $u_3(\tau)$ are as in (62). The same we do with the roots $\hat{\alpha}_3$ and $\hat{\alpha}_4$ of the second polynomial $E_{t+1}(\hat{\alpha}, \tau')$ associated with the index τ', if these roots are complex. Next, regardless whether all the roots were originally real or not, we proceed to the rule (70) to form the finale estimate $\hat{\alpha}_t$.

IV. Simulation experiments

In this section we compare the proposed divergence estimators against the BB-MC method [1] within AMP [6], VAMP [8], CG-VAMP [10] and WS-CG-VAMP [10] where the denoiser is chosen to be BM3D[2]. We consider the problem of recovering natural images shown on Figure 1 from the measurement system (1), where the subsampling ratio is chosen to be $\delta = \frac{M}{N} = 0.05$ and we set the measurement noise variance v_w to achieve SNR $\frac{||\mathbf{x}||^2}{||\mathbf{w}||^2}$ of $40dB$. Additionally, in the experiments with all the algorithms, except for AMP, we set the condition number $\kappa(\mathbf{A}) = 1000$, unless stated otherwise. Furthermore, we choose $\mathbf{A} = \mathbf{SPHD}$ to be the Fast ill-conditioned Johnson-Lindenstrauss (FIJL) transform, which is composed of the following matrices [15]: the values of the diagonal matrix $\mathbf{D}$ are either -1 or 1 with equal probability; the matrix $\mathbf{H}$ is some fast orthogonal transform. In our simulations, we chose $\mathbf{H}$ to be the Discrete Cosine Transform (DCT); The matrix $\mathbf{P}$ is a random permutation matrix and the matrix $\mathbf{S}$ is an M by N matrix of zeros except for the main diagonal, where the singular values follow geometric progression leading to the desired condition number. Although the FIJL operator is rather artificial, it is convenient for evaluating the performance of algorithms since it acts as a prototypical ill-conditioned Compressed Sensing matrix, requires no storing of matrices and has a fast implementation. Additionally, the FIJL operator that we consider enables us to directly implement VAMP, since

$$\mathbf{A}\mathbf{A}^T = \mathbf{SPHDD}^T\mathbf{H}^T\mathbf{P}^T\mathbf{S}^T = \mathbf{S}\mathbf{S}^T \tag{74}$$

and, therefore, the matrix inverse

$$\mathbf{W}_t^{-1} = (v_w \mathbf{I}_M + v_{q_t} \mathbf{A}\mathbf{A}^T)^{-1} = (v_w \mathbf{I}_M + v_{q_t} \mathbf{S}\mathbf{S}^T)^{-1}$$

requires inverting only a diagonal matrix. However, here we emphasize that the AMP, CG-VAMP and WS-CG-VAMP algorithms that we implement do not utilize the fact that $\mathbf{W}_t$ is diagonal and operate as if $\mathbf{W}$ is an arbitrary matrix.

To the best of our knowledge, there is no general practice for tuning the BB-MC divergence estimator for denoisers that violate the continuity assumption like in the case of the BM3D denoiser. In this work we use the heuristic for choosing the scalar ϵ from (2) as in the GAMP library[3]

$$\epsilon = 0.1 \min\left(\sqrt{v_{h_t}}, \frac{1}{N}||\mathbf{r}_t||_1\right) + e$$

where e is the the float point precision in MATLAB. This choice of the parameter ϵ demonstrated stable estimation throughout iterations t for all the considered algorithms and in all the experiments involving BB-MC, we use a single MC trial (additional execution of the denoiser) to estimate the divergence.

Lastly, because there is no access to the exact value of the divergence α_t for the BM3D denoiser, we use the following approximation

$$\alpha_t^{oracle} = \frac{\mathbf{h}_t^T \mathbf{g}_t(\mathbf{r}_t)}{\mathbf{h}_t^T \mathbf{h}_t} \tag{75}$$

[2]The BM3D library used throughout the simulations can be downloaded from the website of the authors of the denoiser http://www.cs.tut.fi/ foi/GCF-BM3D/. For this particular implementation we used the 'profile' to be 'np'.

[3]The link to the code is https://sourceforge.net/projects/gampmatlab/


Fig. 1: The ground truth images

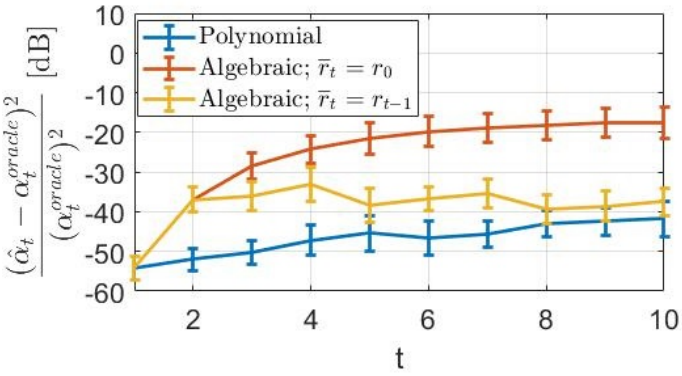

Fig. 2: Divergence estimation error with the standard deviation error bars of the polynomial and the algebraic estimators

to measure the accuracy of the divergence estimates produced. This estimator represents the finite dimensional approximation of the Stein's identity (13).

A. Setting up the polynomial divergence estimator

In all the simulation experiments discussed next, we observed two tendencies of the polynomial divergence estimator. The first one is supporting the idea that there is a pair $\tau \neq \tau'$ that satisfies (71). In fact, the experiments showed that this condition was satisfied by any pair of indices and in the following simulations we will stick to the pair $(t, t-1)$, which led to a slightly better overall performance of SMP algorithms.

The second tendency is related to the root assignment problem. It is observed that when an SMP algorithm makes substantial progress after each iteration, i.e. reduces the intrinsic variance substantially, the divergence α_t is by a few orders of magnitude closer (in the absolute value sense) to the smallest root of the polynomial (61). This tendency is observed up to roughly iteration $t = 10$, which depends on the chosen SMP algorithm and how quickly the algorithms converges. Note that the proposed method for identifying the right root of the polynomial (61) cannot be implemented at the first iteration since it requires a pair of indices $\tau \neq \tau'$. Therefore, motivated by the empirical observation, at the first iteration of an SMP algorithm, we generate only one polynomial with the roots $\hat{\alpha}_1$ and $\hat{\alpha}_2$, and use

$$\hat{\alpha}_t = \min(\hat{\alpha}_1, \hat{\alpha}_2) \tag{76}$$

for the estimate of α_t for $t = 0$.

B. Polynomial vs algebraic estimators

We begin with the comparison of the polynomial estimator (70) against the algebraic estimators (37) with $\bar{\mathbf{r}}_t = \mathbf{r}_0$ and $\bar{\mathbf{r}}_t = \mathbf{r}_{t-1}$. For this purpose, we consider the CG-VAMP algorithm recovering a natural image shown on the right of Figure 1 of dimension 2048 by 2048. We run a single CG-VAMP algorithm with $i = 5$ CG iterations and where α_t is estimated by the polynomial estimator and, additionally, the two algebraic estimators are computed in parallel (these two values are not used within the algorithm and are only archived). For this experiment, we computed the normalized error $\frac{(\hat{\alpha}_t - \alpha_t^{oracle})^2}{(\alpha_t^{oracle})^2}$, where $\hat{\alpha}_t$ corresponds to either the estimate produced by the polynomial or by the two algebraic estimators, and the "oracle" correction α_t^{oracle} is as in (75). The results averaged over 15 iterations are shown on Figure 2. As seen from the figure, the polynomial estimator demonstrates the best accuracy of estimating the "oracle" correction (75), while the algebraic estimator with $\bar{\mathbf{r}}_t = \mathbf{r}_{t-1}$ demonstrates second to the best performance. On the other hand, the algebraic estimator with $\bar{\mathbf{r}}_t = \mathbf{r}_0$ turns out to perform considerably worse than the other two and therefore is not recommended either for computing α_t or for estimating the divergence of the denoiser $\mathbf{g}_t$ for its optimization via SURE.

Next, we assess the stability of CG-VAMP that uses different proposed divergence estimation methods. For this, we compare two CG-VAMP algorithms: one where α_t is computed based on the polynomial estimator as in the previous experiment, and one where α_t is estimated by the algebraic estimator with $\bar{\mathbf{r}}_t = \mathbf{r}_{t-1}$. Here, we computed the same error for α_t and the Normalized MSE (NMSE) $\frac{||\mathbf{g}_t(\mathbf{r}_t) - \mathbf{x}||^2}{||\mathbf{x}||^2}$. The two error measures averaged over 15 realizations are shown on Figure 3. As seen from the left plot depicting the NMSE, the CG-VAMP algorithm with the algebraic estimator with $\bar{\mathbf{r}}_t = \mathbf{r}_{t-1}$ diverges halfway through the execution, while the same algorithm but with the polynomial estimator demonstrates high stability. This result in combination with the previous experiment suggests that the algebraic estimator is capable of producing a relatively accurate estimate of α_t, if it is not used as the main correction method.

C. Black Box Monte Carlo and the polynomial methods for divergence estimation

Next we compare the performance of SMP algorithms when two different divergence estimation methods – BB-MC (2) and the proposed polynomial method (70), are used to estimate α_t. First, we compare these two methods in terms of accuracy, by running two identical CG-VAMP algorithms but with two different divergence estimators. For this, we consider recovering the image 'man' of dimensions 1024 by 1024 shown to the left on Figure 1 and measured by an operator $\mathbf{A}$ with three condition numbers $\kappa(\mathbf{A}) = (100, 1000, 10000)$. As in the previous experiment, we used the fixed number of iterations for the CG algorithm $i = 5$. The NMSE of the algorithms averaged over 15 realizations is shown on Figure

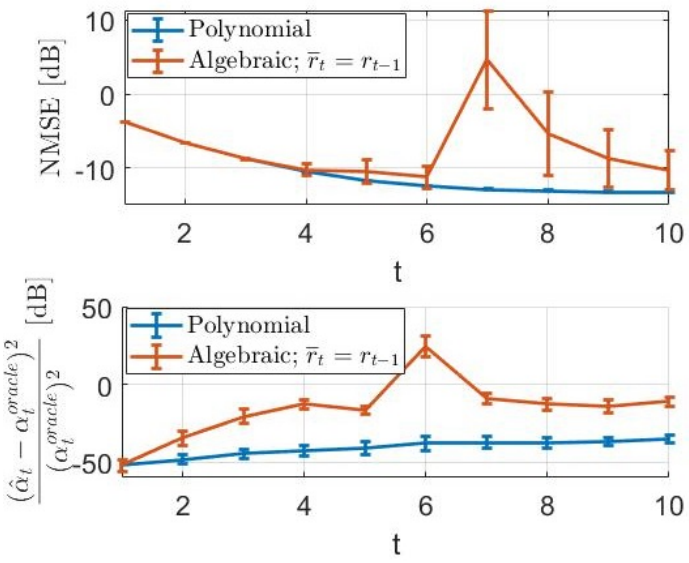

Fig. 3: Top: the NMSE of CG-VAMP algorithms with the two correction methods. Bottom: divergence estimation error with the standard deviation error bars of the polynomial and the algebraic estimators

