Exponentially Fast Concentration of Vector Approximate Message Passing to its State Evolution

Collin Cademartori
Columbia University, USA
Email: cac2301@columbia.edu

Cynthia Rush
Columbia University, USA
Email: cgr2130@columbia.edu

***Abstract*—Vector Approximate Message Passing is an iterative algorithm for computationally-efficient estimation in high-dimensional regression problems. Due to the presence of an 'Onsager' correction term in its iterates, for a wide class of $N \times M$ design matrices $\mathbf{A}$, namely those that are right orthogonally-invariant, the asymptotic distribution of the estimate at any iteration of the algorithm can be exactly characterized in the large system limit as $M/N \to \delta \in (0, \infty)$ via a scalar recursion referred to as state evolution. In this paper, we show that appropriate functionals of the iterates in fact concentrate around their limiting values predicted by these asymptotic distributions with rates exponentially fast in N.**

A full version of this paper is accessible at: http://www.columbia.edu/~cgr2130/pubs.html

I. Introduction

Approximate message passing (AMP) algorithms are a class of iterative methods for solving various high-dimensional statistical estimation and inference problems [1]–[5]. In this paper we focus specifically on the problem of high-dimensional regression. In particular, we consider estimating an unknown coefficient vector or signal $\mathbf{x}_0$ from models of the form

$$\mathbf{y} = \mathbf{A}\mathbf{x}_0 + \mathbf{w}, \qquad \mathbf{w} \overset{iid}{\sim} p(\mathbf{w}), \qquad \mathbf{x}_0 \overset{iid}{\sim} p(\mathbf{x}_0), \tag{1}$$

where $\mathbf{A} \in \mathbb{R}^{M \times N}$ with $M < N$ and $p(\mathbf{x}_0)$ and $p(\mathbf{w})$ are prior distributions on the signal and noise, respectively. AMP-style algorithms can accommodate a range of estimation procedures for the problem set-up in (1), including maximum a posteriori (MAP) and minimum mean squared error (MMSE) estimation.

AMP is the following two-step iteration. At iteration $k \geq 0$, the algorithm updates its estimate of the signal $\mathbf{x}_0$ with estimates $\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2, ...$ Initializing with $\mathbf{x}_{-1} = \mathbf{v}_{-1} = \mathbf{1}$, calculate

$$\hat{\mathbf{x}}_k = g_k(\mathbf{r}_k), \qquad \mathbf{r}_k = \hat{\mathbf{x}}_{k-1} + \mathbf{A}^T\mathbf{v}_{k-1}, \tag{2}$$

$$\mathbf{v}_k = \mathbf{y} - \mathbf{A}\hat{\mathbf{x}}_k + \frac{N}{M}\mathbf{v}_{k-1}\mathrm{div}[g_k(\mathbf{r}_k)], \tag{3}$$

where g_k is the so-called 'denoiser', an appropriately-chosen Lipschitz function depending on the estimation procedure to be performed, and $\mathrm{div}[g_k(\mathbf{r}_k)]$ is the divergence of the denoiser, measuring the sensitivity of g_k at its input. The $\mathbf{v}_k$ update in (3) can be interpreted as a corrected residual: the usual residual $\mathbf{y} - \mathbf{A}\hat{\mathbf{x}}_k$ with an 'Onsager correction', $\frac{N}{M}\mathbf{v}_{k-1}\mathrm{div}[g_k(\mathbf{r}_k)]$.

Due to the presence of the correction term in the residual step, it is possible under certain conditions to characterize the asymptotic distribution of $\mathbf{r}_k$ in the large system limit where $N \to \infty$ and $M/N \to \delta \in (0, \infty)$. In particular, for variances τ_k that can be characterized exactly by a scalar recursion referred to as state evolution, it can be shown that in this limit, the elements of the vectors $\mathbf{r}_k$ behave like samples from an $N(\mathbf{x}_0, \tau_k\mathbf{I}_N)$ distribution, where $\mathbf{x}_0$ is the true signal.

More formally, it was shown in [3] that for all loss functions ϕ in the class of pseudo-Lipschitz functions (defined in Section 3), empirical averages $\frac{1}{N}\sum_{i=1}^{N}\phi([\mathbf{r}_k]_i, [\mathbf{x}_0]_i)$ will converge to $\mathbb{E}\{\phi(X_0 + \tau_k Z, X_0)\}$ where $Z \sim N(0,1)$ independent of $X_0 \sim p(\mathbf{x}_0)$. In [6], this distributional convergence is refined with a finite sample analysis showing that these empirical averages exhibit concentration around their limits with rates exponential in N. However, such asymptotic guarantees for the AMP iterates have only been established when the elements of the design matrices $\mathbf{A}$ are i.i.d. sub-Gaussian [3], [7] and concentration results only for i.i.d. Gaussian elements.

Vector approximate message passing (VAMP) is an AMP-style algorithm, recently introduced in [8]. Remarkably, it was shown to admit an exact asymptotic characterization in the large system limit, analogous to that of AMP, but under a much larger class of random design matrices $\mathbf{A}$. In particular, this is the class of *right orthogonally invariant* $\mathbf{A}$, meaning that $\mathbf{AV}$ has the same distribution as $\mathbf{A}$ for any $M \times M$ orthogonal matrix $\mathbf{V}$. As we will see, this assumption represents a significant relaxation of the i.i.d. sub-Gaussian condition.

This work extends the VAMP asymptotic analysis to a finite sample analysis under nearly identical conditions. Structurally, the proof is based on an approach used for the finite sample analysis of AMP in [6], though there are critical differences in the required analysis for VAMP; these are detailed in Section IV-E. In short, we develop a suite of concentration of measure tools to prove that, as the algorithm runs, the output retains concentration around its expected values. The idea is that if the algorithm concentrates through iteration k, then we can prove that it will concentrate at iteration $k+1$, with only a slightly degraded rate for the concentration.

II. Vector Approximate Message Passing

VAMP is similar to that of AMP in (2)-(3), and, like AMP, can be derived as a quadratic approximation to a belief propagation algorithm associated to a particular factor graph. The reader is referred to [8] for such a derivation. Here, we provide some self-contained intuition for VAMP.

