A Message-Passing Approach to Phase Retrieval of Sparse Signals

Philip Schniter[1] and Sundeep Rangan[2]

[1] Dept. of Electrical and Computer Engineering, The Ohio State University, Columbus, OH. schniter@ece.osu.edu
[2] Dept. of Electrical and Computer Engineering, Polytechnic Institute of New York University, Brooklyn, NY. srangan@poly.edu

Summary. In phase retrieval, the goal is to recover a signal $\boldsymbol{x} \in \mathbb{C}^N$ from the magnitudes of linear measurements $\boldsymbol{Ax} \in \mathbb{C}^M$. While recent theory has established that $M \approx 4N$ intensity measurements are necessary and sufficient to recover generic $\boldsymbol{x}$, there is great interest in reducing the number of measurements through the exploitation of sparse $\boldsymbol{x}$, which is known as compressive phase retrieval. In this work, we detail a novel, probabilistic approach to compressive phase retrieval based on the generalized approximate message passing (GAMP) algorithm. We then present a numerical study of the proposed PR-GAMP algorithm, demonstrating its excellent phase-transition behavior, robustness to noise, and runtime. For example, to successfully recover K-sparse signals, approximately $M \geq 2K \log_2(N/K)$ intensity measurements suffice when $K \ll N$ and $\boldsymbol{A}$ has i.i.d Gaussian entries. When recovering a 6k-sparse 65k-pixel grayscale image from 32k randomly masked and blurred Fourier intensity measurements, PR-GAMP achieved 99% success rate with a median runtime of only 12.6 seconds. Compared to the recently proposed CPRL, sparse-Fienup, and GESPAR algorithms, experiments show that PR-GAMP has a superior phase transition and orders-of-magnitude faster runtimes as the problem dimensions increase.

Key words: phase retrieval, compressed sensing, sparsity, belief propagation, message passing

1 Introduction

1.1 Phase retrieval

In phase retrieval, the goal is to recover a signal $\boldsymbol{x} \in \mathbb{C}^N$ from the *magnitudes* $y_m = |u_m|$ of possibly noisy linear measurements $\boldsymbol{u} = [u_1, \ldots, u_M]^{\mathsf{T}} = \boldsymbol{Ax} + \boldsymbol{w} \in \mathbb{C}^M$. This problem is motivated by the fact that it is often easier to build detectors (e.g., photographic plates or CCDs) that measure intensity rather than phase [1, 2]. Imaging applications of phase retrieval include X-ray diffraction imaging [3], X-ray crystallography [4, 5], array imaging [6], optics

[7], speckle imaging in astronomy [8], and microscopy [9]. Non-imaging applications include acoustics [10], interferometry [11], and quantum mechanics [12].

To reconstruct $\boldsymbol{x} \in \mathbb{C}^N$ (up to a global phase uncertainty), it has been recently established that $M \geq 4N - o(N)$ intensity measurements are necessary [13] and $M \geq 4N - 4$ are sufficient [14] through appropriate design of the linear transform $\boldsymbol{A}$. Meanwhile, to reconstruct $\boldsymbol{x} \in \mathbb{R}^N$ (up to a global sign uncertainty), it has been shown that $M \geq 2N - 1$ measurements are both necessary and sufficient [10]. However, there exist applications where far fewer measurements are available, such as sub-wavelength imaging [15, 16], Bragg sampling from periodic crystalline structures [17], and waveguide-based photonic devices [18]. To facilitate these *compressive* phase retrieval tasks, it has been proposed to exploit *sparsity*[3] in $\boldsymbol{x}$. In fact, very recent theory confirms the potential of this approach: to reconstruct K-sparse N-length $\boldsymbol{x}$ using a generic (e.g., i.i.d Gaussian) $\boldsymbol{A}$, only $M \geq 4K - 2$ intensity measurements suffice in the complex case and $M \geq 2K$ suffice in the real case (where $M \geq 2K$ is also necessary) when $K < N$ [19]. While these bounds are extremely encouraging, achieving them with a practical algorithm remains elusive.

To our knowledge, the first algorithm for compressive phase retrieval was proposed by Moravec, Romberg, and Baraniuk in [20] and worked by incorporating an ℓ_1-norm *constraint* into a traditional Fienup-style [1] iterative algorithm. However, this approach requires that the ℓ_1 norm of the true signal is known, which is rarely the case in practice. Recently, a more practical sparse-Fienup algorithm was proposed by Mukherjee and Seelamantula [21], which requires knowledge of only the signal sparsity K but is applicable only to measurement matrices $\boldsymbol{A}$ for which $\boldsymbol{A}^{\mathsf{H}}\boldsymbol{A} = \boldsymbol{I}$. Although this algorithm guarantees that the residual error $\|\boldsymbol{y} - |\boldsymbol{A}\widehat{\boldsymbol{x}}(t)|\|_2^2$ is non-increasing over the iterations t, it succumbs to local minima and, as we show in Section 4.4, is competitive only in the highly sparse regime.

To circumvent the local minima problem, Ohlsson, Yang, Dong, and Sastry proposed the *convex relaxation* known as Compressive Phase Retrieval via Lifting (CPRL) [22], which adds ℓ_1 regularization to the well-known PhaseLift algorithm [6, 23]. Both CPRL and PhaseLift "lift" the unknown vector $\boldsymbol{x} \in \mathbb{C}^N$ into the space of $N \times N$ rank-one matrices and solve a semidefinite program in the lifted space, requiring $O(N^3)$ complexity, which is impractical for practical image sizes N. Subsequent theoretical analysis [19] revealed that, while $M \gtrsim O(K^2 \log N)$ intensity measurements suffice for CPRL when $\boldsymbol{x} \in \mathbb{R}^N$, $M \gtrsim O(K^2/\log^2 N)$ measurements are *necessary*, which is disappointing because this greatly exceeds the $2K$ measurements that suffice for the optimal solver [19]. More recently, a cleverly initialized alternating minimization (AltMin) approach was proposed by Natrapalli, Jain, and Sanghavi in [24] that gives

[3] $\boldsymbol{x}$ may represent the sparse transform coefficients of a non-sparse signal-of-interest $\boldsymbol{s} = \boldsymbol{\Psi}\boldsymbol{x}$ in a sparsifying basis (or frame) $\boldsymbol{\Psi}$, in which case the intensity measurements would be $\boldsymbol{y} = |\boldsymbol{\Phi}\boldsymbol{s} + \boldsymbol{w}|$ and $\boldsymbol{A} \triangleq \boldsymbol{\Phi}\boldsymbol{\Psi}$.

CPRL-like guarantees/performance with only $O(NK^3)$ complexity, but this is still too complex for practical sparsities K (which tend to grow linearly with image size N).

Recently, Shechtman, Beck, and Eldar proposed the GrEedy Sparse PhAse Retrieval (GESPAR) algorithm [25], which applies fast 2-opt local search [26] to a sparsity constrained non-linear optimization formulation of the phase-retrieval problem. Numerical experiments (see Section 4.4) show that GESPAR handles higher sparsities K than the sparse-Fienup technique from [21], but at the cost of significantly increased runtime. In fact, due to the combinatorial nature of GESPAR's support optimization, its complexity scales rapidly in K, making it impractical for many problems of interest.