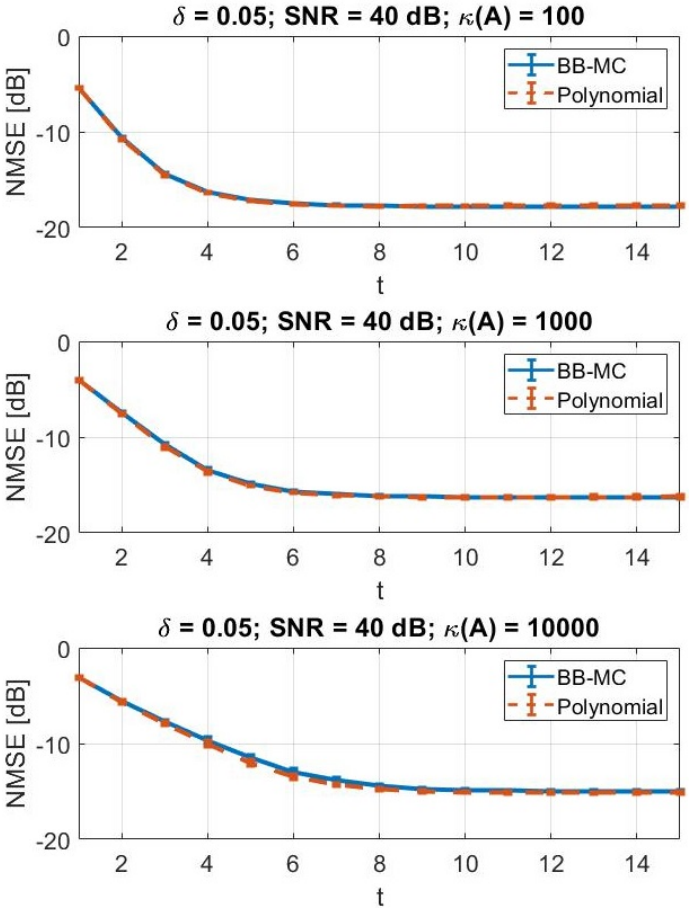

Fig. 4: NMSE with the standard deviation error bars for two CG-VAMP algorithms: with the BB-MC divergence estimator (2) and with the polynomial divergence estimator (70).

4. As we see from the plot, the CG-VAMP algorithm with the polynomial divergence estimator demonstrates a similar reconstruction performance as CG-VAMP with the BB-MC estimator. To demonstrate robustness of the method with respect to other parameters of the inverse problem, we repeated the same experiment but with the SNR of $20dB$ and with higher subsampling factor $\delta = 0.2$, and plotted the result on Figure 5. Additionally, we computed the error of estimating α_t^{oracle} as in the first experiment. The averaged result over 15 realizations for $\kappa(\mathbf{A}) = 1000$ is depicted on Figure 6. As seen from the plot, the polynomial estimator demonstrates higher accuracy of estimation for the initial iterations where CG-VAMP has a substantial per-iteration improvement and exhibits a similar accuracy when CG-VAMP is near the fixed point.

Next we keep the same inverse problem as in the last experiment with $\kappa(\mathbf{A}) = 1000$ and compare the run time and the estimation accuracy of several SMP algorithms when the two divergence estimations methods are used. In particular, we consider the VAMP, CG-VAMP and WS-CG-VAMP algorithms. Each of these algorithms is executed separately with the BB-MC method and with the proposed polynomial method, and the results are averaged over 40 realisations. On Figure 7 we demonstrate the NMSE of the three pairs algorithms and in Table 1 we show the time required for all the algorithms to get to iteration $t = 15$. The first observation is that all the SMP algorithms demonstrate almost identical performance in terms of MSE when we choose different methods for divergence estimation. Secondly, as seen from the table, the run time of the algorithms[4] with the polynomial divergence estimator is almost twice as low as of the same algorithms but with the BB-MC divergence estimator (2). This confirms the initial goal of this work.

TABLE I: Time (in seconds) taken for SMP algorithms with two different divergence estimation methods to execute 15 iterations.

Algorithm	BB-MC estimator	Polynomial estimator
VAMP	164.89	83.55
CG-VAMP	177.11	95.41
WS-CG-VAMP	178.28	96.7

D. AMP

Lastly, we consider the AMP case. As mentioned in Assumption 2, the AMP dynamics is rigorously derived for

[4]Even though here VAMP demonstrates the fastest time-wise convergence, implementing each iteration of the algorithm is only possible because we specifically designed $\mathbf{A}$ as discussed at the beginning of the section. If $\mathbf{A}$ was a general matrix, it would be intractable to implement even a single iteration of VAMP when the size of the inverse problem is as large as in the experiment considered. Yet, we left the run-time performance for VAMP to illustrate the benefit of the proposed technique when one can design this type of measurement matrices $\mathbf{A}$.

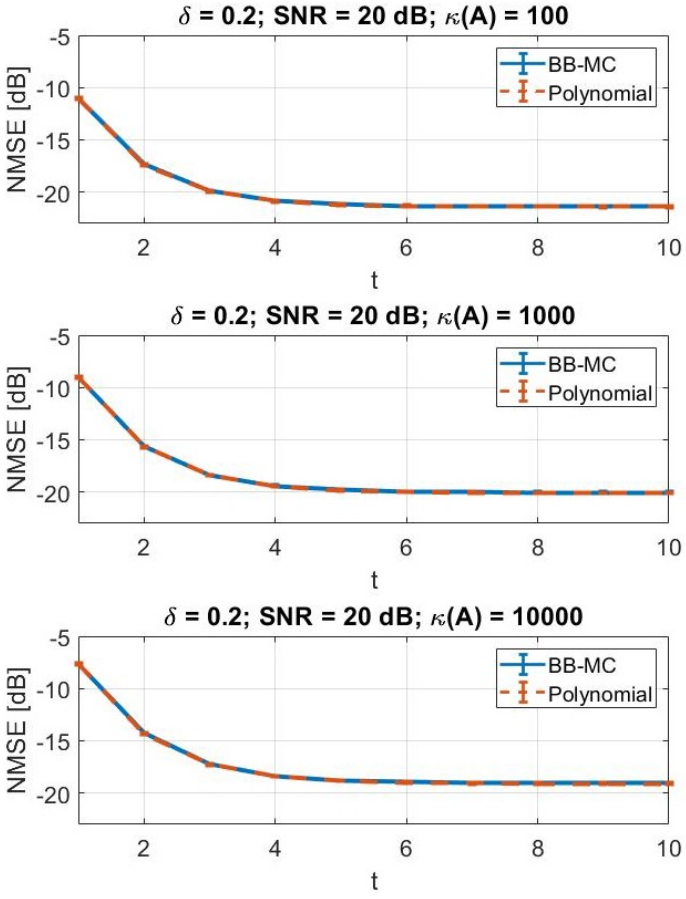

Fig. 5: The same experiments as in Figure 4, but with SNR of $20dB$ and subsampling factor $\delta = 0.2$.

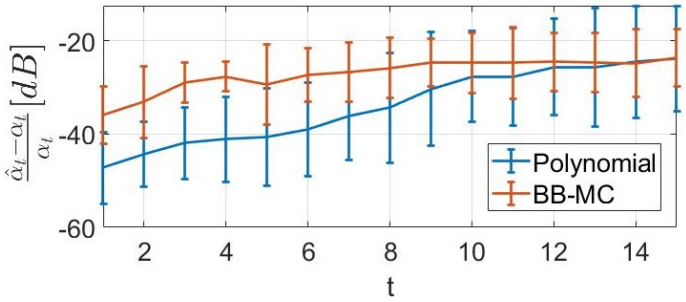

Fig. 6: Mean error with the standard deviation error bars of estimating the correction scalar α_t within CG-VAMP

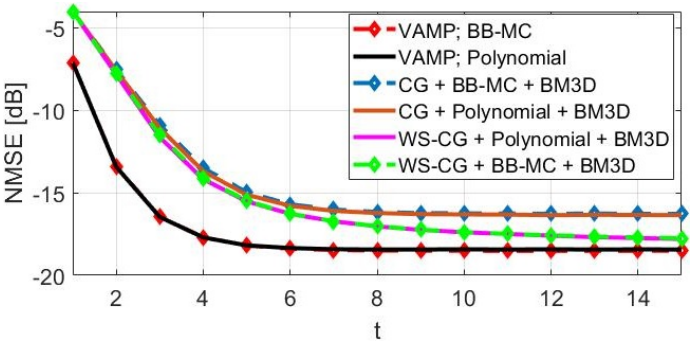

Fig. 7: NMSE of VAMP, CG-VAMP and WS-CG-VAMP algorithms using two different divergence estimation methods.

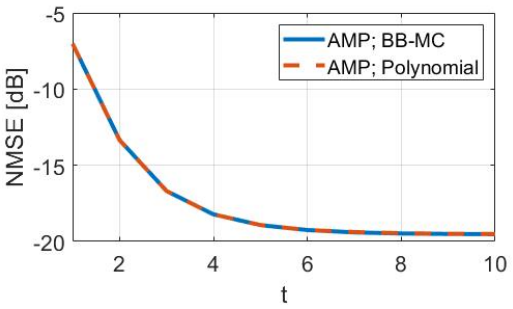

Fig. 8: NMSE of AMP with two different divergence estimation methods.

those measurement systems (1) where $\mathbf{A}^T\mathbf{A}$ has the empirical eigenvalue distribution with the first t moments equivalent to the same order moments of MP law [17]. In the following experiment, we keep the FIJL transform, generate a sequence of i.i.d. MP random values and assign the entries of the matrix $\mathbf{S}$ to be the square root of those random values. The rest of the parameters of the inverse problem are kept the same. The NMSE averaged over 15 realizations of the two version of AMP are shown on Figure 8. As seen from the plot, the two algorithms demonstrate almost identical reconstruction results. However, the AMP version with the polynomial estimator takes 16.23 seconds on average to execute 15 iterations, while the same algorithm but with BB-MC estimator (2) takes 32.1 seconds for the same work.