Algorithm 1 presents VAMP. We use

$$g_2(\mathbf{r}_{2k}, \gamma_{2k}) = (\gamma_w \mathbf{A}^T \mathbf{A} + \gamma_{2k} \mathbf{I})^{-1} (\gamma_w \mathbf{A}^T \mathbf{y} + \gamma_{2k} \mathbf{r}_{2k}), \quad (4)$$

where $\gamma_w = 1/\tau_w$ and $\tau_w < \infty$ is the elementwise variance of the noise $\mathbf{w}$. The function g_2 in (4) has divergence $\operatorname{div}[g_2(\mathbf{r}_{2k}, \gamma_{2k})] = \gamma_{2k} \operatorname{Tr}\left(\gamma_w \mathbf{A}^T \mathbf{A} + \gamma_{2k} \mathbf{I}\right)$. Analogous to the denoisers g_k in the AMP case, $g_1(\mathbf{r}_{1k}, \gamma_{1k})$ must be specified by the user, however now we have a single function g_1 with a continuous parameter $\gamma_{1k} > 0$, rather than an iteration-indexed collection of functions as in AMP.

Algorithm 1 VAMP

Require: Number of iterations K, design matrix $\mathbf{A}$, observed $\mathbf{y}$, and denoiser $g_1(\cdot, \gamma_{1k})$

1: Initialize $\mathbf{r}_{10}$ and $\gamma_{10} \geq 0$.
2: **for** $k \leftarrow 1, \ldots, K$ **do**
3: $\quad \hat{\mathbf{x}}_{1k} \leftarrow g_1(\mathbf{r}_{1k}, \gamma_{1k})$ and $\alpha_{1k} \leftarrow \operatorname{div}[g_1(\mathbf{r}_{1k}, \gamma_{1k})]$
4: $\quad \eta_{1k} \leftarrow \gamma_{1k}/\alpha_{1k}$ and $\gamma_{2k} \leftarrow \eta_{1k} - \gamma_{1k}$
5: $\quad \mathbf{r}_{2k} \leftarrow (\eta_{1k} \hat{\mathbf{x}}_{1k} - \gamma_{1k} \mathbf{r}_{1k}) / \gamma_{2k}$
6: $\quad \hat{\mathbf{x}}_{2k} \leftarrow g_2(\mathbf{r}_{2k}, \gamma_{2k})$ and $\alpha_{2k} \leftarrow \operatorname{div}[g_2(\mathbf{r}_{2k}, \gamma_{2k})]$
7: $\quad \eta_{2k} \leftarrow \gamma_{2k}/\alpha_{2k}$ and $\gamma_{1(k+1)} \leftarrow \eta_{2k} - \gamma_{2k}$
8: $\quad \mathbf{r}_{1(k+1)} \leftarrow (\eta_{2k} \hat{\mathbf{x}}_{2k} - \gamma_{2k} \mathbf{r}_{2k}) / \gamma_{1k}$
9: **end for**

The choice of g_1, and its relationship to g_2, can be better understood by examining VAMP with Gaussian noise $\mathbf{w} \sim N(\mathbf{0}, \tau_w \mathbf{I}_M)$. As described above, we can view VAMP as trying to estimate some summary (e.g. the mode or mean) of the posterior distribution, which is proportion to the product of the data likelihood and the signal prior, given by

$$p(\mathbf{x}_0 \mid \mathbf{y}) \propto f\left(\mathbf{y} \mid \mathbf{A}\mathbf{x}_0, \gamma_w^{-1} \mathbf{I}\right) \cdot \prod_{i=1}^{N} p([\mathbf{x}_0]_i),$$

where $f(\mathbf{y} \mid \boldsymbol{\mu}, \boldsymbol{\Sigma})$ is a multivariate normal density evaluated at $\mathbf{y}$ having mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$. Recall from (1) that we assume $\mathbf{x}_0$ has i.i.d. elements. At each iteration, VAMP replaces the problem of computing a posterior summary with two easier ones. First, observe that $f(\mathbf{y} \mid \mathbf{A}\mathbf{x}_0, \gamma_w^{-1}\mathbf{I})$, as a function of $[\mathbf{x}_0]_i$, is proportional to a normal density $f([\mathbf{x}_0]_i \mid [\mathbf{r}_1]_i, [\boldsymbol{\tau}_1]_i)$. Making the simplifying assumption that $[\boldsymbol{\tau}_1]_i$ are equal across i, then this yields the approximate posterior

$$f(\mathbf{x}_0 \mid \mathbf{r}_1, \tau_1 \mathbf{I}) \cdot \prod_{i=1}^{N} p([\mathbf{x}_0]_i). \quad (5)$$

If we instead approximate the priors $p([\mathbf{x}_0]_i)$ by independent normal distributions with means $[\mathbf{r}_2]_i$ and equal variances τ_2, then we get the approximate posterior

$$f\left(\mathbf{y} \mid \mathbf{A}\mathbf{x}_0, \gamma_w^{-1}\mathbf{I}\right) \cdot f\left(\mathbf{x}_0 \mid \mathbf{r}_2, \tau_2 \mathbf{I}\right). \quad (6)$$

In each iteration, VAMP uses both approximate posteriors to update the estimate of $\mathbf{x}_0$. Estimating $\mathbf{x}_0$ by posterior (5) requires specifying $\mathbf{r}_1$ and τ_1, which control the approximate likelihood of the data. Since approximation (6) uses the true likelihood, VAMP uses the estimate from (6) to update $\mathbf{r}_1$ and τ_1 in (5). Similarly, the estimates from (5) (which use the true prior) are used to update parameters $\mathbf{r}_2$ and τ_2 for the approximate prior in (6). By iterating these steps, VAMP uses past estimates to improve both approximations, and then uses the improved approximations to further improve our estimates.