In this work, we describe a novel[4] approach to compressive retrieval that is based on loopy belief propagation and, in particular, the *generalized approximate message passing* (GAMP) algorithm from [29]. In addition to describing and deriving our phase-retrieval GAMP (PR-GAMP) algorithm, we present a detailed numerical study of its performance. For i.i.d Gaussian, Fourier, and masked-Fourier matrices $\boldsymbol{A}$, we demonstrate that PR-GAMP performs far better than existing compressive phase-retrieval algorithms in terms of both success rate and runtime for large values K and N. Interestingly, we find that PR-GAMP requires approximately $4\times$ the number of measurements as phase-oracle GAMP (i.e., GAMP given the magnitude-and-phase measurements $\boldsymbol{u} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{w}$), which generalizes what is known about phase retrieval of *non*-sparse signals in $\mathbb{C}^N$, where the ratio of magnitude-only to magnitude-and-phase measurements necessary and sufficient for perfect recovery is also 4 for large N [13, 14]. We also find that PR-GAMP is robust to additive noise, giving mean-squared error that is only 3 dB worse than phase-oracle GAMP over a wide SNR range.

Notation: For matrices, we use boldface capital letters like $\boldsymbol{A}$, and we use $\boldsymbol{A}^{\mathsf{T}}$, $\boldsymbol{A}^{\mathsf{H}}$, and $\|\boldsymbol{A}\|_F$ to denote the transpose, Hermitian transpose, and Frobenius norm, respectively. For vectors, we use boldface small letters like $\boldsymbol{x}$, and we use $\|\boldsymbol{x}\|_p = (\sum_n |x_n|^p)^{1/p}$ to denote the ℓ_p norm, with $x_n = [\boldsymbol{x}]_n$ representing the n^{th} element of $\boldsymbol{x}$. For random variable X, we write the pdf as $p_X(x)$, the expectation as $\mathrm{E}\{X\}$, and the variance as $\mathrm{var}\{X\}$. In some cases where it does not cause confusion, we drop the subscript on $p_X(x)$ and write the pdf simply as $p(x)$. For a circular-Gaussian random variable X with mean m and variance v, we write the pdf as $p_X(x) = \mathcal{N}(x; m, v) \triangleq \frac{1}{\pi v}\exp(-|x-m|^2/v)$. For the point mass at $x = 0$, we use the Dirac delta distribution $\delta(x)$. Finally, we use $\mathbb{R}$ for the real field, $\mathbb{C}$ for the complex field, $\mathrm{Re}\{x\}$ and $\mathrm{Im}\{x\}$ for the real and imaginary parts of x, and x^* for the complex conjugate of x.

[4] We previously described PR-GAMP in the conference paper [27] and the workshop presentation [28].

2 Background on GAMP

The approximate message passing (AMP) algorithm was recently proposed by Donoho, Maleki, and Montanari [30, 31] for the task estimating a signal vector $\boldsymbol{x} \in \mathbb{R}^N$ from linearly transformed and additive-Gaussian-noise corrupted measurements[5]

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{w} \in \mathbb{C}^M. \tag{1}$$

The Generalized-AMP (GAMP) algorithm proposed by Rangan [29] then extends the methodology of AMP to the generalized linear measurement model

$$\boldsymbol{y} = q(\boldsymbol{A}\boldsymbol{x} + \boldsymbol{w}) \in \mathbb{C}^M, \tag{2}$$

where $q(\cdot)$ is a component-wise nonlinearity. This nonlinearity affords the application of AMP to phase retrieval.

Both AMP and GAMP can be derived from the perspective of *belief propagation* [32], a Bayesian inference strategy that is based on a factorization of the signal posterior pdf $p(\boldsymbol{x}|\boldsymbol{y})$ into a product of simpler pdfs that, together, reveal the probabilistic structure in the problem. Concretely, if we model the signal coefficients in $\boldsymbol{x}$ and noise samples in $\boldsymbol{w}$ from (1)-(2) as statistically independent, so that $p(\boldsymbol{x}) = \prod_{n=1}^N p_{X_n}(x_n)$ and $p(\boldsymbol{y}|\boldsymbol{z}) = \prod_{m=1}^M p_{Y|Z}(y_m|z_m)$ for $\boldsymbol{z} \triangleq \boldsymbol{A}\boldsymbol{x}$, then we can factor the posterior pdf as

$$p(\boldsymbol{x}|\boldsymbol{y}) \propto p(\boldsymbol{y}|\boldsymbol{x})p(\boldsymbol{x}) \tag{3}$$

$$= \prod_{m=1}^M p_{Y|Z}\big(y_m\big|[\boldsymbol{A}\boldsymbol{x}]_m\big) \prod_{n=1}^N p_{X_n}(x_n), \tag{4}$$

yielding the factor graph in Fig. 2.

In belief propagation [32], beliefs about the unknown variables are passed among the nodes of the factor graph until all agree on a common set of beliefs. The set of beliefs passed into a given variable node are then used to determine the posterior pdf of that variable, or an approximation thereof. The sum-product algorithm [33] is perhaps the most well-known incarnation of belief propagation, wherein the messages take the form of pdfs and exact posteriors are guaranteed whenever the graph does not have loops. For graphs with loops, exact inference is known to be NP hard, and so loopy belief propagation (LBP) is not guaranteed to produce correct posteriors. Still, LBP has shown state-of-the-art performance on many problems in, e.g., decoding, computer vision, and compressive sensing [34].

The conventional wisdom surrounding LBP says that accurate inference is possible only when the circumference of the loops are relatively large. With

[5] Here and elsewhere, we use $\boldsymbol{y}$ when referring to the M measurements that are available for signal reconstruction. In the canonical (noisy) compressive sensing problem, the measurements take the form $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{w}$, but in the (noisy) compressive phase retrieval problem, the measurements instead take the form $\boldsymbol{y} = |\boldsymbol{A}\boldsymbol{x} + \boldsymbol{w}|$.

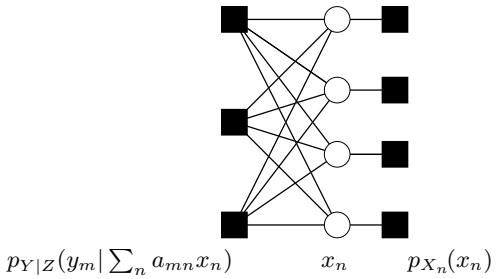

Fig. 1. GAMP factor graph, with white circles denoting random variables and black squares denoting pdf factors, for the case $M = 3$ and $N = 4$.

(1)-(2), this would require that $\boldsymbol{A}$ is a sparse matrix, which precludes most interesting cases of compressive inference, including compressive phase retrieval. Hence, the recent realization by Donoho, Maleki, Montanari, and Bayati that LBP-based compressive sensing is not only feasible [30] for *dense* matrices $\boldsymbol{A}$, but provably accurate [35, 36], was a breakthrough. In particular, they established that, in the large-system limit (i.e., as $M, N \to \infty$ with M/N fixed) and under i.i.d sub-Gaussian $\boldsymbol{A}$, the iterations of AMP are governed by a state-evolution whose fixed points describe the algorithm's performance. To derive the AMP algorithm, [30] proposed an ingenious set of message-passing approximations that become exact in the limit of large sub-Gaussian $\boldsymbol{A}$.

Remarkably, the "approximate message passing" (AMP) principles in [30]—including the state evolution—can be extended from the linear model (1) to the generalized linear model in (2), as established in [29]. The GAMP algorithm from [29] is summarized in Table 2, where $\mathcal{N}(z; \widehat{z}, \nu^z)$ is used to denote the circular-Gaussian pdf in variable z with mean $\widehat{z}$ and variance ν^z. In the sequel, we detail how GAMP, and some extensions of GAMP, allow us to tackle the phase retrieval problem.