V. Conclusions

In this work we have proposed two alternatives to the traditional Black-Box Monte Carlo (BB-MC) [1] methods for estimating the divergence of denoisers within SMP algorithms. Similarly to BB-MC, the proposed methods do not use any additional information about the denoiser apart from its input and output. However, contrary to the BB-MC method, the two suggested estimators do not require executing the denoiser additional times and, therefore, significantly accelerate the SMP algorithm when an expensive denoiser such as BM3D is used. The first method - the *algebraic estimator* – has a negligible computational cost and can produce a rough estimate of the divergence of a denoiser, which can be further used to, for example, optimize the performance of the denoising block. The second estimation method – the *polynomial estimator* – complements the first one and demonstrates high robustness with respect to the dimensionality of the inverse problem and a similar accuracy of correction compared to the BB-MC method.

While the two proposed estimators are exact in the large system limit, for finite N their accuracy suffers from additional stochastic error. In future work, we would like to understand why the polynomial estimator is more robust with respect to the decreased dimensionality and whether it is possible to modify the fast algebraic estimator accordingly to increase its robustness. Additionally, from the thorough numerical study (not demonstrated here), we have found that in the polynomial

estimator, the root associated with the divergence α_t is the smallest root and this tendency holds irrespectively of the chosen SMP, denoiser or parameters of the inverse problem (1). Yet, at the moment we do not have a rigorous explanation for this and it would be interesting to get a better understanding of this phenomena.

APPENDIX A

In this appendix we prove Lemma 2. Recall that we define the vector $\bar{\mathbf{s}}_{t+1}(\hat{\alpha},\tau)$ as

$$\bar{\mathbf{s}}_{t+1}(\hat{\alpha},\tau) = \mathbf{g}_t(\mathbf{r}_t) - \hat{\alpha}(\mathbf{r}_t - \mathbf{s}_\tau) \tag{77}$$

For this vector, next, we aim to simplify the function

$$E_{t+1}(\hat{\alpha},\tau) = J^1_{t+1}(\hat{\alpha},\tau) - J^2_{t+1}(\hat{\alpha},\tau) \tag{78}$$

where

$$J^1_{t+1}(\hat{\alpha},\tau) = \frac{1}{N}||\mathbf{r}_t - \bar{\mathbf{s}}_{t+1}(\hat{\alpha},\tau)||^2 - v_{h_t}$$
$$J^2_{t+1}(\hat{\alpha},\tau) = \frac{1}{N}||\mathbf{y} - \mathbf{A}\bar{\mathbf{s}}_{t+1}(\hat{\alpha},\tau)||^2 - \delta v_w$$

To increase the readability, in the following we use $\mathbf{g}_t$ to refer to $\mathbf{g}_t(\mathbf{r}_t)$ and drop the dependence of $\bar{\mathbf{s}}_{t+1}(\hat{\alpha},\tau)$ on $\hat{\alpha}$ and τ. First, we expand the norm in $J^2_{t+1}(\hat{\alpha},\tau)$ to obtain

$$\begin{aligned}&||\mathbf{r}_t - \bar{\mathbf{s}}_{t+1}||^2 = ||\mathbf{r}_t - \mathbf{g}_t + \hat{\alpha}(\mathbf{r}_t - \mathbf{s}_\tau)||^2 \\ &= ||\mathbf{r}_t - \mathbf{g}_t||^2 + 2(\mathbf{r}_t - \mathbf{g}_t)^T(\mathbf{r}_t - \mathbf{s}_\tau)\hat{\alpha} + ||\mathbf{r}_t - \mathbf{s}_\tau||^2\hat{\alpha}^2\end{aligned}$$

Thus, $J^1_{t+1}(\hat{\alpha},\tau)$ is equivalent to

$$J^1_{t+1}(\hat{\alpha},\tau) = k_0 + k_1\hat{\alpha} + k_2\hat{\alpha}^2 \tag{79}$$

where

$$k_0 = \frac{1}{N}||\mathbf{r}_t - \mathbf{g}_t||^2 - v_{h_t} \tag{80}$$
$$k_1 = 2\frac{1}{N}(\mathbf{r}_t - \mathbf{g}_t)^T(\mathbf{r}_t - \mathbf{s}_\tau) \tag{81}$$
$$k_2 = \frac{1}{N}||\mathbf{r}_t - \mathbf{s}_\tau||^2 \tag{82}$$

In the same way, we can show that

$$\begin{aligned}&||\mathbf{y} - \mathbf{A}\bar{\mathbf{s}}_{t+1}(\hat{\alpha},\tau)||^2 = ||\mathbf{y} - \mathbf{A}\mathbf{g}_t + \hat{\alpha}\mathbf{A}(\mathbf{r}_t - \mathbf{s}_\tau)||^2 \\ &= ||\mathbf{y} - \mathbf{A}\mathbf{g}_t||^2 + 2(\mathbf{y} - \mathbf{A}\mathbf{g}_t)^T\mathbf{A}(\mathbf{r}_t - \mathbf{s}_\tau)\hat{\alpha} \\ &+ ||\mathbf{A}(\mathbf{r}_t - \mathbf{s}_\tau)||^2\hat{\alpha}^2\end{aligned} \tag{83}$$

which implies

$$J^2_{t+1}(\hat{\alpha},\tau) = d_0 + d_1\hat{\alpha} + d_2\hat{\alpha}^2 \tag{84}$$

where

$$d_0 = \frac{1}{N}||\mathbf{y} - \mathbf{A}\mathbf{g}_t||^2 - \delta v_w$$
$$d_1 = 2\frac{1}{N}(\mathbf{y} - \mathbf{A}\mathbf{g}_t)^T\mathbf{A}(\mathbf{r}_t - \mathbf{s}_\tau)$$
$$d_2 = \frac{1}{N}||\mathbf{A}(\mathbf{r}_t - \mathbf{s}_\tau)||^2$$

Combining these results, we can show that (78) is equivalent to

$$E_{t+1}(\hat{\alpha},\tau) = u_0 + u_1\hat{\alpha} + u_2\hat{\alpha}^2 \tag{85}$$

with

$$\begin{aligned}u_0 &= k_0 - d_0 \\ &= \frac{1}{N}||\mathbf{r}_t - \mathbf{g}_t||^2 - \frac{1}{N}||\mathbf{y} - \mathbf{A}\mathbf{g}_t||^2 - v_{h_t} + \delta v_w \\ u_1 &= k_1 - d_1 = 2\frac{1}{N}(\mathbf{r}_t - \mathbf{s}_\tau)^T\Big(\mathbf{r}_t - \mathbf{g}_t - \mathbf{A}^T(\mathbf{y} - \mathbf{A}\mathbf{g}_t)\Big) \\ u_2 &= k_2 - d_2 = \frac{1}{N}||\mathbf{r}_t - \mathbf{s}_\tau||^2 - \frac{1}{N}||\mathbf{A}(\mathbf{r}_t - \mathbf{s}_\tau)||^2\end{aligned}$$

which completes the proof.

APPENDIX B

In the following we will study the interaction of the error vectors

$$\mathbf{h}_t = \mathbf{r}_t - \mathbf{x} \qquad \mathbf{q}_t = \mathbf{s}_t - \mathbf{x}$$

where $\mathbf{r}_t$ and $\mathbf{s}_t$ are as in (3) and (4) respectively. Additionally, we will frequently refer to the whole history of these vectors

$$\mathbf{H}_{t+1} = \big(\mathbf{h}_0, \mathbf{h}_1, ..., \mathbf{h}_t\big)$$
$$\mathbf{Q}_{t+1} = \big(\mathbf{q}_0, \mathbf{q}_1, ..., \mathbf{q}_t\big)$$

and their mapped versions

$$\mathbf{M}_{t+1} = \mathbf{V}^T\mathbf{H}_{t+1} \tag{86}$$
$$\mathbf{B}_{t+1} = \mathbf{V}^T\mathbf{Q}_{t+1} \tag{87}$$

Note that these four error vector matrices can be simultaneously represented as

$$\big(\mathbf{M}_{\tau'}, \mathbf{B}_\tau\big) = \mathbf{V}^T\big(\mathbf{H}_{\tau'}, \mathbf{Q}_\tau\big)$$

With this relationship between the error vectors, one can represent the effect of applying the matrix $\mathbf{V}$ through the so-called conditioning technique [6], [8]. For this, define two vectors

$$\boldsymbol{\beta}_\tau = \mathbf{Q}^\dagger_\tau\mathbf{q}_\tau \tag{88}$$
$$\boldsymbol{\rho}_\tau = \mathbf{M}^\dagger_\tau\mathbf{m}_\tau \tag{89}$$

and the set

$$G_{t,t'} = \big\{\mathbf{B}_t, \mathbf{Q}_t, \mathbf{M}_{t'}, \mathbf{H}_{t'}, \mathbf{x}, \tilde{\boldsymbol{w}}, \mathbf{S}\big|\big(\mathbf{M}_{t'}, \mathbf{B}_t\big) = \mathbf{V}^T\big(\mathbf{H}_{t'}, \mathbf{Q}_t\big)\big\} \tag{90}$$

Lastly, for a matrix $\mathbf{R}$, let $\boldsymbol{\Phi}^\perp_{\mathbf{R}}$ be the set of the left-singular vectors associated with the zero singular values of $\mathbf{R}$. With these definitions, one can obtain the following asymptotic result for $\mathbf{V}$ and $\mathbf{V}^T$.

Lemma 3. *[15]: Let Assumptions 1-3 hold. Define a vector $\mathbf{v} \in \mathbb{R}^N$ such that $\lim_{N\to\infty}\frac{1}{N}||\mathbf{v}||^2 = \sigma \leq \infty$. Then, for $\tau = 0, 1, ...$ and $\tau' = 0, 1, ..., \tau$ we have*

1) The matrix $\mathbf{V}^T$ conditioned on the set $G_{\tau,\tau}$ almost surely converges to

$$\begin{aligned}\lim_{N\to\infty}\mathbf{V}^T_{|G_{\tau,\tau}} &\overset{a.s.}{=} \lim_{N\to\infty}(\mathbf{M}_\tau, \mathbf{B}_\tau)\begin{pmatrix}\mathbf{H}^\dagger_\tau \\ \mathbf{Q}^\dagger_\tau\end{pmatrix} \\ &+ \boldsymbol{\Phi}^\perp_{(\mathbf{M}_\tau,\mathbf{B}_\tau)}\tilde{\mathbf{V}}\boldsymbol{\Phi}^\perp_{(\mathbf{H}_\tau,\mathbf{Q}_\tau)}\end{aligned} \tag{91}$$

with $\tilde{\mathbf{V}}$ being Haar distributed and independent of $G_{\tau,\tau}$. Additionally, we have

$$\mathbf{p} = \boldsymbol{\Phi}^\perp_{(\mathbf{M}_\tau,\mathbf{B}_\tau)}\tilde{\mathbf{V}}\boldsymbol{\Phi}^\perp_{(\mathbf{H}_\tau,\mathbf{Q}_\tau)}\mathbf{v} = \breve{\boldsymbol{p}} + \boldsymbol{o}(\mathbf{v}) \tag{92}$$

where $\breve{\boldsymbol{p}} \sim \mathcal{N}(\mathbf{0}, \sigma\mathbf{I})$ is independent of $G_{\tau,\tau}$ and the vector $\boldsymbol{o}(\mathbf{v}) \in \mathbb{R}^N$ satisfies $\lim_{N\to\infty}\frac{1}{N}||\boldsymbol{o}(\mathbf{v})||^2 \overset{a.s.}{=} 0$.