In this context, g_1 and g_2 perform estimation of the two approximate models. The definitions of these functions depend on both the signal prior and desired posterior summary. In (6), the prior is modeled as normal regardless of the true prior. Since the resulting approximate posterior is again normal, and the MMSE and MAP estimates are identical, (4) is the natural choice for g_2, as it calculates the mean of (6). However, (5) depends on the choice of prior $p(\mathbf{x}_0)$ and the desired summary, so g_1 must be specified by the user accordingly.

III. Main Result

A. Empirical Convergence and Concentration

Before stating our main concentration result for VAMP, we first revisit the notion of empirical convergence. We say that a sequence of vectors $\{\mathbf{v}_n\}_{n=0}^{\infty} \subset \mathbb{R}^J$ for some $J \geq 1$ converges empirically (with 2nd moment) to a random vector $\mathbf{V} \in \mathbb{R}^J$ if $\mathbb{E}\{[\mathbf{V}]_j^2\} < \infty$ for all $1 \leq j \leq J$ and if $\frac{1}{N}\sum_{n=0}^{N} \phi(\mathbf{v}_n) \to \mathbb{E}\{\phi(\mathbf{V})\}$, almost surely for any $\phi : \mathbb{R}^J \to \mathbb{R}$ that is pseudo-Lipschitz[1] of order 2, denoted $\phi \in \mathrm{PL}(2)$. In what follows, we will simply write that a vector sequence converges empirically to some random variable, without explicitly stating that the random variable has a finite 2nd moment.

Next we give the assumptions we need for our main result.

Assumption 0. The truth $\mathbf{x}_0$ converges empirically to a random variable X_0, the singular values s_i of $\mathbf{A}$ converge empirically to a bounded random variable S, and the initial precision estimate γ_{10} converges to non-negative $\overline{\gamma}_{1k}$.

Assumption 1. The design matrix $\mathbf{A}$ is right orthogonally invariant. If $\mathbf{A}$ has singular value decomposition $\mathbf{USV}^T$, then this is equivalent to $\mathbf{V}$ being Haar distributed on the group of $M \times M$ orthogonal matrices, denoted $O(M)$. In other words, $\mathbf{VV}' \stackrel{d}{=} \mathbf{V}$ for any other $\mathbf{V}' \in O(M)$.

Assumption 2. The priors $p(\mathbf{x}_0)$ and $p(\mathbf{w})$ on the signal and noise are sub-Gaussian, and $p(\mathbf{x}_0)$ isstrongly log-concave.

Assumption 3. The estimating functions $g_1$are separable[2], and both g_1 and its derivative are uniformly Lipschitz[3] at $\overline{\gamma}_{1k}$ for all $k \geq 0$. When g_1 is the MAP or MMSE estimator for the model (5), it is separable. Henceforth, we will slightly abuse notation by writing g_ℓ as taking both vector and scalar input.

Assumption 4. The γ_{ik} are clipped so that they lie in some interval $[\gamma_{\min}, \gamma_{\max}]$ with $0 < \gamma_{\min} < \gamma_{\max} < \infty$. This prevents the iteration from returning invalid values of α_{ik} and thus aids with convergence.

Assumption 5. We terminate the algorithm according to stopping criteria that are given explicitly in the next section.

[1] A function $\phi : \mathbb{R}^J \to \mathbb{R}$ is in $\mathrm{PL}(2)$ if, for vectors $\mathbf{v}, \mathbf{v}' \in \mathbb{R}^J$ and some constant $L \geq 0$, it satisfies $|\phi(\mathbf{v}) - \phi(\mathbf{v}')| \leq L\|\mathbf{v} - \mathbf{v}'\|\left(1 + \|\mathbf{v}\| + \|\mathbf{v}'\|\right)$.

[2] A function $g : \mathbb{R}^J \to \mathbb{R}$ is *separable* if for $\mathbf{v} \in \mathbb{R}^J$, there exist a scalar function $\widetilde{g} : \mathbb{R} \to \mathbb{R}$ for which $[g(\mathbf{v})]_i = \widetilde{g}([\mathbf{v}]_i)$.

[3] A function $\phi(\mathbf{w}, c)$ is uniformly Lipschitz at c_0 if there is an open neighborhood U of c_0 and a constant $L > 0$ such that $\phi(\cdot, c)$ is $L-$Lipschitz for any fixed $c \in U$, and such that $|\phi(\mathbf{w}, c_1) - \phi(\mathbf{w}, c_2)| \leq L\left(1 + \|\mathbf{w}\|\right)|c_1 - c_2|$.

Essentially they imply that we stop if the MSE of our current estimate is sufficiently small or if there is a sufficiently small change in the estimate between successive iterations.

Under slightly milder conditions, Rangan *et al.* show empirical convergence of the vector sequence

$$\{([\hat{\mathbf{x}}_{1k}]_i, [\mathbf{r}_{1k}]_i, [\mathbf{x}_0]_i)\}_{i=1}^N,$$

to $(\hat{X}_{1k}, R_{1k}, X_0)$ where $R_{1k} = X_0 + \sqrt{\tau_{1k}}Z$ with $\hat{X}_{1k} = g_1(R_{1k}, \overline{\gamma}_{1k})$ and $Z \sim N(0,1)$ independent of X_0. Our main result, Theorem 1, characterizes the asymptotic rate of this empirical convergence.

Theorem 1. *Under the assumptions given above, for any* $\phi : \mathbb{R}^2 \to \mathbb{R}$ *with* $\phi \in \mathrm{PL}(2)$ *and any* $k \geq 0$*,*

$$\mathbb{P}\Big(\Big|\frac{1}{N}\sum_{i=1}^N \phi([\hat{\mathbf{x}}_{1k}]_i, [\mathbf{x}_0]_i) - \mathbb{E}\{\phi(\hat{X}_{1k}, X_0)\}\Big| \geq \epsilon\Big) \leq CC_k e^{-cc_k N\epsilon^2},$$

where C, c, C_k*, and* c_k *are universal constants not depending on* ϵ *or* N*, with* C_k *and* c_k *depending on* k *(but not explicitly specified), and* $\hat{X}_{1k}$ *is defined just above.*

The proof of Theorem 1 is given in Section IV-D. It requires a more general convergence result given in Section IV-C. The constants τ_{1k} and $\overline{\gamma}_{1k}$ that describe the limiting variable $\hat{X}_{1k}$ can be characterized exactly by the state evolution equations for VAMP, which we turn our attention to now.