3 Phase Retrieval GAMP

To apply the GAMP algorithm outlined in Table 2 to compressive phase retrieval, we specify a measurement likelihood function $p_{Y|Z}(y_m|\cdot)$ that models the lack of phase information in the observations y_m and a signal prior pdf $p_{X_n}(\cdot)$ that facilitates measurement compression, e.g., a sparsity-inducing pdf. In addition, we propose several modifications to the GAMP algorithm that aim to improve its robustness, and we propose an expectation-maximization method to learn the noise variance that parameterizes $p_{Y|Z}(y_m|\cdot)$.

3.1 Likelihood function

Before deriving the likelihood function $p_{Y|Z}(y_m|\cdot)$, we introduce some notation. First, we will denote the noiseless transform outputs by

input $\boldsymbol{A}, \{p_{X_n}(\cdot), \widehat{x}_n(1), \nu^x_n(1)\}_{n=1}^N, \{p_{Y|Z}(y_m|\cdot), \widehat{s}_m(0)\}_{m=1}^M$

define

$$p_{Z|Y,P}(z|y,\widehat{p};\nu^p) = \tfrac{p_{Y|Z}(y|z)\,\mathcal{N}(z;\widehat{p},\nu^p)}{\int_{z'} p_{Y|Z}(y|z')\,\mathcal{N}(z';\widehat{p},\nu^p)} \quad \text{(D1)}$$

$$g_{\mathsf{out},m}(\widehat{p},\nu^p) = \tfrac{1}{\nu^p}\big(\mathrm{E}_{Z|Y,P}\{Z|y_m,\widehat{p};\nu^p\} - \widehat{p}\big) \quad \text{(D2)}$$

$$g'_{\mathsf{out},m}(\widehat{p},\nu^p) = \tfrac{1}{\nu^p}\Big(\tfrac{\mathrm{var}_{Z|Y,P}\{Z|y_m,\widehat{p};\nu^p\}}{\nu^p} - 1\Big) \quad \text{(D3)}$$

$$p_{X_n|R_n}(x|\widehat{r};\nu^r) = \tfrac{p_{X_n}(x)\,\mathcal{N}(x;\widehat{r},\nu^r)}{\int_{x'} p_{X_n}(x')\,\mathcal{N}(x';\widehat{r},\nu^r)} \quad \text{(D4)}$$

$$g_{\mathsf{in},n}(\widehat{r},\nu^r) = \mathrm{E}_{X_n|R_n}\{X_n|\widehat{r};\nu^r\} \quad \text{(D5)}$$

$$g'_{\mathsf{in},n}(\widehat{r},\nu^r) = \mathrm{var}_{X_n|R_n}\{X_n|\widehat{r};\nu^r\} \quad \text{(D6)}$$

for $t = 1, 2, 3, \ldots, T_{\max}$

$$\forall m: \nu^p_m(t) = \textstyle\sum_{n=1}^N |a_{mn}|^2 \nu^x_n(t) \quad \text{(R1)}$$

$$\forall m: \widehat{p}_m(t) = \textstyle\sum_{n=1}^N a_{mn}\widehat{x}_n(t) - \nu^p_m(t)\,\widehat{s}_m(t-1) \quad \text{(R2)}$$

$$\forall m: \widehat{s}_m(t) = g_{\mathsf{out},m}(\widehat{p}_m(t), \nu^p_m(t)) \quad \text{(R3)}$$

$$\forall m: \nu^s_m(t) = -g'_{\mathsf{out},m}(\widehat{p}_m(t), \nu^p_m(t)) \quad \text{(R4)}$$

$$\forall n: \nu^r_n(t) = \big(\textstyle\sum_{m=1}^M |a_{mn}|^2 \nu^s_m(t)\big)^{-1} \quad \text{(R5)}$$

$$\forall n: \widehat{r}_n(t) = \widehat{x}_n(t) + \nu^r_n(t) \textstyle\sum_{m=1}^M a^*_{mn}\widehat{s}_m(t) \quad \text{(R6)}$$

$$\forall n: \nu^x_n(t+1) = \nu^r_n(t) g'_{\mathsf{in},n}(\widehat{r}_n(t), \nu^r_n(t)) \quad \text{(R7)}$$

$$\forall n: \widehat{x}_n(t+1) = g_{\mathsf{in},n}(\widehat{r}_n(t), \nu^r_n(t)) \quad \text{(R8)}$$

end

output $\{\widehat{x}_n(T_{\max}+1), \nu^x_n(T_{\max}+1)\}_{n=1}^N, \{\widehat{s}_m(T_{\max})\}_{m=1}^M$

Table 1. The GAMP Algorithm from [29] with $T_{\max}$ iterations.

$$z_m \triangleq \boldsymbol{a}_m^{\mathsf{H}}\boldsymbol{x} = |z_m| e^{j\phi_m} \text{ with } \phi_m \in [0, 2\pi), \tag{5}$$

where $\boldsymbol{a}_m^{\mathsf{H}}$ is the mth row of $\boldsymbol{A}$ and $j \triangleq \sqrt{-1}$. Next, we will assume the presence of additive noise w_m and denote the noisy transform outputs by

$$u_m \triangleq z_m + w_m = |u_m| e^{j\theta_m} \text{ with } \theta_m \in [0, 2\pi). \tag{6}$$

Our (noisy) intensity measurements are then

$$y_m = |u_m| \text{ for } m = 1, \ldots, M, \tag{7}$$

Henceforth, we assume additive white circular-Gaussian noise (AWGN) $w_m \sim \mathcal{N}(0, \nu^w)$. Thus, if we condition on z_m, then u_m is circular Gaussian with mean z_m and variance ν^w, and y_m is Rician with pdf [37]

$$p_{Y|Z}(y_m|z_m;\nu^w) = \frac{2y_m}{\nu^w} \exp\bigg(-\frac{y_m^2 + |z_m|^2}{\nu^w}\bigg) I_0\bigg(\frac{2y_m|z_m|}{\nu^w}\bigg) 1_{y_m \geq 0}, \tag{8}$$

where $I_0(\cdot)$ is the 0^{th}-order modified Bessel function of the first kind.

The functions $g_{\mathsf{out},m}(\cdot,\cdot)$ and $g'_{\mathsf{out},m}(\cdot,\cdot)$ defined in steps (D1)-(D3) of Table 2 can be computed using the expressions

$$\mathrm{E}_{Z|Y,P}\{Z|y_m,\widehat{p}_m;\nu^p_m\} = \frac{\int_{\mathbb{C}} z\, p_{Y|Z}(y_m|z;\nu^w)\mathcal{N}(z;\widehat{p}_m,\nu^p_m)dz}{\int_{\mathbb{C}} p_{Y|Z}(y_m|z';\nu^w)\mathcal{N}(z';\widehat{p}_m,\nu^p_m)dz'} \tag{9}$$

$$= \left(\frac{y_m}{1+\nu^w/\nu^p_m}R_0(\varrho_m) + \frac{|\widehat{p}_m|}{\nu^p_m/\nu^w+1}\right)\frac{\widehat{p}_m}{|\widehat{p}_m|} \tag{10}$$

and

$$\mathrm{var}_{Z|Y,P}\{Z|y_m,\widehat{p}_m;\nu^p_m\}$$

$$= \frac{\int_{\mathbb{C}} |z|^2\, p_{Y|Z}(y_m|z;\nu^w)\mathcal{N}(z;\widehat{p}_m,\nu^p_m)dz}{\int_{\mathbb{C}} p_{Y|Z}(y_m|z';\nu^w)\mathcal{N}(z';\widehat{p}_m,\nu^p_m)dz'} - |\mathrm{E}_{Z|Y,P}\{Z|y_m,\widehat{p}_m;\nu^p_m\}|^2 \tag{11}$$

$$= \frac{y_m^2}{(1+\nu^w/\nu^p_m)^2} + \frac{|\widehat{p}_m|^2}{(\nu^p_m/\nu^w+1)^2} + \frac{1+\varrho_m R_0(\varrho_m)}{1/\nu^w+1/\nu^p_m}$$
$$- |\mathrm{E}_{Z|Y,P}\{Z|y_m,\widehat{p}_m;\nu^p_m\}|^2, \tag{12}$$

where

$$R_0(\varrho_m) \triangleq \frac{I_1(\varrho_m)}{I_0(\varrho_m)} \text{ and } \varrho_m \triangleq \frac{2y_m\,|\widehat{p}_m|}{\nu^w+\nu^p_m}, \tag{13}$$

as shown in Appendix A.