2) *The matrix* $\mathbf{V}$ *conditioned on the set* $G_{\tau+1,\tau}$ *almost surely converges to*

$$\lim_{N\to\infty} \mathbf{V}_{|G_{\tau+1,\tau}} \overset{a.s.}{=} (\mathbf{H}_\tau, \mathbf{Q}_{\tau+1}) \begin{pmatrix} \mathbf{M}_\tau^\dagger \\ \mathbf{B}_{\tau+1}^\dagger \end{pmatrix} + \mathbf{\Phi}^\perp_{(\mathbf{H}_t,\mathbf{Q}_{\tau+1})} \tilde{\mathbf{V}} \mathbf{\Phi}^\perp_{(\mathbf{M}_t,\mathbf{B}_{\tau+1})} \tag{93}$$

with $\tilde{\mathbf{V}}$ *being Haar distributed and independent of* $G_{\tau+1,\tau}$*. Additionally, we have*

$$\mathbf{p} = \mathbf{\Phi}^\perp_{(\mathbf{H}_t,\mathbf{Q}_{\tau+1})} \tilde{\mathbf{V}} \mathbf{\Phi}^\perp_{(\mathbf{M}_t,\mathbf{B}_{\tau+1})} \mathbf{v} = \breve{\boldsymbol{p}} + \boldsymbol{o}(\mathbf{v}) \tag{94}$$

where $\breve{\boldsymbol{p}} \sim \mathcal{N}(\mathbf{0}, \sigma\mathbf{I})$ *is independent of* $G_{\tau+1,\tau}$ *and the vector* $\boldsymbol{o}(\mathbf{v}) \in \mathbb{R}^N$ *satisfies* $\lim_{N\to\infty} \frac{1}{N}||\boldsymbol{o}(\mathbf{v})||^2 \overset{a.s.}{=} 0$.

With this lemma and Theorem 1, we aim to study the behaviour of the following parametrized denoising step and its error

$$\bar{\mathbf{s}}_{\tau+1}(\hat{\alpha}, \tau') = \mathbf{g}_\tau(\mathbf{r}_\tau) - \hat{\alpha}(\mathbf{r}_\tau - \mathbf{s}_{\tau'}) \tag{95}$$

$$\overline{\mathbf{q}}_{t+1}(\hat{\alpha}, \tau') = \bar{\mathbf{s}}_{t+1}(\hat{\alpha}, \tau') - \mathbf{x} \tag{96}$$

In particular, we are interested in the roots of the function

$$E_{\tau+1}(\hat{\alpha}, \tau') = \bar{J}^1_{\tau+1}(\hat{\alpha}, \tau') - \bar{J}^2_{\tau+1}(\hat{\alpha}, \tau') \tag{97}$$

where

$$\bar{J}^1_{\tau+1}(\hat{\alpha}, \tau') = \frac{1}{N}\Big(||\bar{\mathbf{s}}_{\tau+1}(\hat{\alpha}, \tau') - \mathbf{r}_\tau||^2 - ||\mathbf{h}_\tau||^2\Big) \tag{98}$$

$$\bar{J}^2_{\tau+1}(\hat{\alpha}, \tau') = \frac{1}{N}||\mathbf{y} - \mathbf{A}\bar{\mathbf{s}}_{\tau+1}(\hat{\alpha}, \tau')||^2 - \delta v_w \tag{99}$$

Before beginning the analysis, we define two vectors that will arise in the derivation

$$\overline{\boldsymbol{\beta}}_{t+1}(\hat{\alpha}, \tau') = \frac{1}{N}\mathbf{Q}^\dagger_{t+1}\overline{\mathbf{q}}_{t+1}(\hat{\alpha}, \tau') \tag{100}$$

$$\overline{\boldsymbol{\nu}}_{t+1}(\hat{\alpha}, \tau') = \frac{1}{N}\mathbf{H}^\dagger_{t+1}\overline{\mathbf{q}}_{t+1}(\hat{\alpha}, \tau') \tag{101}$$

From Theorem 1 we know that the matrices $\mathbf{Q}_{t+1}$ and $\mathbf{H}_{t+1}$ are asymptotically full rank, so the pseudo-inverses above are well-defined in the limit $N \to \infty$. Additionally, we have that

$$\begin{aligned}
&\lim_{N\to\infty} \frac{1}{N}||\overline{\mathbf{q}}_{\tau+1}(\hat{\alpha}, \tau')||^2 = \lim_{N\to\infty} \frac{1}{N}||\bar{\mathbf{s}}_{\tau+1}(\hat{\alpha}, \tau') - \mathbf{x}||^2 \\
&= \lim_{N\to\infty} \frac{1}{N}||\mathbf{g}_\tau(\mathbf{r}_\tau) - \hat{\alpha}(\mathbf{r}_\tau - \mathbf{s}_{\tau'}) - \mathbf{x}||^2 \\
&= \lim_{N\to\infty} \frac{1}{N}||\mathbf{g}_\tau(\mathbf{x} + \mathbf{h}_\tau) - \hat{\alpha}(\mathbf{h}_\tau - \mathbf{q}_{\tau'}) - \mathbf{x}||^2 \\
&\leq \lim_{N\to\infty} \frac{1}{N}||\mathbf{g}_\tau(\mathbf{x} + \mathbf{h}_\tau)||^2 - \hat{\alpha}^2\Big(\frac{1}{N}||(\mathbf{h}_\tau||^2 - \frac{1}{N}||\mathbf{q}_{\tau'}||^2\Big) \\
&- \frac{1}{N}||\mathbf{x}||^2 < \infty
\end{aligned} \tag{102}$$

where the bound comes from Theorem 1 stating that in the limit $\mathbf{h}_\tau$ and $\mathbf{q}_t$ have finite variances, and Assumption 3 in combination with (18). Thus, in the limit, the vectors (100) and (101) are almost surely finite.

In the following we simplify the notations and drop the dependence of $\bar{\mathbf{s}}_{t+1}(\hat{\alpha}, \tau')$, $\overline{\mathbf{q}}_{t+1}(\hat{\alpha}, \tau')$, $\overline{\boldsymbol{\beta}}_{t+1}(\hat{\alpha}, \tau')$, $\overline{\boldsymbol{\nu}}_{t+1}(\hat{\alpha}, \tau')$, $\bar{J}^1_{\tau+1}(\hat{\alpha}, \tau')$ and of $\bar{J}^2_{\tau+1}(\hat{\alpha}, \tau')$ on $\hat{\alpha}$ and τ'.

A. Analysis of $\bar{J}^1_{\tau+1}$

First, we consider the function $\bar{J}^1_{\tau+1}$ and its limiting behaviour. Since $\bar{\mathbf{s}} = \mathbf{x} + \overline{\mathbf{q}}_{t+1}$ and $\mathbf{r}_\tau = \mathbf{x} + \mathbf{h}_\tau$, we can rewrite (98) as

$$\begin{aligned}
\bar{J}^1_{\tau+1}(\hat{\alpha}, \tau') &= \frac{1}{N}\Big(||\overline{\mathbf{q}}_{\tau+1} - \mathbf{h}_\tau||^2 - ||\mathbf{h}_\tau||^2\Big) \\
&= \frac{1}{N}||\overline{\mathbf{q}}_{\tau+1}||^2 - 2\frac{1}{N}\mathbf{h}_\tau^T\overline{\mathbf{q}}_{\tau+1} = \frac{1}{N}||\overline{\mathbf{q}}_{\tau+1}||^2 + e^1_{\tau+1}
\end{aligned} \tag{103}$$

Next we consider $e^1_{\tau+1}(\hat{\alpha})$ in the last result given that $N \to \infty$ and Theorem 1 holds up to iteration $\tau = t$. First, we can follow the same steps as in the proof of Theorem 2 to obtain

$$\begin{aligned}
\lim_{N\to\infty} \frac{1}{N}\mathbf{h}_t^T\mathbf{g}_t(\mathbf{r}_t) &\overset{a.s.}{=} \lim_{N\to\infty} \frac{1}{N}\breve{\boldsymbol{h}}_t^T\mathbf{g}_t(\mathbf{x} + \breve{\boldsymbol{h}}_t) \\
&\overset{a.s.}{=} v_{h_t}\alpha_t
\end{aligned} \tag{104}$$

Following the same steps and the asymptotic independence result (14), we can obtain

$$\begin{aligned}
\lim_{N\to\infty} \frac{1}{N}\mathbf{h}_t^T(\mathbf{r}_t - \mathbf{s}_{t'}) &= \lim_{N\to\infty} \frac{1}{N}\mathbf{h}_t^T(\mathbf{h}_t - \mathbf{q}_{t'}) \\
&\overset{a.s.}{=} \lim_{N\to\infty} \frac{1}{N}\mathbf{h}_t^T\mathbf{h}_t \overset{a.s.}{=} v_{h_t}
\end{aligned} \tag{105}$$

Combining (104) and (105) implies

$$\begin{aligned}
&\lim_{N\to\infty} e^1_{t+1}(\hat{\alpha}, t') \\
&= -\lim_{N\to\infty} 2\frac{1}{N}\mathbf{h}_t^T\big(\mathbf{g}_t(\mathbf{r}_t) - \hat{\alpha}(\mathbf{r}_t - \mathbf{s}_{t'}) - \mathbf{x}\big) \\
&\overset{a.s.}{=} -2v_{h_t}(\alpha_t - \hat{\alpha})
\end{aligned} \tag{106}$$

Thus, the error term $e^1_{t+1}(\hat{\alpha})$ converges to a linear function of $(\alpha_t - \hat{\alpha})$ in the limit $N \to \infty$. We will return to this result shortly.