B. State Evolution

To specify the state evolution, we must define sensitivity and error functions for each of the approximate models (5) and (6). Following [8], the sensitivity functions are defined as

$$\begin{aligned} A_1(\gamma_1, \tau_1) &= \mathbb{E}\{g_1'(R_1, \gamma_1)\}, \\ A_2(\gamma_2, \tau_2) &= \lim_{N\to\infty} \frac{\gamma_2}{N}\mathrm{Tr}\left[(\gamma_w \mathbf{A}^T\mathbf{A} + \gamma_2\mathbf{I})^{-1}\right], \end{aligned} \quad (7)$$

where $R_1 \sim N(X_0, \tau_1)$, and the error functions are defined as

$$\begin{aligned} \mathcal{E}_1(\gamma_1, \tau_1) &= \mathbb{E}\left(g_1(R_1, \gamma_1) - X_0\right)^2, \\ \mathcal{E}_2(\gamma_2, \tau_2) &= \lim_{N\to\infty} \frac{1}{N}\mathbb{E}\left\|g_2(\mathbf{r}_2, \gamma_2) - \mathbf{x}_0\right\|^2, \end{aligned} \quad (8)$$

where $\mathbf{r}_2 \sim N(\mathbf{x}_0, \tau_2\mathbf{I})$ and $\mathbf{y} \sim N\left(\mathbf{A}\mathbf{x}_0, \gamma_w^{-1}\mathbf{I}\right)$. Note that the limits in (7) and (8) exist since the limiting expressions only depend on the singular values of $\mathbf{A}$ (as shown in [8]), and these are assumed to converge empirically. With these functions defined, we can give the state evolution as follows.

$$\begin{aligned} &\overline{\alpha}_{1k} = A_1(\overline{\gamma}_{1k}, \tau_{1k}), \quad \overline{\eta}_{1k} = \overline{\gamma}_{1k}/\overline{\alpha}_{1k}, \\ &\overline{\gamma}_{2k} = \overline{\eta}_{1k} - \overline{\gamma}_{1k}, \quad \tau_{2k} = \frac{\mathcal{E}_1(\overline{\gamma}_{1k}, \tau_{1k} - \overline{\alpha}_{1k}^2\tau_{1k})}{(1-\overline{\alpha}_{1k})^2}, \\ &\overline{\alpha}_{2k} = A_2(\overline{\gamma}_{2k}, \tau_{2k}), \quad \overline{\eta}_{2k} = \overline{\gamma}_{2k}/\overline{\alpha}_{2k}, \\ &\overline{\gamma}_{1(k+1)} = \overline{\eta}_{2k} - \overline{\gamma}_{2k}, \quad \tau_{1(k+1)} = \frac{\mathcal{E}_1(\overline{\gamma}_{2k}, \tau_{2k} - \overline{\alpha}_{2k}^2\tau_{2k})}{(1-\overline{\alpha}_{2k})^2}. \end{aligned}$$

These equations are initialized with the assumed first iteration limiting precision γ_{10} and $\tau_{10} = \mathbb{E}\{(R_{10} - X_0)^2\}$.

IV. A General Concentration Result

Theorem 1 is a consequence of a concentration result for a more general iteration, Algorithm 2. Relative to the large system behavior of the iterates, the primary innovation of VAMP can be understood in terms of the fact that it is a special case of this iteration. We do not explicitly state the correspondence between the two algorithms, but refer the reader to [8, Appendix G] for details. For simplicity, we assume that $\mathbf{A} \in \mathbb{R}^{N\times N}$ without loss of generality since, as noted in [8], we can always achieve this by padding the singular values of $\mathbf{A}$ with zeros. Notice that expressing $\mathbf{A}$ with it singular value decomposition, $\mathbf{A} = \mathbf{U}\mathbf{S}\mathbf{V}^T$ allows for the effect of $\mathbf{A}$ to be broken up within the general algorithm, isolating the effects of the singular values $\mathbf{S}$ and the effects of the orthogonal matrix $\mathbf{V}$ in various stages.

Algorithm 2 General Algorithm

Require: Orthogonal matrix $\mathbf{V} \in \mathbb{R}^{N\times N}$, separable denoisers f_p, f_q, divergence functions C_i, parameter update functions Γ_i, initial data $\mathbf{u}_0$, and disturbance vectors $\mathbf{w}^p, \mathbf{w}^q$.

1: $\mathbf{p}_k \leftarrow \mathbf{V}\mathbf{u}_k$
2: $\alpha_{1k} \leftarrow \mathrm{div}[f_p(\mathbf{p}_k, \mathbf{w}^p, \gamma_{1k})]$ and $\gamma_{2k} \leftarrow \Gamma_1(\gamma_{1k}, \alpha_{1k})$
3: $\mathbf{v}_k \leftarrow C_1(\alpha_{1k})\left[f_p(\mathbf{p}_k, \mathbf{w}^p, \gamma_{1k}) - \alpha_{1k}\mathbf{p}_k\right]$
4: $\mathbf{q}_k \leftarrow \mathbf{V}^T\mathbf{v}_k$
5: $\alpha_{2k} \leftarrow \mathrm{div}[f_q(\mathbf{q}_k, \mathbf{w}^q, \gamma_{2k})]$ and $\gamma_{1(k+1)} \leftarrow \Gamma_2(\gamma_{2k}, \alpha_{2k})$
6: $\mathbf{u}_{k+1} \leftarrow C_2(\alpha_{2k})\left[f_q(\mathbf{q}_k, \mathbf{w}^q, \gamma_{2k}) - \alpha_{2k}\mathbf{q}_k\right]$

The effect of the singular values can be entirely subsumed within the denoiser f_q through the disturbance vector $\mathbf{w}^q$. Thus, we do not require any distributional assumptions about the singular values beyond empirical convergence to bounded random variables. Moreover, we clearly do not require any distributional assumptions on $\mathbf{U}$ to analyze VAMP. Thus, the large system behavior of VAMP can be characterized using only distributional assumptions on $\mathbf{V}$.