3.2 EM update of the noise variance

Above, the noise variance ν^w was treated as a known parameter. In practice, however, ν^w may be unknown, in which case it is not clear what value to use in (10) and (12). To address this problem, we now describe how ν^w can be learned using an expectation-maximization (EM) [38] procedure. The methodology is similar to that proposed in [39] for the case of a Gaussian $p_{Y|Z}(y_m|\cdot)$, but the details differ due to the form of $p_{Y|Z}(y_m|\cdot)$ in (8).

Choosing $\boldsymbol{x}$ as the hidden data, the ith iteration EM update of the ν^w estimate is [38]

$$\widehat{\nu^w}[i+1] = \arg\max_{\nu^w\geq 0} \mathrm{E}\big\{\ln p(\boldsymbol{y},\boldsymbol{x};\nu^w)\big|\boldsymbol{y};\widehat{\nu^w}[i]\big\}, \tag{14}$$

where square brackets are used to distinguish EM iterations from GAMP iterations (recall Table 2). After a somewhat lengthy derivation, Appendix B shows that the EM update can be approximated as

$$\widehat{\nu^w}[i+1] \approx \frac{2}{M}\sum_{m=1}^{M}\big(y_m - |\boldsymbol{a}_m^{\mathsf{H}}\widehat{\boldsymbol{x}}[i]|\big)^2, \tag{15}$$

where $\widehat{\boldsymbol{x}}[i]$ denotes the posterior mean of $\boldsymbol{x}$ under the hypothesis $\nu^w = \widehat{\nu^w}[i]$. In practice, we use GAMP's estimate of the posterior mean (i.e., the GAMP output $\widehat{\boldsymbol{x}}(t)$ from Table 2 after the final GAMP iteration $t = T_{\max}$) in place of the true one, because computation of the latter is NP-hard in general [40].

3.3 Signal prior distributino

GAMP offers great flexibility with respect to the choice of prior distribution on the signal vector $\boldsymbol{x}$. In this work, we focus on separable priors, which have the form $p(\boldsymbol{x}) = \prod_{n=1}^{N} p_{X_n}(x_n)$ with arbitrary $p_{X_n}(\cdot)$ (recalling (4)), but we note that various forms of non-separable priors can be supported using the "turbo GAMP" formulation proposed in [41] or the "analysis GAMP" formulation proposed in [42].

For separable priors, $p_{X_n}(\cdot)$ should be chosen to reflect whatever form of probabilistic structure is known about coefficient x_n. For example, if $\boldsymbol{x} \in \mathbb{C}^N$ is known to be K-sparse, but nothing is know about the support, then it is typical to choose the Bernoulli-Gaussian (BG) model

$$p_{X_n}(x_n) = (1-\lambda)\delta(x_n) + \lambda \mathcal{N}(x_n; 0, \varphi), \tag{16}$$

with sparsity rate $\lambda = \frac{K}{N}$ and non-zero-coefficient variance φ that, if unknown, can be estimated from the observations via [39, eqn. (71)]

$$\varphi = \frac{\|\boldsymbol{y}\|_2^2 - M\nu^w}{\lambda \|\boldsymbol{A}\|_F^2}, \tag{17}$$

where $\|\cdot\|_F$ denotes the Frobenius norm. For this BG prior, expressions for the thresholding functions $g_{\mathsf{in},n}(\cdot,\cdot)$ and $g'_{\mathsf{in},n}(\cdot,\cdot)$ defined in steps (D5)-(D6) of Table 2 were given in [41]. When the sparsity rate λ in (16) is unknown, it can be learned using the EM-BG procedure described in [39]. In most cases, improved performance is obtained when a Gaussian mixture (GM) pdf is used in place of the Gaussian pdf in (16) [39].

Various extensions of the above are possible. For example, when all coefficients x_n are known to be real-valued or positive, the circular-Gaussian pdf in (16) should be replaced by a real-Gaussian or truncated-Gaussian pdf, respectively, or even a truncated-GM [43]. Furthermore, when certain coefficient subsets are known to be more or less sparse than others, a non-uniform sparsity [44] rate λ_n should be used in (16).

3.4 GAMP normalization and damping

To increase the numerical robustness of GAMP, it helps to normalize certain internal GAMP variables. To do this, we define $\alpha(t) \triangleq \frac{1}{M}\sum_{m=1}^{M} \nu_m^p(t)$ (which tends to grow very small with t at high SNR), normalize both $\widehat{s}_m(t)$ and $\nu_m^s(t)$ (which tend to grow very large) by $1/\alpha(t)$, and normalize $\nu_n^r(t)$ (which tends to grow very small) by $\alpha(t)$. The resulting GAMP iterations are shown in Table 3.4, with normalized quantities denoted by underbars. We note that, under infinite precision, these normalizations would cancel each other out and have no effect.

To reduce the chance of GAMP misconvergence, we find that it helps to "damp" the iterations. Damping helps to slow the algorithm using a stepsize

`for` $t=1,2,3,\ldots,T_{\max}$	
$\forall m: \nu^p_m(t)=\beta\sum_{n=1}^N \lvert a_{mn}\rvert^2\nu^x_n(t)+(1-\beta)\nu^p_m(t-1)$	(S1)
$\alpha(t)=\frac{1}{M}\sum_{m=1}^M \nu^p_m(t)$	(S2)
$\forall m: \widehat{p}_m(t)=\sum_{n=1}^N a_{mn}\widehat{x}_n(t)-\frac{\nu^p_m(t)}{\alpha(t)}\widehat{s}_m(t-1)$	(S3)
$\forall m: \widehat{s}_m(t)=\beta\alpha(t)g_{\mathsf{out},m}(\widehat{p}_m(t),\nu^p_m(t))+(1-\beta)\widehat{s}_m(t-1)$	(S4)
$\forall m: \underline{\nu}^s_m(t)=-\beta\alpha(t)g'_{\mathsf{out},m}(\widehat{p}_m(t),\nu^p_m(t))+(1-\beta)\underline{\nu}^s_m(t-1)$	(S5)
$\forall n: \underline{\nu}^r_n(t)=\big(\sum_{m=1}^M \lvert a_{mn}\rvert^2\underline{\nu}^s_m(t)\big)^{-1}$	(S6)
$\forall n: \bar{x}_n(t)=\beta\widehat{x}_n(t)+(1-\beta)\bar{x}_n(t-1)$	(S7)
$\forall n: \widehat{r}_n(t)=\bar{x}_n(t)+\underline{\nu}^r_n(t)\sum_{m=1}^M a^*_{mn}\widehat{s}_m(t)$	(S8)
$\forall n: \nu^x_n(t+1)=\alpha(t)\underline{\nu}^r_n(t)g'_{\mathsf{in},n}\big(\widehat{r}_n(t),\alpha(t)\underline{\nu}^r_n(t)\big)$	(S9)
$\forall n: \widehat{x}_n(t+1)=g_{\mathsf{in},n}\big(\widehat{r}_n(t),\alpha(t)\underline{\nu}^r_n(t)\big)$	(S10)
`end`	

Table 2. GAMP steps with variance normalization $\alpha(t)$ and damping parameter $\beta\in(0,1]$.