B. Analysis of $\bar{J}^2_{\tau+1}$

Next, we analyze $\bar{J}^2_{\tau+1}(\hat{\alpha}, \tau')$ which involves the following norm

$$\begin{aligned}
\frac{1}{N}||\mathbf{y} - \mathbf{A}\bar{\mathbf{s}}_{\tau+1}||^2 &= \frac{1}{N}||\mathbf{w} - \mathbf{A}\overline{\mathbf{q}}_{\tau+1}||^2 \\
&= \frac{1}{N}||\mathbf{w}||^2 + \frac{1}{N}||\mathbf{A}\overline{\mathbf{q}}_{\tau+1}||^2 - 2\frac{1}{N}\mathbf{w}^T\mathbf{A}\overline{\mathbf{q}}_{\tau+1}
\end{aligned} \tag{107}$$

where we used the fact that $\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{w}$. Similarly to the analysis of $\bar{J}^1_{t+1}$, next we assume that $N \to \infty$ and that Theorem 1 holds up to iteration $\tau = t$. Then, we can use (91) to obtain

$$\begin{aligned}
&\lim_{N\to\infty} \mathbf{V}^T_{|G_{t+1,t+1}} \overline{\mathbf{q}}_{t+1} \overset{a.s.}{=} (\mathbf{M}_{t+1}, \mathbf{B}_{t+1}) \begin{pmatrix} \mathbf{H}^\dagger_{t+1}\overline{\mathbf{q}}_{t+1} \\ \mathbf{Q}^\dagger_{t+1}\overline{\mathbf{q}}_{t+1} \end{pmatrix} \\
&+ \mathbf{\Phi}^\perp_{(\mathbf{M}_{t+1},\mathbf{B}_{t+1})} \tilde{\mathbf{V}} \mathbf{\Phi}^\perp_{(\mathbf{H}_{t+1},\mathbf{Q}_{t+1})} \overline{\mathbf{q}}_{t+1} \\
&= \lim_{N\to\infty} \mathbf{M}_{t+1}\overline{\boldsymbol{\nu}}_{\tau+1} + \mathbf{B}_{t+1}\overline{\boldsymbol{\beta}}_{t+1} + \mathbf{p}_{t+1}
\end{aligned} \tag{108}$$

where we used (101) and (100), and defined

$$\mathbf{p}_{t+1} = \mathbf{\Phi}^\perp_{(\mathbf{M}_{t+1},\mathbf{B}_{t+1})} \tilde{\mathbf{V}} \mathbf{\Phi}^\perp_{(\mathbf{H}_{t+1},\mathbf{Q}_{t+1})} \overline{\mathbf{q}}_{t+1} \tag{109}$$

which asymptotically acts as a zero-mean i.i.d. Gaussian vector independent of $\mathbf{M}_{t+1}$, $\mathbf{B}_{t+1}$, $\mathbf{U}$, $\mathbf{S}$ and $\mathbf{w}$, as follows from Lemma 3. With this result we can obtain the following

$$\lim_{N\to\infty}\frac{1}{N}||\mathbf{A}\overline{\mathbf{q}}_{t+1}||^2=\lim_{N\to\infty}\frac{1}{N}||\mathbf{S}\mathbf{V}^T\overline{\mathbf{q}}_{t+1}||^2$$
$$\overset{a.s.}{=}\lim_{N\to\infty}\frac{1}{N}||\mathbf{S}(\mathbf{B}_{t+1}\overline{\boldsymbol{\beta}}_{t+1}+\mathbf{M}_{t+1}\overline{\boldsymbol{\nu}}_{t+1}+\mathbf{p}_{t+1})||^2$$
$$\overset{a.s.}{=}\lim_{N\to\infty}\frac{1}{N}||\mathbf{S}(\breve{\boldsymbol{B}}_{t+1}\overline{\boldsymbol{\beta}}_{t+1}+\mathbf{M}_{t+1}\overline{\boldsymbol{\nu}}_{t+1}+\mathbf{p}_{t+1})||^2 \quad (110)$$
$$\overset{a.s.}{=}\lim_{N\to\infty}\frac{1}{N}\Big(||\mathbf{S}\breve{\boldsymbol{B}}_{t+1}\overline{\boldsymbol{\beta}}_{t+1}||^2+||\mathbf{S}\mathbf{p}_{t+1}||^2$$
$$+||\mathbf{S}\mathbf{M}_{t+1}\overline{\boldsymbol{\nu}}_{t+1}||^2\Big)+\frac{2}{N}\overline{\boldsymbol{\beta}}_{t+1}^T\breve{\boldsymbol{B}}_{t+1}^T\mathbf{S}^T\mathbf{S}\mathbf{M}_{t+1}\overline{\boldsymbol{\nu}}_{t+1} \quad (111)$$
$$\overset{a.s.}{=}\lim_{N\to\infty}\frac{1}{N}\Big(||\breve{\boldsymbol{B}}_{t+1}\overline{\boldsymbol{\beta}}_{t+1}||^2+||\mathbf{p}_{t+1}||^2$$
$$+||\mathbf{S}\mathbf{M}_{t+1}\overline{\boldsymbol{\nu}}_{t+1}||^2\Big)+\frac{2}{N}\overline{\boldsymbol{\beta}}_{t+1}^T\breve{\boldsymbol{B}}_{t+1}^T\mathbf{S}^T\mathbf{S}\mathbf{M}_{t+1}\overline{\boldsymbol{\nu}}_{t+1} \quad (112)$$

where in (110) we used the asymptotic model of the error vectors $\mathbf{b}_\tau$ from Theorem 1, (111) follows from the fact that $\mathbf{p}_{t+1}$ is asymptotically independent of $\breve{\boldsymbol{B}}_{t+1}$ and $\mathbf{M}_{t+1}$ and in (112) we used the normalization $\frac{1}{N}Tr\{\mathbf{S}^T\mathbf{S}\}=1$, the Stein's Lemma and the fact that the columns of $\breve{\boldsymbol{B}}_{t+1}$ and the vector $\mathbf{p}_{t+1}$ are i.i.d. Gaussian to obtain

$$\lim_{N\to\infty}\frac{1}{N}||\mathbf{S}\breve{\boldsymbol{B}}_{t+1}\overline{\boldsymbol{\beta}}_{t+1}||^2\overset{a.s.}{=}\lim_{N\to\infty}\frac{1}{N}||\breve{\boldsymbol{B}}_{t+1}\overline{\boldsymbol{\beta}}_{t+1}||^2 \quad (113)$$
$$\lim_{N\to\infty}\frac{1}{N}||\mathbf{S}\mathbf{p}_{t+1}||^2\overset{a.s.}{=}\lim_{N\to\infty}\frac{1}{N}||\mathbf{p}_{t+1}||^2 \quad (114)$$

To relate (112) to the MSE of $\overline{\mathbf{q}}_{t+1}$, next we consider the later. Again, by referring to (108) and following the same steps as in (112), we can obtain

$$\lim_{N\to\infty}\frac{1}{N}||\overline{\mathbf{q}}_{t+1}||^2$$
$$\overset{a.s.}{=}\lim_{N\to\infty}\frac{1}{N}||\mathbf{B}_{t+1}\overline{\boldsymbol{\beta}}_{t+1}+\mathbf{M}_{t+1}\overline{\boldsymbol{\nu}}_{t+1}+\mathbf{p}_{t+1}||^2$$
$$\overset{a.s.}{=}\lim_{N\to\infty}\frac{1}{N}||\breve{\boldsymbol{B}}_{t+1}\overline{\boldsymbol{\beta}}_{t+1}+\mathbf{M}_{t+1}\overline{\boldsymbol{\nu}}_{t+1}+\mathbf{p}_{t+1}||^2$$
$$\overset{a.s.}{=}\lim_{N\to\infty}\frac{1}{N}||\breve{\boldsymbol{B}}_{t+1}\overline{\boldsymbol{\beta}}_{t+1}||^2+\frac{1}{N}||\mathbf{M}_{t+1}\overline{\boldsymbol{\nu}}_{t+1}||^2+\frac{1}{N}||\mathbf{p}_{t+1}||^2$$

where we used the asymptotic independence of $\mathbf{b}_\tau$ and $\mathbf{m}_{\tau'}$, which follows from (14). By comparing the last result to (112), we find that $\lim_{N\to\infty}\frac{1}{N}||\mathbf{A}\overline{\mathbf{q}}_{t+1}||^2$ is equivalent to

$$\lim_{N\to\infty}\frac{1}{N}||\mathbf{A}\overline{\mathbf{q}}_{t+1}||^2\overset{a.s.}{=}\lim_{N\to\infty}\frac{1}{N}||\overline{\mathbf{q}}_{t+1}||^2$$
$$+\frac{1}{N}||\mathbf{S}\mathbf{M}_{t+1}\overline{\boldsymbol{\nu}}_{t+1}||^2-\frac{1}{N}||\mathbf{M}_{t+1}\overline{\boldsymbol{\nu}}_{t+1}||^2$$
$$+\frac{2}{N}\overline{\boldsymbol{\beta}}_{t+1}^T\breve{\boldsymbol{B}}_{t+1}^T\mathbf{S}^T\mathbf{S}\mathbf{M}_{t+1}\overline{\boldsymbol{\nu}}_{t+1} \quad (115)$$

Then, define a function

$$e_{t+1}^2(\hat{\alpha})=\frac{1}{N}||\mathbf{S}\mathbf{M}_{t+1}\overline{\boldsymbol{\nu}}_{t+1}||^2-\frac{1}{N}||\mathbf{M}_{t+1}\overline{\boldsymbol{\nu}}_{t+1}||^2$$
$$+\frac{2}{N}\overline{\boldsymbol{\beta}}_{t+1}^T\mathbf{B}_{t+1}^T\mathbf{S}^T\mathbf{S}\mathbf{M}_{t+1}\overline{\boldsymbol{\nu}}_{t+1}-2\frac{1}{N}\mathbf{w}^T\mathbf{A}\overline{\mathbf{q}}_{t+1} \quad (116)$$

Assuming $N\to\infty$ and Theorem 1 holds up to iteration $\tau=t$, we can use (116), (115) and (107) to show that the function $\bar{J}_{\tau+1}^2$ from (99) almost surely converges to

$$\lim_{N\to\infty}\bar{J}_{\tau+1}^2\overset{a.s.}{=}\lim_{N\to\infty}\frac{1}{N}||\overline{\mathbf{q}}_{t+1}||^2+e_{t+1}^2(\hat{\alpha}) \quad (117)$$

Next, we analyze the behaviour of the error $e_{t+1}^2(\hat{\alpha})$. First, we consider the term $\frac{1}{N}\mathbf{w}^T\mathbf{A}\overline{\mathbf{q}}_{t+1}$. Using the SVD of $\mathbf{A}=\mathbf{U}\mathbf{S}\mathbf{V}^T$, the model (108) and defining $\tilde{\boldsymbol{w}}=\mathbf{U}\mathbf{w}$, we can obtain

$$\lim_{N\to\infty}\frac{1}{N}\mathbf{w}^T\mathbf{A}\overline{\mathbf{q}}_{t+1}\overset{a.s.}{=}\lim_{N\to\infty}\frac{1}{N}\tilde{\boldsymbol{w}}^T\mathbf{S}\mathbf{V}^T\overline{\mathbf{q}}_{t+1}$$
$$=\lim_{N\to\infty}\frac{1}{N}\tilde{\boldsymbol{w}}^T\mathbf{S}\big(\mathbf{M}_{t+1}\overline{\boldsymbol{\nu}}_{t+1}+\mathbf{B}_{t+1}\overline{\boldsymbol{\beta}}_{t+1}+\mathbf{p}_{t+1}\big)$$
$$\overset{a.s.}{=}\lim_{N\to\infty}\frac{1}{N}\tilde{\boldsymbol{w}}^T\mathbf{S}\mathbf{M}_{t+1}\overline{\boldsymbol{\nu}}_{t+1} \quad (118)$$

where we used the asymptotic independence of $\tilde{\boldsymbol{w}}$ and $\mathbf{p}_{t+1}$ and of $\tilde{\boldsymbol{w}}$ and $\mathbf{S}\mathbf{b}_\tau$ as follows from Theorem 3 and (15) respectively. Next, we analyze the vector $\overline{\boldsymbol{\nu}}_{t+1}$ in the large system limit

$$\lim_{N\to\infty}\overline{\boldsymbol{\nu}}_{t+1}=\mathbf{H}_t^\dagger\overline{\mathbf{q}}_{t+1}$$
$$=\lim_{N\to\infty}(\frac{1}{N}\mathbf{H}_{t+1}^T\mathbf{H}_{t+1})^{-1}\frac{1}{N}\mathbf{H}_{t+1}^T\overline{\mathbf{q}}_{t+1} \quad (119)$$