A. Notation

Before giving the concentration guarantees for Algorithm 2, we introduce some necessary notation. Our result will imply that iterates $(\mathbf{p}_1, \ldots, \mathbf{p}_k)$ and $(\mathbf{q}_1, \ldots, \mathbf{q}_k)$ converge empirically to mean-zero Gaussian vectors with variances τ_{1j} and τ_{2j}, respectively, for $1 \leq j \leq k$. We denote these vectors $(P_1, \ldots, P_k)$ and $(Q_1, \ldots, Q_k)$. It follows that $\mathbf{u}_j$ and $\mathbf{v}_j$ also convergence empirically to Gaussian U_j and V_j as well.

The covariance matrices of these Gaussian vectors are denoted $\mathbf{\Sigma}_k^p$, $\mathbf{\Sigma}_k^q$, $\mathbf{\Sigma}_k^u$, and $\mathbf{\Sigma}_k^v$, respectively. Define the vector $\mathbf{b}_k^p = (\mathbb{E}[P_1P_k], \ldots, \mathbb{E}[P_{k-1}P_k])$, and $\mathbf{b}_k^q$, $\mathbf{b}_k^u$, and $\mathbf{b}_k^v$ analogously. By [8, Theorem 4], the P_k can be decomposed as

$$P_k \overset{d}{=} \sum_{j=1}^{k-1}[\boldsymbol{\beta}_{1k}]_j P_j + \sqrt{\rho_{1k}}Z \quad (9)$$

where Z is a standard Gaussian variable, independent of $(P_1, \ldots, P_{k-1})$, and the coefficients are given by

$$\boldsymbol{\beta}_{1k} = [\mathbf{\Sigma}_{k-1}^p]^{-1}\mathbf{b}_k^p, \text{ and } \rho_{1k} = \mathbb{E}\{P_k^2\} - (\mathbf{b}_k^p)^T[\mathbf{\Sigma}_k^p]^{-1}\mathbf{b}_k^p. \quad (10)$$

The variance ρ_{1k} represents how much additional randomness is introduced in iterate $\mathbf{p}_k$. Similarly, we have coefficients $\boldsymbol{\beta}_{2k}$ and variances ρ_{2k} which play an analogous role for Q_k.

Define the matrix $\mathbf{U}_k$ having columns u_i for $0 \leq i \leq k$, and define $\mathbf{V}_k$, $\mathbf{P}_k$, and $\mathbf{Q}_k$ analogously. Define the matrices $\mathbf{A}_k = [\mathbf{P}_k \ \mathbf{V}_{k-1}]$ and $\mathbf{B}_k = [\mathbf{U}_k \ \mathbf{Q}_{k-1}]$, as well as

$$\mathbf{C}_k = [\mathbf{P}_{k-1} \ \mathbf{V}_{k-1}], \text{ and } \mathbf{D}_k = [\mathbf{U}_{k-1} \ \mathbf{Q}_{k-1}]. \tag{11}$$

For a matrix $\mathbf{M}$, let $\mathsf{P}_{\mathbf{M}}$ be the projection onto the column space of $\mathbf{M}$ and $\mathsf{P}_{\mathbf{M}^\perp} = \mathbf{I} - \mathsf{P}_{\mathbf{M}}$. Finally, define the sigma-algebras generated by the columns of these matrices as

$$G_k = \sigma\{\mathbf{U}_k, \mathbf{P}_k, \mathbf{V}_k, \mathbf{Q}_{k-1}\}, \ H_{k+1} = \sigma\{\mathbf{U}_{k+1}, \mathbf{P}_k, \mathbf{V}_k, \mathbf{Q}_k\}. \tag{12}$$

B. Assumptions

To prove our general convergence result, we make a number of assumptions on the quantities in Algorithm 2. These are broadly similar to those we made for Theorem 1 above.

Assumption 0'. The initial input $\mathbf{u}_0$ converges empirically to U_0, and γ_{10} converges to $\overline{\gamma}_{10} \geq 0$ as $N \to \infty$.

Assumptions 1' & 2'. Same as above.

Assumption 3'. The denoisers f_p and f_q are separable, and both them and their derivatives are uniformly Lipschitz at each $\overline{\gamma}_{ik}$ for $k \geq 1$, and $i = 1, 2$. In the translation between the general recursion and VAMP, the scalar components of f_q are defined as $f_q(q_i, (\xi_i, s_i), \gamma) = \frac{\gamma_w s_i \xi_i + \gamma q_i}{\gamma_w s_i^2 + \gamma}$, where $\boldsymbol{\xi} = \mathbf{U}^T \mathbf{w}$ (see [8, Appendix G]), which is separable by definition and uniformly Lipschitz. Also, we take $f_p(\mathbf{p}, \mathbf{x}_0, \gamma) = g_1(\mathbf{p} + \mathbf{x}_0, \gamma) - \mathbf{x}_0$, which is separable and uniformly Lipschitz when g_1 is, which is guaranteed by Assumption 3 above.

Assumption 4'. The functions $C_1, C_2, \Gamma_1, \Gamma_2$ are Lipschitz continuous and bounded over their domains. The log-concavity of $p(\mathbf{x}_0)$ (Assumption 2) and the clipping of γ_{1k} (Assumption 4) imply that $g_1'(\mathbf{r}_1, \gamma_1) \in [a, b]$ for some $0 < a < b < 1$ and all $\mathbf{r}_1$ and $\gamma_1 > 0$. Similarly for g_2'. This implies that $C_1, C_2, \Gamma_1, \Gamma_2$ have compact domains. Since they are also continuously differentiable, they are Lipschitz and bounded.