$\beta\in(0,1]$ that is incorporated into GAMP as shown in Table 3.4. Based on our experiments, the value $\beta=0.25$ seems to work well for phase retrieval. One consequence of the damping modification is the existence of additional state variables like $\bar{x}_n(t)$. To avoid the need to initialize these variables, we use $\beta=1$ during the first iteration. We note that the damping modifications described here are the ones included in the public domain GAMPmatlab implementation,[6] which differ slightly from the ones described in [45].

3.5 Avoiding bad local minima

As is well known, the compressive phase retrieval problem is plagued by bad local minima. We now propose methods to initialize and restart PR-GAMP that aims to avoid these local minima. Based on our experience (see Section 4), these methods are much more important for Fourier $\boldsymbol{A}$ than randomized (e.g., i.i.d Gaussian or masked-Fourier) $\boldsymbol{A}$.

GAMP initialization

The GAMP algorithm in Table 2 requires an initialization of the signal coefficient estimates $\{\widehat{x}_n(1)\}_{n=1}^N$, their variances $\{\nu^x_n(1)\}_{n=1}^N$, and the state variables $\{\widehat{s}_m(0)\}_{m=1}^M$ (which can be interpreted as Lagrange multipliers [45]). As recommended in [29], the standard procedure uses the fixed choices $\widehat{x}_n(1)=\mathrm{E}\{X_n\}$, $\nu^x_n(1)=\mathrm{var}\{X_n\}$, $\widehat{s}_m(0)=0$. For phase retrieval, we instead suggest to set each $\widehat{x}_n(1)$ using an independent draw of the random variable X_n and to set $\nu^x_n(1)=\frac{1}{N}\sum_{k=1}^N\lvert\widehat{x}_k(1)-\mathrm{E}\{X_k\}\rvert^2\ \forall n$. This initialization, however, only applies to the first EM iteration; for subsequent EM

[6] `http://sourceforge.net/projects/gampmatlab/`

iterations, GAMP should be warm-started using the outputs of the previous EM iteration.

EM initialization

For the EM algorithm described in Section 3.2, we must choose the initial noise-variance estimate $\widehat{\nu^w}[0]$. Even when accurate knowledge of ν^w is available, we find that setting $\widehat{\nu^w}[0]$ at a large value helps to avoid bad local minima. In particular, our empirical experience leads us to suggest setting $\widehat{\nu^w}[0]$ in correspondence with an initial SNR estimate of 10, i.e., $\widehat{\nu^w}[0] = \frac{\|\boldsymbol{y}\|_2^2}{M(\mathsf{SNR}_{\mathsf{init}}+1)}$ with $\mathsf{SNR}_{\mathsf{init}} = 10$.

Multiple restarts

To further facilitate the avoidance of bad local minima, we propose to run multiple attempts of EM-GAMP, each using a different random GAMP initialization (constructed as above). The attempt leading to the lowest normalized residual ($\mathsf{NR} \triangleq \|\boldsymbol{y} - |\boldsymbol{A}\widehat{\boldsymbol{x}}|\|_2^2/\|\boldsymbol{y}\|_2^2$) is then selected as the algorithm output. The efficacy of multiple attempts is numerically investigated in Section 4.

Furthermore, to avoid unnecessary restarts, we allow the algorithm to be stopped as soon as the NR drops below a user-defined stopping tolerance of $\mathsf{NR}_{\mathsf{stop}}$. When the true SNR is known, we suggest setting $\mathsf{NR}_{\mathsf{stop}}\mathsf{dB} = -(\mathsf{SNR}_{\mathsf{true}}\mathsf{dB} + 2)$.

Algorithm summary

The PR-GAMP algorithm is summarized in Table 3.5, where $A_{\max}$ controls the number of attempts, $\mathsf{SNR}_{\mathsf{init}}$ controls the initial SNR, and $\mathsf{NR}_{\mathsf{stop}}$ controls the stopping tolerance.

4 Numerical Results

In this section we numerically investigate the performance of PR-GAMP[7] under various scenarios and in comparison to several existing algorithms: Compressive Phase Retrieval via Lifting (CPRL) [22], GrEedy Sparse PhAse Retrieval (GESPAR) from [25], and the sparse Fienup technique from [21], As a benchmark, we also compare to "phase oracle" (PO) GAMP, i.e., GAMP operating on the magnitude-and-phase measurements $\boldsymbol{u} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{w}$ rather than on the intensity measurements $\boldsymbol{y} = |\boldsymbol{u}|$.

Unless otherwise noted, we generated random realizations the true signal vector $\boldsymbol{x}$ as K-sparse length-N with support chosen uniformly at random and

[7] PR-GAMP is part of the GAMPmatlab package at `http://sourceforge.net/projects/gampmatlab/`.

input $\boldsymbol{y}, \boldsymbol{A}, \{p_{X_n}(\cdot)\}_{n=1}^N, \mathsf{SNR}_{\mathsf{init}}, \mathsf{NR}_{\mathsf{stop}}$

$\widehat{\nu^w}[0] = \dfrac{\|\boldsymbol{y}\|_2^2}{M(\mathsf{SNR}_{\mathsf{init}}+1)}$

$\forall m : \widehat{s}_m[0] = 0$

$\mathsf{NR}_{\mathsf{best}} = \infty$

for $a = 1, 2, 3, \ldots, A_{\max}$,

  draw random $\widehat{\boldsymbol{x}}[0]$

  $\forall n : \nu_n^x[0] = \|\widehat{\boldsymbol{x}}[0]\|_2^2/N$

  for $i = 1, 2, 3, \ldots, I_{\max}$

    $\big(\widehat{\boldsymbol{x}}[i], \widehat{\boldsymbol{\nu}}^x[i], \widehat{\boldsymbol{s}}[i]\big) = \mathsf{GAMP}\big(\boldsymbol{A}, \{p_{X_n}(\cdot)\}_{n=1}^N, \{p_{Y|Z}(y_m|\cdot; \widehat{\nu^w}[i-1])\}_{m=1}^M, \widehat{\boldsymbol{x}}[i-1], \widehat{\boldsymbol{\nu}}^x[i-1], \widehat{\boldsymbol{s}}[i-1]\big)$

    $\widehat{\nu^w}[i] = \frac{2}{M}\|\boldsymbol{y} - |\boldsymbol{A}\widehat{\boldsymbol{x}}[i]|\|_2^2$

  end

  $\mathsf{NR} = \|\boldsymbol{y} - |\boldsymbol{A}\widehat{\boldsymbol{x}}[i]|\|_2^2/\|\boldsymbol{y}\|_2^2$

  if $\mathsf{NR} < \mathsf{NR}_{\mathsf{best}}$

    $\widehat{\boldsymbol{x}}_{\mathsf{best}} = \widehat{\boldsymbol{x}}[I_{\max}]$

    $\mathsf{NR}_{\mathsf{best}} = \mathsf{NR}$

  end

  if $\mathsf{NR} < \mathsf{NR}_{\mathsf{stop}}$

    stop

  end

end

output $\widehat{\boldsymbol{x}}_{\mathsf{best}}$

Table 3. The proposed PR-GAMP algorithm with $A_{\max}$ attempts, SNR initialization $\mathsf{SNR}_{\mathsf{init}}$, and stopping residual $\mathsf{NR}_{\mathsf{stop}}$.