Here, the vector $\frac{1}{N}\mathbf{H}_{t+1}^T\overline{\mathbf{q}}_{t+1}$ is composed of the elements $\big(\frac{1}{N}\mathbf{H}_{t+1}^T\overline{\mathbf{q}}_{t+1}\big)_k=\frac{1}{N}\mathbf{h}_k^T\overline{\mathbf{q}}_{t+1}$. Following the same steps as in (106), we can show that this element almost surely converges to

$$\lim_{N\to\infty}\frac{1}{N}\mathbf{h}_k^T\overline{\mathbf{q}}_{t+1}\overset{a.s.}{=}\lim_{N\to\infty}\frac{1}{N}\mathbf{h}_k^T\mathbf{h}_t(\alpha_t-\hat{\alpha}) \quad (120)$$

which together with (119) implies that

$$\lim_{N\to\infty}\overline{\boldsymbol{\nu}}_{t+1}\overset{a.s.}{=}\mathbf{e}_{t+1}(\alpha_t-\hat{\alpha}) \quad (121)$$

where $\mathbf{e}_{t+1}\in\mathbb{R}^{t+1}$ is $(t+1)$th vector of the $t+1$ dimensional natural basis. Since $\mathbf{M}_{t+1}\mathbf{e}_{t+1}=\mathbf{m}_t$, substituting this result into (118) leads to

$$\lim_{N\to\infty}\frac{1}{N}\mathbf{w}^T\mathbf{A}\overline{\mathbf{q}}_{t+1}\overset{a.s.}{=}\lim_{N\to\infty}\frac{1}{N}\tilde{\boldsymbol{w}}^T\mathbf{S}\mathbf{m}_t(\alpha_t-\hat{\alpha})$$
$$\overset{a.s.}{=}v_w(\alpha_t-\hat{\alpha}) \quad (122)$$

where the the asymptotic result $\lim_{N\to\infty}\frac{1}{N}\tilde{\boldsymbol{w}}^T\mathbf{S}\mathbf{m}_t\overset{a.s.}{=}v_w$ was proven in [10].

Following similar steps, we can show that

$$\lim_{N\to\infty}\frac{1}{N}||\mathbf{S}\mathbf{M}_{t+1}\overline{\boldsymbol{\nu}}_{t+1}||^2-\frac{1}{N}||\mathbf{M}_{t+1}\overline{\boldsymbol{\nu}}_{t+1}||^2$$
$$+\frac{2}{N}\overline{\boldsymbol{\beta}}_{t+1}^T\mathbf{B}_{t+1}^T\mathbf{S}^T\mathbf{S}\mathbf{M}_{t+1}\overline{\boldsymbol{\nu}}_{t+1}$$
$$\overset{a.s.}{=}\lim_{N\to\infty}\frac{1}{N}||\mathbf{S}\mathbf{m}_t||^2(\alpha_t-\hat{\alpha})^2-\frac{1}{N}||\mathbf{m}_t||^2(\alpha_t-\hat{\alpha})^2$$
$$+\frac{2}{N}\overline{\boldsymbol{\beta}}_{t+1}^T\mathbf{B}_{t+1}^T\mathbf{S}^T\mathbf{S}\mathbf{m}_t(\alpha_t-\hat{\alpha}) \quad (123)$$

Combining this result with (122), we conclude that the error e_2 from (116) almost surely converges to

$$\lim_{N\to\infty}e_{t+1}^2(\hat{\alpha})\overset{a.s.}{=}\lim_{N\to\infty}\Big(\frac{1}{N}||\mathbf{S}\mathbf{m}_t||^2-v_{h_t}\Big)(\alpha_t-\hat{\alpha})^2$$
$$+2\Big(\frac{1}{N}\overline{\boldsymbol{\beta}}_{t+1}^T\mathbf{B}_{t+1}^T\mathbf{S}^T\mathbf{S}\mathbf{m}_t-v_w\Big)(\alpha_t-\hat{\alpha}) \quad (124)$$

C. Roots of $E_{\tau+1}(\hat{\alpha},\tau')$

Combining (103) with (106) and (117) with (124), we can obtain the following asymptotic result for $E_{\tau+1}(\hat{\alpha},\tau')$ from (97) under the Assumptions 1-3 and assuming Theorem 1 holds up to iteration $\tau=t$.

$$\lim_{N\to\infty} E_{t+1}(\hat{\alpha},\tau') \overset{a.s.}{=} \lim_{N\to\infty} e^1_{t+1}(\hat{\alpha}) - e^2_{t+1}(\hat{\alpha})$$
$$\overset{a.s.}{=} \lim_{N\to\infty} \Big(v_{h_t} - \frac{1}{N}||\mathbf{Sm}_t||^2\Big)(\alpha_t-\hat{\alpha})^2$$
$$-2\Big(v_{h_t} + \frac{1}{N}\overline{\boldsymbol{\beta}}^T_{t+1}\mathbf{B}^T_{t+1}\mathbf{S}^T\mathbf{Sm}_t - v_w\Big)(\alpha_t-\hat{\alpha}) \quad (125)$$

From this result, we immediately notice that the first root $\hat{\alpha}_1$ to $E_{t+1}(\hat{\alpha},\tau')$ almost surely converges to $\lim_{N\to\infty}\hat{\alpha}_1 = \alpha_t$. Next, we aim to obtain the closed-form solution for the second root $\hat{\alpha}_2$ to this function. Consider the asymptotic behaviour of $\overline{\boldsymbol{\beta}}_{t+1}$ from (100)

$$\lim_{N\to\infty}\overline{\boldsymbol{\beta}}_{t+1}(\hat{\alpha},\tau') = \lim_{N\to\infty}\mathbf{Q}^\dagger_{t+1}\overline{\mathbf{q}}_{t+1}(\hat{\alpha},\tau')$$
$$= \lim_{N\to\infty}\mathbf{Q}^\dagger_{t+1}\big(\bar{\mathbf{s}}_{t+1}(\hat{\alpha},\tau') - \mathbf{x}\big)$$
$$= \lim_{N\to\infty}\mathbf{Q}^\dagger_{t+1}\big(\mathbf{g}_\tau(\mathbf{r}_t) - \hat{\alpha}(\mathbf{r}_t - \mathbf{s}_{\tau'}) - \mathbf{x}\big)$$
$$= \lim_{N\to\infty}\mathbf{Q}^\dagger_{t+1}\big(\mathbf{g}_t(\mathbf{r}_t) - \hat{\alpha}(\mathbf{h}_t - \mathbf{q}_{\tau'}) - \mathbf{x}\big)$$
$$\overset{a.s.}{=} \lim_{N\to\infty}\mathbf{Q}^\dagger_{t+1}\big(\mathbf{g}_t(\mathbf{r}_t) + \hat{\alpha}\mathbf{q}_{\tau'} - \mathbf{x}\big) \quad (126)$$

where in the last step we used the fact that $\mathbf{Q}_{t+1} = (\mathbf{q}_0, ..., \mathbf{q}_t)$ and (14) to obtain

$$\lim_{N\to\infty}\mathbf{Q}^\dagger_{t+1}\mathbf{h}_t = \lim_{N\to\infty}(\frac{1}{N}\mathbf{Q}^T_{t+1}\mathbf{Q}_{t+1})^{-1}\frac{1}{N}\mathbf{Q}_{t+1}\mathbf{h}_t \overset{a.s.}{=} \mathbf{0}$$

Additionally, note that

$$\mathbf{Q}^\dagger_{t+1}\mathbf{q}_{\tau'} = \mathbf{e}_{\tau'+1} \quad (127)$$

where $\mathbf{e}_i \in \mathbb{R}^{t+1}$ is the i-th vector of the natural basis. Therefore (126) is equivalent to

$$\overline{\boldsymbol{\beta}}_{t+1}(\hat{\alpha},\tau') = \lim_{N\to\infty}\mathbf{Q}^\dagger_{t+1}\big(\mathbf{g}_t(\mathbf{r}_t) - \mathbf{x}\big) + \hat{\alpha}\mathbf{e}_{\tau'+1}$$
$$=: \beta_{t+1} + \hat{\alpha}\mathbf{e}_{\tau'+1} \quad (128)$$

where we defined

$$\beta_{t+1} := \lim_{N\to\infty}\mathbf{Q}^\dagger_{t+1}\big(\mathbf{g}_t(\mathbf{r}_t) - \mathbf{x}\big) \quad (129)$$

Using (128) and grouping terms together, we can rewrite the cost function $E_{t+1}(\hat{\alpha},\tau')$ from (125) as

$$\lim_{N\to\infty} E_{t+1}(\hat{\alpha},\tau')$$
$$\overset{a.s.}{=} \lim_{N\to\infty}(c^0_t + \hat{\alpha}c^1_t(\tau'))(\alpha_t-\hat{\alpha}) + c^2_t(\alpha_t-\hat{\alpha})^2$$
$$= \lim_{N\to\infty}\Big(c^0_t + \hat{\alpha}c^1_t(\tau') + c^2_t(\alpha_t-\hat{\alpha})\Big)(\alpha_t-\hat{\alpha}) \quad (130)$$

where

$$c^0_t = -2\big(v_{h_t} - v_w + \frac{1}{N}\beta^T_{t+1}\mathbf{B}^T_{t+1}\mathbf{S}^T\mathbf{Sm}_t\big) \quad (131)$$
$$c^1_t(\tau') = -2\frac{1}{N}\mathbf{b}^T_{\tau'}\mathbf{S}^T\mathbf{Sm}_t \quad (132)$$
$$c^2_t = v_{h_t} - \frac{1}{N}||\mathbf{Sm}_t||^2 \quad (133)$$

Thus, the second root $\hat{\alpha}_2$ of this function follows

$$\lim_{N\to\infty} c^0_t + \hat{\alpha}_2 c^1_t(\tau') + c^2_t(\alpha_t - \hat{\alpha}_2) = 0 \quad (134)$$

Solving for $\hat{\alpha}_2$ gives

$$\lim_{N\to\infty}\hat{\alpha}_2(\tau') \overset{a.s.}{=} \lim_{N\to\infty}\frac{c^0_t + c^2_t\alpha_t}{c^2_t - c^1_t(\tau')} \quad (135)$$

which completes the proof.