Assumption 5'. Following [6], we define stopping criteria that determine when the algorithm has effectively converged. First, we stop the iteration if $\tau_{1k} < \epsilon_1$ or $\tau_{2k} < \epsilon_2$. In terms Algorithm 1, this is equivalent to stopping the algorithm when the variance of $|R_{1k} - X_0|$ or $|R_{2k} - X_0|$ is sufficiently small. Next, we stop if $\rho_{1k} < \delta_1$ or $\rho_{2k} < \delta_2$. By the interpretation of ρ_{1k} and ρ_{2k} from the last section, this condition stops the algorithm if the difference in the (asymptotic distributions of) successive iterates is sufficiently small.

C. General Algorithm Concentration

Under these assumptions, we state our general concentration result for Algorithm 2 in Lemma 1 and Lemma 2. The proofs can be found in a longer version of this paper [9].

First, we introduce deviance terms $\boldsymbol{\Delta}_k^p$ and $\boldsymbol{\Delta}_k^q$ to quantify the discrepancy between the finite sample behavior of $(\mathbf{p}_1, \ldots, \mathbf{p}_k)$ and $(\mathbf{q}_1, \ldots, \mathbf{q}_k)$ and the limiting behavior (e.g. in (9)). This is done, in the following lemma, by studying the distributions of the vectors conditional on the previous output of the algorithm summarized by the sigma-algebras in (12). We state the lemma only for $\mathbf{p}_k$ conditional on G_k, but an analogous statement holds for $\mathbf{q}_k$ conditional on H_k as well.

Lemma 1. *Conditioning on the sigma-algebras in* (12), $\mathbf{p}_0|_{G_0} \overset{d}{=} \sqrt{\rho_{10}}\mathbf{Z}_{10} + \boldsymbol{\Delta}_0^p$, *and* $\mathbf{p}_k|_{G_k} \overset{d}{=} \sum_{\ell=0}^{k-1}[\boldsymbol{\beta}_{1k}]_\ell \mathbf{p}_\ell + \sqrt{\rho_{1k}}\mathbf{Z}_{1k} + \boldsymbol{\Delta}_k^p$, *where* $\boldsymbol{\beta}_{1k}$ *and* ρ_{1k} *are defined in* (10) *and*

$$\boldsymbol{\Delta}_0^p = \Big(\frac{\|\mathbf{u}_0\|}{\|\mathbf{Z}_{10}\|} - \sqrt{\rho_{10}}\Big)\mathbf{Z}_{10},$$

$$\boldsymbol{\Delta}_k^p = \mathbf{C}_k\Big((\mathbf{C}_k^T\mathbf{C}_k)^{-1}\mathbf{D}_k^T\mathbf{u}_k - \begin{bmatrix}\boldsymbol{\beta}_{1k} \\ \mathbf{0}\end{bmatrix}\Big), \\ + \Big(\frac{\|\mathsf{P}_{\mathbf{D}_k^\perp}\mathbf{u}_k\|}{\|\mathbf{Z}_{1k}\|} - \sqrt{\rho_{1k}}\Big)\mathbf{Z}_{1k} - \frac{\|\mathsf{P}_{\mathbf{D}_k^\perp}\mathbf{u}_k\|}{\|\mathbf{Z}_{1k}\|}\mathsf{P}_{\mathbf{C}_k}\mathbf{Z}_{1k}.$$

where $\mathbf{Z}_{1k} \in \mathbb{R}^M$ *are mutually independent standard Gaussian vectors, independent of* G_k *and* $\mathbf{C}_k, \mathbf{D}_k$ *are defined in* (11).

The proof of Lemma 1 is given in a longer version of this paper [9]. It requires studying the distribution of the Haar-distributed matrix $\mathbf{V}$ (treated as random) conditional on the sigma-algebras in (12). These results are given in [8, Appendix D], and essentially use both the rotational invariance of $\mathbf{V}$ and the fact that conditioning on the sigma-algebras is equivalent to conditioning on linear constraints. Now, using the conditional distributions in Lemma 1, we can state our main concentration result for the general recursion (Algorithm 2).

Lemma 2. *Throughout,* C *and* c *are generic universal constants not depending on* ϵ *or* N *and let* C_k *and* c_k *are universal constants depending on* k *(but are not explicitly specified). Let* $X_{N,k} \overset{..}{=} c$ *mean* $\mathbb{P}(|X_{N,k} - c| \geq \epsilon) \leq CC_k e^{-cc_k N\epsilon^2}$. *Then, for all* $k \geq 0$ *such that the stopping criteria are satisfied, we have the following concentration results.*

(a) *For deviation terms* $\boldsymbol{\Delta}_k^q$ *and* $\boldsymbol{\Delta}_k^p$ *defined in Lemma 1,*

$$\frac{1}{N}\|\boldsymbol{\Delta}_k^q\|^2 \overset{..}{=} 0, \text{ and } \frac{1}{N}\|\boldsymbol{\Delta}_k^p\|^2 \overset{..}{=} 0.$$

(b) *For* $\phi : \mathbb{R}^{k+2} \to \mathbb{R}$ *pseudo-Lipschitz,*

$$\frac{1}{N}\sum_{i=1}^N \phi([\mathbf{q}_1]_i, ..., [\mathbf{q}_{k+1}]_i, [\mathbf{w}^q]_i) \overset{..}{=} \mathbb{E}\{\phi(Q_0, ..., Q_k, W^q)\},$$

$$\frac{1}{N}\sum_{i=1}^N \phi([\mathbf{p}_0]_i, ..., [\mathbf{p}_k]_i, [\mathbf{w}^p]_i) \overset{..}{=} \mathbb{E}\{\phi(P_0, ..., P_k, W^p)\},$$

where the Q_i *are jointly Gaussian with* $Q_i \sim N(0, \tau_{2i})$ *and* $\mathbb{E}\{Q_iQ_j\} = [\boldsymbol{\Sigma}_{k+1}^q]_{ij}$, *and the* P_i *are jointly Gaussian with* $P_i \sim N(0, \tau_{1i})$ *and* $\mathbb{E}\{P_iP_j\} = [\boldsymbol{\Sigma}_k^p]_{ij}$.