with nonzero coefficients drawn i.i.d zero-mean circular-Gaussian. Then, for a given matrix $\boldsymbol{A}$, we generated M noisy intensity measurements $\boldsymbol{y} = |\boldsymbol{A}\boldsymbol{x}+\boldsymbol{w}|$, where $\boldsymbol{w}$ was i.i.d circular-Gaussian with variance selected to achieve a target signal-to-noise ratio of $\mathsf{SNR} \triangleq \|\boldsymbol{A}\boldsymbol{x}\|_2^2/\mathrm{E}\{\|\boldsymbol{w}\|_2^2\}$. Finally, each algorithm computed an estimate $\widehat{\boldsymbol{x}}$ from $(\boldsymbol{y}, \boldsymbol{A})$ in an attempt to best match $\boldsymbol{x}$ up to a tolerated ambiguity. For $\boldsymbol{A}$ with i.i.d random entries, we tolerate only a phase rotation on $\widehat{\boldsymbol{x}}$, while for Fourier $\boldsymbol{A}$ and real-valued $\boldsymbol{x}$, we tolerate a flip, circular shift, and phase rotation on $\widehat{\boldsymbol{x}}$. Performance was then assessed using normalized mean-squared error on the disambiguated estimate:

$$\mathsf{NMSE}(\widehat{\boldsymbol{x}}) \triangleq \min_{\boldsymbol{\Theta}} \frac{\|\boldsymbol{x} - \mathsf{disambig}(\widehat{\boldsymbol{x}}, \boldsymbol{\Theta})\|_2^2}{\|\boldsymbol{x}\|_2^2}, \tag{18}$$

where $\boldsymbol{\Theta}$ are the ambiguity parameters. When computing empirical phase-transition curves, we defined a "successful" recovery as one that produced $\mathsf{NMSE} < 10^{-6}$.

4.1 Empirical phase transitions: i.i.d Gaussian $\boldsymbol{A}$

First we investigated the phase-transition performance of PR-GAMP with i.i.d circular-Gaussian sensing matrices $\boldsymbol{A}$. Figure 4.1 plots the empirical success rate (averaged over 100 independent problem realizations) as a function of signal sparsity K and measurement length M for a fixed signal length of $N = 512$. Here we used $\mathsf{SNR} = 100$ dB, which makes the observations essentially "noiseless," and we allowed PR-GAMP up to 10 attempts from random initializations (i.e., $A_{\max} = 10$ in Table 3.5). The figure shows a "phase transition" behavior that separates the (K, M) plane into two regions: perfect recovery in the top-left and failure in the bottom-right. Moreover, the figure shows that, for $K \ll N$, approximately $M \geq 2K \log_2(N/K)$ measurements suffice for PR-GAMP.

To see how well (versus how often) PR-GAMP recovers the signal, we plot the median NMSE over the same problem realizations in Fig.4.1. There we see that the signal estimates are extremely accurate throughout the region above the phase transition.

To investigate the effect of number-of-attempts $A_{\max}$, we extracted the 50%-success contour (i.e., the phase-transition curve) from Fig. 4.1 and plotted it in Fig. 4.1, along with the corresponding contours obtained under different choices of $A_{\max}$. Figure 4.1 shows that, in the case of i.i.d $\boldsymbol{A}$, there is relatively little to gain from multiple restarts from random realizations. With Fourier $\boldsymbol{A}$, however, we will see in the sequel that multiple restarts are indeed important.

Figure 4.1 also plots the phase-transition curve of phase-oracle (PO)-GAMP calculated from the same problem realizations. Comparing the PO-GAMP phase transition to that of PR-GAMP, we conclude that PR-GAMP requires approximately $4\times$ the number of measurements as PO-GAMP, regardless of sparsity rate K. Remarkably, this "$4\times$" rule generalizes what is known about the recovery of *non*-sparse signals in $\mathbb{C}^N$, where the ratio of (necessary and sufficient) magnitude-only to magnitude-and-phase measurements is also 4 (as $N \to \infty$) [13, 14].

Overall, Figures 4.1–4.1 demonstrate that PR-GAMP is indeed capable of *compressive* phase retrieval, i.e., successful $\mathbb{C}^N$-signal recovery from $M \ll 4N$ intensity measurements, when the signal is sufficiently sparse. Moreover, to our knowledge, these phase transitions are far better than those of any other algorithm reported in the literature.

4.2 Robustness to noise

We now demonstrate the robustness of PR-GAMP to non-trivial levels of additive white circular-Gaussian noise $\boldsymbol{w}$ in the M intensity measurements $\boldsymbol{y} = |\boldsymbol{A}\boldsymbol{x} + \boldsymbol{w}|$. As before, we use an $N = 512$-length K-sparse signal with an i.i.d Gaussian $\boldsymbol{A}$, but now we focus on the case $(K, M) = (4, 256)$, which is on the good side of the phase-transition in Fig. 4.1. Figure 4.2 shows median NMSE performance over 200 independent problem realizations as a function of

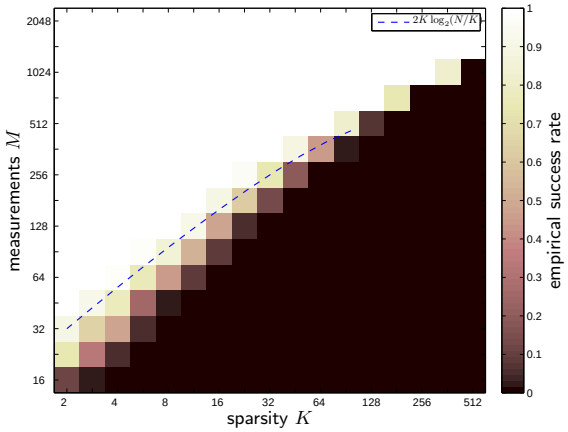

Fig. 2. Empirical probability of successful PR-GAMP recovery of an $N = 512$-length signal, versus signal sparsity K and number of intensity measurements M, using i.i.d Gaussian $\boldsymbol{A}$ at $\mathsf{SNR} = 100$ dB. Here, PR-GAMP was allowed up to 10 attempts from different random initializations.

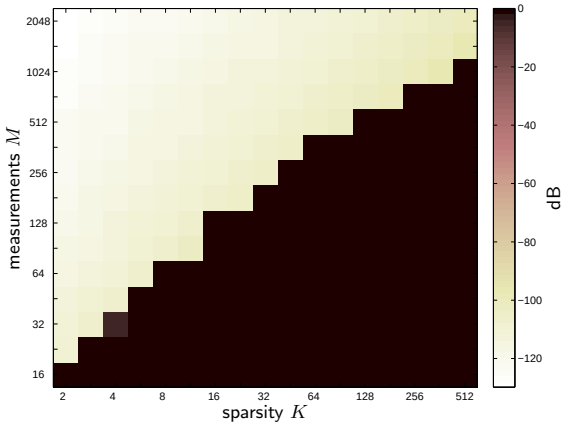

Fig. 3. Median NMSE for PR-GAMP recovery of an $N = 512$-length signal, versus signal sparsity K and number of intensity measurements M, using i.i.d Gaussian $\boldsymbol{A}$ at $\mathsf{SNR} = 100$ dB. Here, PR-GAMP was allowed up to 10 attempts from different random initializations.

$\mathsf{SNR} \triangleq \|\boldsymbol{Ax}\|_2^2/\|\boldsymbol{w}\|_2^2$. At larger values of SNR (i.e., $\mathsf{SNR} \geq 30$ dB), There we see that, throughout the tested SNR range, PR-GAMP performs only about 3 dB worse than PO-GAMP. The existence of a 3 dB gap can be explained by the fact that PO-GAMP is able to average the noise over twice as many real-valued measurements as PR-GAMP (i.e., $\{\mathrm{Re}\{u_m\}, \mathrm{Im}\{u_m\}\}_{m=1}^M$ versus $\{|u_m|\}_{m=1}^M$).