APPENDIX C

In this section, we prove Lemma 1 which states that Theorem 1 holds for the error vectors $\tilde{\boldsymbol{h}}_t = \tilde{\boldsymbol{r}}_t - \mathbf{x}$ and $\tilde{\mathbf{q}}_t = \tilde{\boldsymbol{s}}_t - \mathbf{x}$, where

$$\tilde{\boldsymbol{r}}_t = \frac{1}{\tilde{C}_r}\Big(\mathbf{A}^T\boldsymbol{f}_t(\tilde{\boldsymbol{S}}_{t+1},\mathbf{y}) - \tilde{\boldsymbol{S}}_{t+1}\boldsymbol{\gamma}_t\Big) \quad (136)$$
$$\tilde{\boldsymbol{s}}_{t+1} = \frac{1}{\tilde{C}_s}\Big(\mathbf{g}_t(\tilde{\boldsymbol{r}}_t) - \tilde{\boldsymbol{r}}_t\tilde{\alpha}_t\Big) \quad (137)$$

and $\tilde{\alpha}_t$ satisfies

$$\lim_{N\to\infty}\tilde{\alpha}_t \overset{a.s.}{=} \lim_{N\to\infty}\alpha_t \quad (138)$$

at ever iteration. The proof is by induction: we assume that $\frac{1}{N}||\mathbf{r}_t - \tilde{\boldsymbol{r}}_t||^2 \overset{a.s.}{=} 0$ holds and that this implies $\frac{1}{N}||\mathbf{s}_{t+1} - \tilde{\boldsymbol{s}}_{t+1}||^2 \overset{a.s.}{=} 0$. The implication that $\frac{1}{N}||\mathbf{s}_{t+1} - \tilde{\boldsymbol{s}}_{t+1}||^2 \overset{a.s.}{=} 0$ leads to $\frac{1}{N}||\mathbf{r}_{t+1} - \tilde{\boldsymbol{r}}_{t+1}||^2 \overset{a.s.}{=} 0$ is proved in the same way.

Using the triangular inequality, we can bound the difference of $\mathbf{s}_{t+1}$ and $\tilde{\boldsymbol{s}}_{t+1}$ from (4) and (137) as

$$||\mathbf{s}_{t+1} - \tilde{\boldsymbol{s}}_{t+1}||^2$$
$$= \Big|\Big|\frac{1}{C_s}\big(\mathbf{g}_t(\mathbf{r}_t) - \mathbf{r}_t\alpha_t\big) - \frac{1}{\tilde{C}_s}\big(\mathbf{g}_t(\tilde{\boldsymbol{r}}_t) + \tilde{\boldsymbol{r}}_t\tilde{\alpha}_t\big)\Big|\Big|^2$$
$$\leq \Big|\Big|\frac{1}{C_s}\mathbf{g}_t(\mathbf{r}_t) - \frac{1}{\tilde{C}_s}\mathbf{g}_t(\tilde{\boldsymbol{r}}_t)\Big|\Big|^2 + \Big|\Big|\frac{1}{\tilde{C}_s}\tilde{\boldsymbol{r}}_t\tilde{\alpha}_t - \frac{1}{C_s}\mathbf{r}_t\alpha_t\Big|\Big|^2 \quad (139)$$

Similarly, we can show the following for the first inner-product from above

$$\Big|\Big|\frac{1}{C_s}\mathbf{g}_t(\mathbf{r}_t) - \frac{1}{\tilde{C}_s}\mathbf{g}_t(\tilde{\boldsymbol{r}}_t)\Big|\Big|^2$$
$$= \Big|\Big|\frac{1}{C_s}\mathbf{g}_t(\mathbf{r}_t) - \frac{1}{\tilde{C}_s}\mathbf{g}_t(\tilde{\boldsymbol{r}}_t) - \frac{1}{C_s}\mathbf{g}_t(\tilde{\boldsymbol{r}}_t) + \frac{1}{C_s}\mathbf{g}_t(\tilde{\boldsymbol{r}}_t)\Big|\Big|^2$$
$$= \Big|\Big|\frac{1}{C_s}(\mathbf{g}_t(\mathbf{r}_t) - \mathbf{g}_t(\tilde{\boldsymbol{r}}_t)) + \Big(\frac{1}{C_s} - \frac{1}{\tilde{C}_s}\Big)\mathbf{g}_t(\tilde{\boldsymbol{r}}_t)\Big|\Big|^2$$
$$\leq \Big|\Big|\frac{1}{C_s}(\mathbf{g}_t(\mathbf{r}_t) - \mathbf{g}_t(\tilde{\boldsymbol{r}}_t))\Big|\Big|^2 + \Big|\Big|\Big(\frac{1}{C_s} - \frac{1}{\tilde{C}_s}\Big)\mathbf{g}_t(\tilde{\boldsymbol{r}}_t)\Big|\Big|^2 \quad (140)$$

Here, we can use Assumption 3 about $\mathbf{g}_t$ being a Lipschitz continuous function, which implies

$$\lim_{N\to\infty}\frac{1}{C^2_s}\frac{1}{N}\Big|\Big|\mathbf{g}_t(\mathbf{r}_t) - \mathbf{g}_t(\tilde{\boldsymbol{r}}_t)\Big|\Big|^2 \leq \lim_{N\to\infty}\frac{L}{C^2_s}\frac{1}{N}||\mathbf{r}_t - \tilde{\boldsymbol{r}}_t||^2$$
$$\overset{a.s.}{=} 0 \quad (141)$$

where $L = O(1)$ is some constant and the last step follows from the induction hypothesis. Recall that the definition of the scalars C_s and $\tilde{C}_s$ are [8], [15]

$$C_s = 1 - \alpha_t \qquad \tilde{C}_s = 1 - \tilde{\alpha}_t$$

so that $\frac{1}{C_s} - \frac{1}{\tilde{C}_s} = \frac{\tilde{\alpha}_t - \alpha_t}{(1-\alpha_t)(1-\tilde{\alpha}_t)}$. Then we can use Assumption 3 stating that the norm $\frac{1}{N}||\mathbf{g}_t(\tilde{\boldsymbol{r}}_t)||^2$ is bounded and (138) to obtain

$$\lim_{N\to\infty}\Big(\frac{1}{C_s} - \frac{1}{\tilde{C}_s}\Big)^2\frac{1}{N}\Big|\Big|\mathbf{g}_t(\tilde{\boldsymbol{r}}_t)\Big|\Big|^2$$
$$= \lim_{N\to\infty}\left(\frac{\tilde{\alpha}_t - \alpha_t}{(1-\alpha_t)(1-\tilde{\alpha}_t)}\right)^2\frac{1}{N}\Big|\Big|\mathbf{g}_t(\tilde{\boldsymbol{r}}_t)\Big|\Big|^2 \overset{a.s.}{=} 0 \quad (142)$$

Thus, the first component of (139) almost surely converges to zero. In the same way, we can analyze the second component. Following the same steps, we obtain

$$\begin{aligned}
&\Big|\Big|\frac{1}{\tilde{C}_s}\tilde{\boldsymbol{r}}_t\tilde{\alpha}_t - \frac{1}{C_s}\mathbf{r}_t\alpha_t\Big|\Big|^2 \\
&= \Big|\Big|\frac{1}{\tilde{C}_s}\tilde{\boldsymbol{r}}_t\tilde{\alpha}_t - \frac{1}{C_s}\mathbf{r}_t\alpha_t - \frac{1}{C_s}\tilde{\boldsymbol{r}}_t\tilde{\alpha}_t + \frac{1}{C_s}\tilde{\boldsymbol{r}}_t\tilde{\alpha}_t\Big|\Big|^2 \\
&= \Big|\Big|\Big(\frac{1}{\tilde{C}_s} - \frac{1}{C_s}\Big)\tilde{\boldsymbol{r}}_t\tilde{\alpha}_t + \frac{1}{C_s}\big(\tilde{\boldsymbol{r}}_t\tilde{\alpha}_t - \mathbf{r}_t\alpha_t\big)\Big|\Big|^2 \\
&\leq \Big(\frac{1}{\tilde{C}_s} - \frac{1}{C_s}\Big)^2\tilde{\alpha}^2||\tilde{\boldsymbol{r}}_t||^2 + \frac{1}{C_s^2}\Big|\Big|\tilde{\boldsymbol{r}}_t\tilde{\alpha}_t - \mathbf{r}_t\alpha_t\Big|\Big|^2 \qquad (143)
\end{aligned}$$

Since $\mathbf{r}_t = \mathbf{x} + \mathbf{h}_t$, and both of these vectors have bounded second moments as follows from Theorem 1 and Assumption 3, we have that

$$\lim_{N\to\infty}\Big(\frac{1}{\tilde{C}_s} - \frac{1}{C_s}\Big)^2\tilde{\alpha}^2\frac{1}{N}||\tilde{\boldsymbol{r}}_t||^2 \overset{a.s.}{=} 0 \qquad (144)$$

Similarly, we have

$$\begin{aligned}
&||\tilde{\boldsymbol{r}}_t\tilde{\alpha}_t - \mathbf{r}_t\alpha_t||^2 \\
&= ||\tilde{\boldsymbol{r}}_t\tilde{\alpha}_t - \mathbf{r}_t\alpha_t + \mathbf{r}_t\tilde{\alpha}_t - \mathbf{r}_t\tilde{\alpha}_t||^2 \\
&= ||(\tilde{\boldsymbol{r}}_t - \mathbf{r}_t)\tilde{\alpha}_t + (\tilde{\alpha}_t - \alpha_t)\mathbf{r}_t||^2 \\
&\leq \tilde{\alpha}_t^2||\tilde{\boldsymbol{r}}_t - \mathbf{r}_t||^2 + (\tilde{\alpha}_t - \alpha_t)^2||\mathbf{r}_t||^2 \overset{a.s.}{=} 0 \qquad (145)
\end{aligned}$$

where we used the induction hypothesis and (138). Combining all the above results confirms that (139) almost surely converges to zero under the induction hypothesis. Since $\boldsymbol{f}_t(\tilde{\boldsymbol{S}}_{t+1}, \mathbf{y})$ is a linear mapping, which further implies that $\boldsymbol{\gamma}_t$ is Lipschitz continues as follows from the definition (11), the proof of that $\frac{1}{N}||\mathbf{s}_{t+1} - \tilde{\boldsymbol{s}}_{t+1}||^2 \overset{a.s.}{=} 0$ implies $\frac{1}{N}||\mathbf{r}_{t+1} - \tilde{\boldsymbol{r}}_{t+1}||^2 \overset{a.s.}{=} 0$ follows exactly the same steps as above.