The full proof is given in a longer version of this paper [9], but we sketch the main ideas here. In Section IV-D we use Lemma 2 to prove Theorem 1 and in Section IV-E we discuss the major differences between the proof of Lemma 2 and the corresponding AMP result given in [6].

The proof of Lemma 2 uses induction on the iteration k. From the definition of the deviation terms in Lemma 1, we see that establishing part **(a)** requires proving concentration to 0 for various expressions involving the VAMP iterates -

particularly projections, matrix products, and matrix inverses - which is ultimately related to establishing concentration for inner products of the VAMP iterates, $\mathbf{u}_k, \mathbf{p}_k, \mathbf{v}_k$, and $\mathbf{q}_k$. In the inductive step k, this is done by relating these values to pseudo-Lipschitz functions of the previous output at iterations $0, 1, \ldots, k-1$, and thus the arguments here ultimately boil down to applying part **(b)** with the inductive assumption.

Next part for **(b)**, we sketch the result for $\mathbf{p}_k$ and the result for $\mathbf{q}_k$ follows similarly. The proof uses two steps. First, note that it was proven in [8] that $(\mathbf{p}_1, \ldots, \mathbf{p}_k)$ converges empirically to a zero mean Gaussian vector given in (9). Now, if the vector $([\mathbf{p}_0]_i, \ldots, [\mathbf{p}_k]_i)$ is replaced by independent copies of this limiting variable, then the result follows just by properties of the normal distribution and pseudo-Lipschitz functions (see Lemma B.4 in [6]). Finishing the proof then requires controlling the difference between this asymptotic behavior and the finite sample behavior. It follows from Lemma 1 and the pseudo-Lipschitz property of ϕ that this can be controlled by showing that the deviation terms $\boldsymbol{\Delta}_k^p$ concentrate to 0 fast enough and this is done in part **(a)**.

D. Theorem 1 Proof Sketch

Theorem 1 follows from Lemma 2 part **(b)**. In translating VAMP (Algorithm 1) to the general iteration (Algorithm 2), we set $\mathbf{p}_{1k} = \mathbf{r}_{1k} - \mathbf{x}_0$ and $\mathbf{w}^p = \mathbf{x}_0$. Along with the definition of $\hat{\mathbf{x}}_{1k}$, it follows $\phi([\hat{\mathbf{x}}_{1k}]_i, [\mathbf{x}_0]_i) = \phi(g_1([\mathbf{p}_{1k}]_i + [\mathbf{w}^p]_i, \gamma_{1k}), [\mathbf{w}^p]_i)$. To shorten notation we let $\mathbf{p}_k := \mathbf{p}_{1k} + \mathbf{w}^p$. Using the pseudo-Lipschitz property of ϕ and the uniformly Lipchitz property of g_1, one can show that

$$\frac{1}{N}\sum_{i=1}^{N}\left|\phi(g_1([\mathbf{p}_k]_i, \gamma_{1k}), [\mathbf{w}^p]_i) - \phi(g_1([\mathbf{p}_k]_i, \overline{\gamma}_{1k}), [\mathbf{w}^p]_i)\right|$$

concentrates to 0. Then the desired result follows from showing that $\frac{1}{N}\sum_{i=1}^{N}\phi(g_1([\mathbf{p}_k]_i, \overline{\gamma}_{1k}), [\mathbf{w}^p]_i)$ concentrates to the desired limit. Since $g_1(\cdot, \overline{\gamma}_{1k})$ is Lipschitz, the composition $\phi(g_1(\mathbf{p}_k, \overline{\gamma}_{1k}), \mathbf{w}^p)$ is pseudo-Lipschitz, and therefore the desired concentration follows by Lemma 2 **(b)**.

To show that the sum above concentrates to 0, we use the pseudo-Lipschitz property of ϕ and the Triangle Inequality to upper bound the terms of the sum with the following:

$$L\Big[1 + \Big|g_1([\mathbf{p}_k]_i, \gamma_{1k}) - g_1([\mathbf{p}_k]_i, \overline{\gamma}_{1k})\Big| + 2\|(g_1([\mathbf{p}_k]_i, \overline{\gamma}_{1k}), \mathbf{w}_i^p)\|\Big] \cdot \Big|g_1([\mathbf{p}_k]_i, \gamma_{1k}) - g_1([\mathbf{p}_k]_i, \overline{\gamma}_{1k})\Big|,$$

with constant $L > 0$. Next, since g_1 is uniformly Lipschitz, $|g_1([\mathbf{p}_k]_i, \gamma_{1k}) - g_1([\mathbf{p}_k]_i, \overline{\gamma}_{1k})| \leq L(1 + |[\mathbf{p}_k]_i|)|\gamma_{1k} - \overline{\gamma}_{1k}|$. Plugging this into the above bound, and noting that products of Lipschitz functions are pseudo-Lipschitz, we get terms that are products of pseudo-Lipschitz functions with powers of $|\gamma_{1k} - \overline{\gamma}_{1k}|$. The former concentrates by Lemma 2 **(b)**, and the latter is shown to concentrate in the proof of Lemma 2 **(b)** in [9].

E. Lemma 2 Proof Discussion

Structurally, Lemma 2 is similar to [6, Lemma 5], which is used to establish the AMP analog of Theorem 1. However, there are a number of differences between the algorithms that lead to important differences in the proofs, discussed here.

(1) In AMP, we study vectors $\mathbf{Ax}$ where $\mathbf{A}$ has i.i.d. Gaussian entries and $\mathbf{x}$ is deterministic. For $\mathbf{Z}$ having i.i.d. $N(0,1)$ elements, $\mathbf{Ax} \overset{d}{=} \|\mathbf{x}\|\mathbf{Z}$, since $\|\mathbf{x}\|$ is deterministic. Importantly, the elements of $\|\mathbf{x}\|\mathbf{Z}$ are independent.