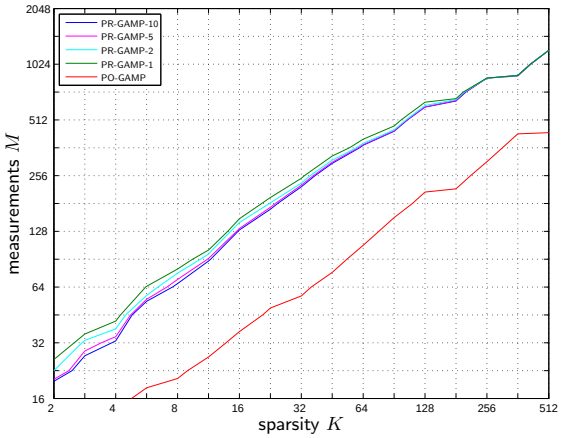

Fig. 4. 50%-success contours for PR-GAMP and phase-oracle GAMP recovery of an $N = 512$-length signal, versus signal sparsity K and number of intensity measurements M, using i.i.d Gaussian $\boldsymbol{A}$ at $\mathsf{SNR} = 100$ dB. PR-GAMP-$A_{\max}$ denotes PR-GAMP under a maximum of $A_{\max}$ attempts.

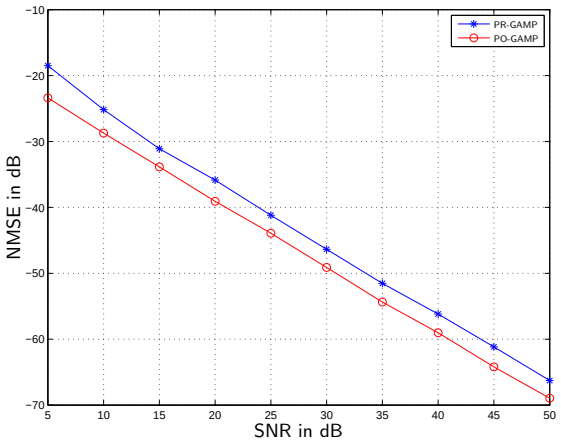

Fig. 5. Median NMSE for PR-GAMP and phase-oracle GAMP recovery of an $N = 512$-length $K = 4$-sparse signal, versus SNR, from $M = 256$ measurements and i.i.d Gaussian $\boldsymbol{A}$.

4.3 Comparison to CPRL

In this section, we present compare PR-GAMP to the state-of-the-art convex-relaxation approach to compressive phase retrieval, CPRL [22]. To implement CPRL, we used the authors' CVX-based matlab code[8] under default algorith-

[8] `http://users.isy.liu.se/rt/ohlsson/code/CPRL.zip`

	$(M,N)=(20,32)$	$(M,N)=(30,48)$	$(M,N)=(40,64)$
CPRL	1.00 (3.4 sec)	1.00 (37 sec)	1.00 (434 sec)
PR-GAMP	1.00 (0.22 sec)	1.00 (0.20 sec)	1.00 (0.18 sec)

Table 4. Empirical success rate and median runtime over 100 problem realizations for several combinations of signal length N, measurement length M, and signal sparsity $K = 1$.

	$(M,N)=(20,32)$	$(M,N)=(30,48)$	$(M,N)=(40,64)$
CPRL	0.55 (4.1 sec)	0.65 (42 sec)	0.66 (496 sec)
PR-GAMP	0.99 (0.28 sec)	0.99 (0.24 sec)	1.00 (0.22 sec)

Table 5. Empirical success rate and median runtime over 100 problem realizations for several combinations of signal length N, measurement length M, and signal sparsity $K = 2$.

mic settings. We also tried the authors' ADMM implementation, but found that it gave significantly worse performance. As before, we examine the recovery of a K-sparse signal in $\mathbb{C}^N$ from M intensity measurements $\boldsymbol{y} = |\boldsymbol{A}\boldsymbol{x}+\boldsymbol{w}|$, but now we use $\boldsymbol{A} = \boldsymbol{\Phi}\boldsymbol{F}$ with i.i.d circular-Gaussian $\boldsymbol{\Phi}$ and discrete Fourier transform (DFT) $\boldsymbol{F}$, to be consistent with the setup assumed in [22].

Table 4.3 shows empirical success[9] rate and runtime (on a standard personal computer) for a problem with sparsity $K = 1$, signal lengths $N \in \{32, 48, 64\}$, and compressive measurement lengths $M \in \{20, 30, 40\}$. The table shows that, over 100 problem realizations, both algorithms were 100% successful in recovering the signal at all tested combinations of (M, N). But the table also shows that CPRL's runtime increases rapidly with the signal dimensions, whereas that of PR-GAMP remains orders-of-magnitude smaller and independent of (M, N) over the tested range.[10]

Table 4.3 repeats the experiment carried out in Table 4.3, but at the sparsity $K = 2$. For this more difficult problem, the table shows that CPRL is much less successful at recovering the signal than PR-GAMP. Meanwhile, the runtimes reported in Table 4.3 again show CPRL complexity scaling rapidly with the problem dimension, whereas GAMP complexity stays orders-of-magnitude smaller and constant over the tested problem dimensions. In fact, the comparisons conducted in this section were restricted to very small problem dimensions precisely due to the poor complexity scaling of CPRL.

[9] Since CPRL rarely gave $\mathsf{NMSE} < 10^{-6}$, we reduced the definition of "success" to $\mathsf{NMSE} < 10^{-4}$ for this subsection only.

[10] Although the complexity of GAMP is known to scale as $O(MN)$ for this type of $\boldsymbol{A}$, the values of M and N in Table 4.3 are too small for this scaling law to manifest.

4.4 Comparison to sparse-Fienup and GESPAR: Fourier $\boldsymbol{A}$

In this section, we compare PR-GAMP to the sparse-Fienup [21] and GESPAR[11] [25] algorithms, which requires[12] us to restrict our attention to Fourier-based $\boldsymbol{A}$ and real-valued sparse vectors $\boldsymbol{x}$. For the experiments below, we generated realizations of $\boldsymbol{x}$ the same was as above, but with the non-zero elements drawn from a real-Gaussian distribution. Also, we used $ITER = 6400$ in GESPAR as recommended by the authors in [25], and we allowed sparse-Fienup 1000 attempts from random initializations.

We first consider 2D Fourier $\boldsymbol{A}$, which is especially important for imaging applications. In particular, we repeat an experiment from [25], where the measurement and signal lengths were fixed at $M = N$ and the signal sparsity K was varied. For $N = 1024$, Fig. 4.4 shows the empirical success rate (over 200 realizations) for PR-GAMP, GESPAR, and sparse Fienup. Meanwhile, Fig. 4.4 shows the corresponding median runtime for each algorithm, where all algorithms leveraged fast Fourier transform (FFT) implementations of $\boldsymbol{A}$. From Fig. 4.4, we can see that PR-GAMP yields a significantly better phase-transition than GESPAR and sparse Fienup. Meanwhile, from Fig. 4.4 we see that, for the challenging case of $K \geq 40$, PR-GAMP-10 has uniformly better runtime *and* success rate than GESPAR and sparse Fienup.