APPENDIX D

Next, we show that for a pair of indices $\tau \neq \tau'$, $(\tau, \tau') \leq t$, the condition

$$\lim_{N\to\infty}\frac{1}{N}\mathbf{q}_\tau^T\mathbf{A}^T\mathbf{A}\mathbf{h}_t \neq \lim_{N\to\infty}\frac{1}{N}\mathbf{q}_{\tau'}^T\mathbf{A}^T\mathbf{A}\mathbf{h}_t \qquad (146)$$

is asymptotically equivalent to

$$\lim_{N\to\infty}\frac{1}{N}\mathbf{z}_\tau^T(\mathbf{y} - \mathbf{A}\mathbf{r}_t) \neq \lim_{N\to\infty}\frac{1}{N}\mathbf{z}_{\tau'}^T(\mathbf{y} - \mathbf{A}\mathbf{r}_t). \qquad (147)$$

Here $\mathbf{y}$ and $\mathbf{A}$ are from (1), $\mathbf{z}_\tau = \mathbf{y} - \mathbf{A}\mathbf{s}_t$ and the error vectors are

$$\mathbf{q}_t = \mathbf{s}_t - \mathbf{x} \qquad \mathbf{h}_t = \mathbf{r}_t - \mathbf{x} \qquad (148)$$

where $\mathbf{r}_t$ and $\mathbf{s}_t$ are from (3)-(4). First, note that from the definition of $\mathbf{z}_t$, $\mathbf{y}$ and $\mathbf{q}_t$, we have

$$\mathbf{z}_t = \mathbf{w} - \mathbf{A}\mathbf{q}_t$$

Then, we can show that the left hand side of (146) is equivalent to

$$\begin{aligned}
&\lim_{N\to\infty}\frac{1}{N}\mathbf{q}_\tau^T\mathbf{A}^T\mathbf{A}\mathbf{h}_t \\
&= \lim_{N\to\infty} -\frac{1}{N}(\mathbf{w} - \mathbf{A}\mathbf{q}_\tau)^T\mathbf{A}(\mathbf{r}_t - \mathbf{x}) + \frac{1}{N}\mathbf{w}^T\mathbf{A}\mathbf{h}_t \\
&\overset{a.s.}{=} \lim_{N\to\infty} -\frac{1}{N}\mathbf{z}_\tau^T\mathbf{A}(\mathbf{r}_t - \mathbf{x}) + v_w \qquad (149) \\
&= \lim_{N\to\infty} -\frac{1}{N}\mathbf{z}_\tau^T(\mathbf{A}\mathbf{r}_t - \mathbf{A}\mathbf{x} + \mathbf{w} - \mathbf{w}) + v_w \\
&= \lim_{N\to\infty} -\frac{1}{N}\mathbf{z}_\tau^T(\mathbf{A}\mathbf{r}_t - \mathbf{y}) + v_w - \frac{1}{N}\mathbf{z}_\tau^T\mathbf{w} \\
&\overset{a.s.}{=} \lim_{N\to\infty}\frac{1}{N}\mathbf{z}_\tau^T(\mathbf{y} - \mathbf{A}\mathbf{r}_t) + v_w - \delta v_w \qquad (150)
\end{aligned}$$

where (149) uses the result $\lim_{N\to\infty}\frac{1}{N}\mathbf{w}^T\mathbf{A}\mathbf{h}_t \overset{a.s.}{=} v_w$ from [10], while the last step is based on the fact that

$$\begin{aligned}
\lim_{N\to\infty}\frac{1}{N}\mathbf{w}^T\mathbf{z}_\tau &= \lim_{N\to\infty}\frac{1}{N}\mathbf{w}^T(\mathbf{w} - \mathbf{A}\mathbf{q}_\tau) \\
&\overset{a.s.}{=} \lim_{N\to\infty}\frac{1}{N}\mathbf{w}^T\mathbf{w} \overset{a.s.}{=} \delta v_w \qquad (151)
\end{aligned}$$

where we used the asymptotic orthogonality of $\mathbf{w}$ and $\mathbf{A}\mathbf{q}_\tau$ from (15). Finally, applying (150) to both sides of (146) gives (147).

REFERENCES

[1] S. Ramani, T. Blu, and M. Unser, "Monte-carlo sure: A black-box optimization of regularization parameters for general denoising algorithms," *IEEE transactions on image processing : a publication of the IEEE Signal Processing Society*, vol. 17, pp. 1540–54, 2008.

[2] S. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Transactions on Signal Processing*, vol. 41, no. 12, pp. 3397–3415, 1993.

[3] J. Tropp, "Greed is good: Algorithmic results for sparse approximation," *IEEE Transactions on Information Theory*, vol. 50, no. 10, pp. 2231–2242, 2004.

[4] T. Blumensath and M. Davies, "Iterative hard thresholding for compressed sensing," *Applied and Computational Harmonic Analysis*, vol. 27, no. 3, pp. 265–274, 2009.

[5] I. Daubechies, M. Defrise, and C. De Mol, "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint," *Communications on Pure and Applied Mathematics*, vol. 57, no. 11, pp. 1413–1457, 2004.

[6] D. Donoho., A. Maleki, and A. Montanari, "Message-passing algorithms for compressed sensing," *Proceedings of the National Academy of Sciences*, vol. 106, no. 45, pp. 18 914–18 919, 2009.

[7] J. Ma and L. Ping, "Orthogonal amp for compressed sensing with unitarily-invariant matrices," in *2016 IEEE Information Theory Workshop (ITW)*, 2016, pp. 280–284.

[8] S. Rangan, P. Schniter, and A. Fletcher, "Vector approximate message passing," *IEEE Transactions on Information Theory*, vol. 65, no. 10, pp. 6664–6684, 2019.

[9] K. Takeuchi and C. Wen, "Rigorous dynamics of expectation-propagation signal detection via the conjugate gradient method," in *2017 IEEE 18th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, 2017, pp. 1–5.

[10] N. Skuratovs and M. Davies, *Compressed sensing with upscaled vector approximate message passing*, 2020. arXiv: 2011.01369 [cs.IT].

[11] N. Skuratovs and M. Davies, "Upscaling vector approximate message passing," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 4757–4761.

[12] K. Takeuchi, "Convolutional approximate message-passing," *IEEE Signal Processing Letters*, vol. 27, pp. 416–420, 2020.

[13] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-d transform-domain collaborative filtering," *IEEE Transactions on Image Processing*, vol. 16, no. 8, pp. 2080–2095, 2007.

[14] A. Buades, B. Coll, and J. Morel, "A non-local algorithm for image denoising," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 2, 2005, 60–65 vol. 2.

[15] A. Fletcher, P. Pandit, S. Rangan, S. Sarkar, and P. Schniter, "Plug-in estimation in high-dimensional linear inverse problems: A rigorous analysis," in *Advances in Neural Information Processing Systems 31*, Curran Associates, Inc., 2018, pp. 7440–7449.

[16] C. Metzler, A. Maleki, and R. Baraniuk, "From denoising to compressed sensing," *IEEE Transactions on Information Theory*, vol. 62, no. 9, pp. 5117–5144, 2016.

[17] K. Takeuchi, "A unified framework of state evolution for message-passing algorithms," in *2019 IEEE International Symposium on Information Theory (ISIT)*, 2019, pp. 151–155.

[18] M. Bayati and A. Montanari, "The dynamics of message passing on dense graphs, with applications to compressed sensing," *IEEE Transactions on Information Theory*, vol. 57, no. 2, pp. 764–785, 2011.

[19] S. Sarkar, R. Ahmad, and P. Schniter, "Mri image recovery using damped denoising vector amp," in *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 8108–8112.

[20] Z. Xue, J. Ma, and X. Yuan, "D-oamp: A denoising-based signal recovery algorithm for compressed sensing," in *2016 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, 2016, pp. 267–271.

[21] P. Schniter, S. Rangan, and A. Fletcher, "Denoising based vector approximate message passing," *CoRR*, vol. abs/1611.01376, 2016.

[22] L. Liu, S. Huang, and B. Kurkoski, "Memory approximate message passing," *CoRR*, vol. abs/2106.02237, 2021. [Online]. Available: https://arxiv.org/abs/2106.02237.

[23] R. Couillet and M. Debbah, *Random Matrix Methods for Wireless Communications*. Cambridge University Press, 2011.

[24] R. Berthier, A. Montanari, and P. Nguyen, "State evolution for approximate message passing with non-separable functions," *Information and Inference: A Journal of the IMA*, vol. 9, no. 1, pp. 33–79, 2019.

[25] C. Stein, "Estimation of the mean of a multivariate normal distribution," *Ann. Statist.*, vol. 9, no. 6, pp. 1135–1151, 1981.

[26] K. Takeuchi, "Rigorous dynamics of expectation-propagation-based signal recovery from unitarily invariant measurements," in *2017 IEEE International Symposium on Information Theory (ISIT)*, 2017, pp. 501–505.

Nikolajs Skuratovs received the B.Eng degree in electrical engineering, in 2017, from the Transport and Telecommunication Institute (TSI), Riga, Latvia, and the M.Sc. degree in signal processing and communications, in 2018, from The University of Edinburgh (UoE), Edinburgh, U.K., where he is currently working towards the Ph.D. degree.

Mike E. Davies (Fellow, IEEE) received the M.A. degree in engineering from Cambridge University, Cambridge, U.K., in 1989, and the Ph.D. degree in nonlinear dynamics from University College London (UCL), London, U.K., in 1993.

He was the Head of the Institute for Digital Communications (IDCOM), The University of Edinburgh (UoE), Edinburgh, U.K., from 2013 to 2016. He holds the Jeffrey Collins Chair in signal and image processing at UoE, where he also leads the Edinburgh Compressed Sensing Research Group. He was awarded a Royal Society University Research Fellowship in 1993 and was a Texas Instruments Distinguished Visiting Professor at Rice University, Houston, TX, USA, in 2012. He leads the U.K. University Defence Research Collaboration (UDRC) Program on signal processing for defense in collaboration with the U.K. Defence Science and Technology Laboratory (DSTL). His research has focused on nonlinear time series, source separation, compressed sensing and computational imaging. He has also explored the application of these ideas to advanced medical imaging, RF sensing applications, and machine learning.

Prof. Davies has been elected as a fellow of EURASIP, the Royal Society of Edinburgh, and the Royal Academy of Engineering. He was a recipient of the European Research Council (ERC) Advanced Grant on Computational Sensing and the Royal Society Wolfson Research Merit award. He was awarded the Foundation Scholarship for the M.A. degree in 1987.