With VAMP, we study vectors $\mathbf{Vx}$ where $\mathbf{V}$ is uniformly distributed on the group of orthogonal matrices. This property of $\mathbf{V}$ implies that $\mathbf{Vx}$ has a rotationally invariant distribution. Then for $\mathbf{Z}$ having i.i.d. $N(0,1)$ elements, $\mathbf{Vx} \overset{d}{=} (\|\mathbf{x}\|/\|\mathbf{Z}\|)\mathbf{Z}$. Thus we pay for relaxing the Gaussianity condition by picking up a $\|\mathbf{Z}\|$ factor in the denominator, causing dependencies in the elements of $(\|\mathbf{x}\|/\|\mathbf{Z}\|)\mathbf{Z}$ that complicate the concentration arguments. By observing that the Gaussiantiy of $\mathbf{Z}$ gives concentration of $\|\mathbf{Z}\|/\sqrt{N}$ around 1, it can be shown that $\mathbf{Vx}$ will concentrate around the same limit as $(\|\mathbf{x}\|/\sqrt{N})\mathbf{Z}$, which essentially returns us to the AMP case.

(2) Because the parameters γ_{1k}, γ_{2k} can vary with N, so too can the behavior of the denoisers g_1, g_2. This is in contrast to AMP, where the denoisers can vary by iteration but not with N. In the concentration arguments, this must be accounted for by using the *uniformly* Lipschitz condition on f_p, f_q to control this additional source of variation.

(3) The order of denoising and subtracting the Onsager term is interchanged in AMP and VAMP. This adds an additional layer of complication in obtaining concentration for $\mathbf{v}_k$: from Algorithm 2, $\mathbf{v}_k = C_1(\alpha_{1k})\left[f_p(\mathbf{p}_k, \mathbf{w}^p, \gamma_{1k}) - \alpha_{1k}\mathbf{p}_k\right]$, with $f_p(\mathbf{p}_k, \mathbf{w}^p, \gamma_{1k}) - \alpha_{1k}\mathbf{p}_k$ failing to be Lipschitz (since α_{1k} also has a dependence on $\mathbf{p}_k$). This prevents us from simply applying Lemma 2 **(b)** when proving concentration results for $\mathbf{v}_k$. However, as with the $1/\|\mathbf{Z}\|$ dependence discussed in **(1)** above, we handle this by using concentration of α_{1k} to its limit $\overline{\alpha}_{1k}$, allowing oneto show that $\mathbf{v}_k$ will concentrate around the same limit as $C_1(\overline{\alpha}_{1k})\left[f_p(\mathbf{p}_k, \mathbf{w}^p, \gamma_{1k}) - \overline{\alpha}_{1k}\mathbf{p}_k\right]$. This function is Lipchitz, so (inductively) applying Lemma 2 **(b)** gives the desired concentration.

V. Conclusion and Future Work

This work presents rigorous non-asymptotic performance guarantees for VAMP characterized by the state evolution. We show that the probability of deviation from the state evolution predictions decay exponentially in the problem size N. A next step is to refine the bound of Theorem 1 to specify explicitly the iteration dependence in the universal constants C_k, c_k. This will be pursued in the journal version of this work.

We expect these concentration results to extend beyond the model (1) and assumptions considered in this work to other settings where the VAMP state evolution has been rigorously proved to characterize performance in the large system limit. For example, when the denoisers are non-separable [10], the distributional parameters of the noise and signal must be learned [11], or the generalized linear model is studied [12].

Acknowledgment

The authors acknowledge NSF CCF #1849883 support.

REFERENCES

[1] D. L. Donoho, A. Maleki, and A. Montanari, "Message passing algorithms for compressed sensing," *Proceedings of the National Academy of Sciences*, vol. 106 45, pp. 18 914–9, 2009.

[2] A. Montanari, "Graphical models concepts in compressed sensing," in *Compressed Sensing*, Y. C. Eldar and G. Kutyniok, Eds. Cambridge University Press, 2012, pp. 394–438. [Online]. Available: http://dx.doi.org/10.1017/CBO9780511794308.010

[3] M. Bayati and A. Montanari, "The dynamics of message passing on dense graphs, with applications to compressed sensing," *IEEE Trans. Inf. Theory*, pp. 764–785, 2011.

[4] F. Krzakala, M. Mézard, F. Sausset, Y. Sun, and L. Zdeborová, "Probabilistic reconstruction in compressed sensing: algorithms, phase diagrams, and threshold achieving matrices," *Journal of Statistical Mechanics: Theory and Experiment*, no. 8, 2012.

[5] S. Rangan, "Generalized approximate message passing for estimation with random linear mixing," *2011 IEEE International Symposium on Information Theory Proceedings*, pp. 2168–2172, 2010.

[6] C. Rush and R. Venkataramanan, "Finite sample analysis of approximate message passing algorithms," *IEEE Transactions on Information Theory*, vol. 64, pp. 7264–7286, 2016.

[7] M. Bayati, M. Lelarge, and A. Montanari, "Universality in polytope phase transitions and message passing algorithms," *Ann. Appl. Probability*, vol. 25, no. 2, pp. 753–822, Feb. 2015.

[8] S. Rangan, A. K. Fletcher, V. K. Goyal, E. Byrne, and P. Schniter, "Vector approximate message passing," *2017 IEEE International Symposium on Information Theory (ISIT)*, pp. 1588–1592, 2017.

[9] C. Cademartori and C. Rush, "Finite-sample analysis of Vector Approximate Message Passing," online: Available on http://www.columbia.edu/ cgr2130/pubs.html.

[10] A. Fletcher, M. Sahraee-Ardakan, S. Rangan, and P. Schniter, "Rigorous dynamics and consistent estimation in arbitrarily conditioned linear systems," *Advances in Neural Information Processing Systems*, pp. 7440–7449, 2018.

[11] A. Fletcher, P. Pandit, S. Rangan, S. Sarkar, and P. Schniter, "Plug-in estimation in high-dimensional linear inverse problems: A rigorous analysis," *Advances in Neural Information Processing Systems*, p. 2542–2551, 2017.

[12] P. Schniter, S. Rangan, and A. Fletcher, "Vector approximate message passing for the generalized linear model," *Proceedings of the 2016 50th Asilomar Conference on Signals, Systems and Computers*, 2017.