Next we consider 1D Fourier $\boldsymbol{A}$. Again, we repeat an experiment from [25], where the measurement and signal lengths were fixed at $M = 2N$ and the signal sparsity K was varied. For $N = 1024$, Fig. 4.4 shows the empirical success rate (over 200 realizations) for PR-GAMP, GESPAR, and sparse Fienup, and Fig. 4.4 shows the corresponding median runtimes. From Fig. 4.4, we can see that PR-GAMP yields a significantly better phase-transition than GESPAR and sparse Fienup. Meanwhile, from Fig. 4.4 we see that, for the challenging case of $K \geq 40$, PR-GAMP-20 has uniformly better runtime *and* success rate than GESPAR and sparse Fienup.

Comparing the results in this section to those in Section 4.1, we conclude that compressive phase retrieval is much more difficult with Fourier matrices $\boldsymbol{A}$ than with i.i.d matrices $\boldsymbol{A}$. This phenomenon has been noticed by other authors as well, which has led to proposals for randomized Fourier-based phase retrieval (e.g., using binary masks [46]). Also, we notice that the use of multiple restarts in PR-GAMP is much more important with Fourier $\boldsymbol{A}$ than it is with i.i.d $\boldsymbol{A}$.

[11] For GESPAR, we used the November 2013 version of the Matlab code provided by the authors at `https://sites.google.com/site/yoavshechtman/resources/software`.

[12] The sparse Fienup from [21] requires $\boldsymbol{A}^{\mathsf{H}}\boldsymbol{A}$ to be a (scaled) identity matrix. Although GESPAR can in principle handle generic $\boldsymbol{A}$, the implementation provided by the authors is based on 1D and 2D Fourier $\boldsymbol{A}$ and is not easily modified.

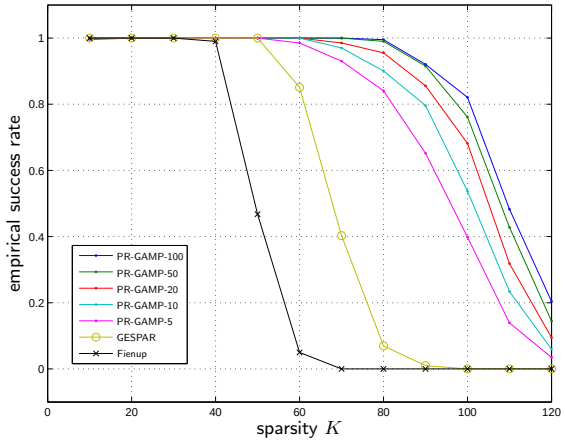

Fig. 6. Empirical success rate versus sparsity K in the recovery of an $N = 1024$-length real-valued signal from $M = 1024$ 2D-Fourier intensities at $\mathsf{SNR} = 100$dB. PR-GAMP-A denotes PR-GAMP under a maximum of A attempts.

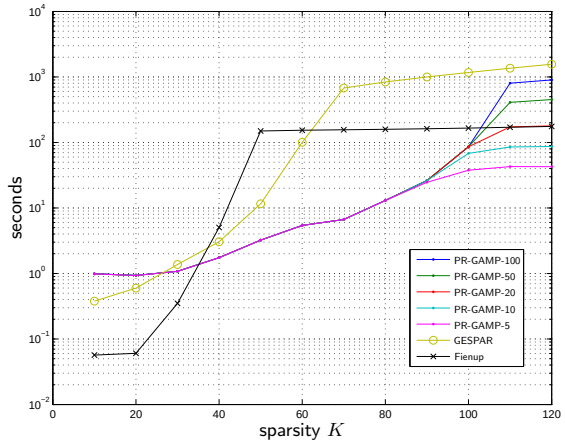

Fig. 7. Median runtime versus sparsity K in the recovery of an $N = 1024$-length real-valued signal from $M = 1024$ 2D-Fourier intensities at $\mathsf{SNR} = 100$dB. PR-GAMP-A denotes PR-GAMP under a maximum of A attempts.

4.5 Practical image recovery with masked Fourier $\boldsymbol{A}$

Finally, we demonstrate practical image recovery from compressed intensity measurements. For this experiment, the signal $\boldsymbol{x}$ was the $N = 65536$-pixel grayscale image shown on the left of Fig. 10, which has a sparsity of $K = 6678$. Since this image is real and non-negative, we ran PR-GAMP with a non-negative-real-BG prior [43], as opposed to the BG prior (16) used in previous experiments.

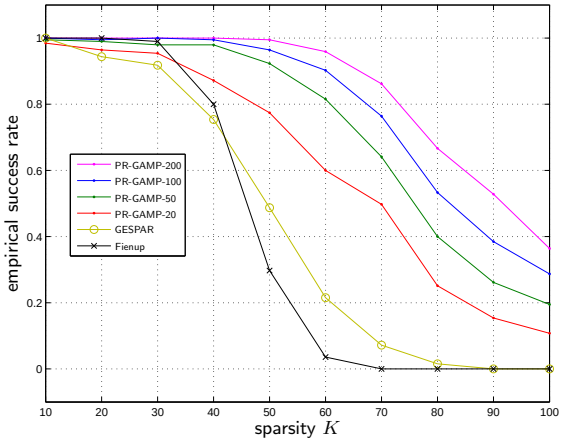

Fig. 8. Empirical success rate versus sparsity K in the recovery of an $N = 512$-length real-valued signal from $M = 1024$ 1D-Fourier intensities at $\mathsf{SNR} = 100$dB. PR-GAMP-A denotes PR-GAMP under a maximum of A attempts.

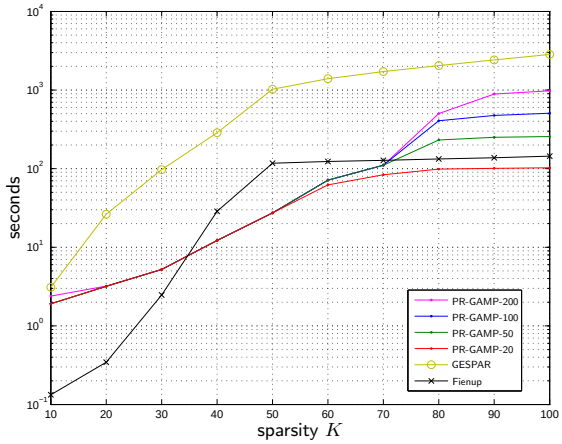

Fig. 9. Median runtime versus sparsity K in the recovery of an $N = 512$-length real-valued signal from $M = 1024$ 1D-Fourier intensities at $\mathsf{SNR} = 100$dB. PR-GAMP-A denotes PR-GAMP under a maximum of A attempts.

For the first set of experiments, we used a "masked" Fourier transformation $\boldsymbol{A} \in \mathbb{C}^{M \times N}$ of the form

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{J}_1 \boldsymbol{F} \boldsymbol{D}_1 \\ \boldsymbol{J}_2 \boldsymbol{F} \boldsymbol{D}_2 \\ \boldsymbol{J}_3 \boldsymbol{F} \boldsymbol{D}_3 \\ \boldsymbol{J}_4 \boldsymbol{F} \boldsymbol{D}_4 \end{bmatrix}, \tag{19}$$

where $\boldsymbol{F}$ was a 2D DFT matrix of size $N \times N$, $\boldsymbol{D}_i$ were diagonal "masking" matrices of size $N \times N$ with diagonal entries drawn uniformly at random from $\{0, 1\}$, and $\boldsymbol{J}_i$ were "selection" matrices of size $\frac{M}{4} \times N$ constructed from rows of the identity matrix drawn uniformly at random. The matrices $\boldsymbol{D}_i$ and $\boldsymbol{J}_i$ help to "randomize" the DFT, and they circumvent unicity issues such as shift and flip ambiguities. For phase retrieval, the use of image masks was discussed in [46]. Note that, because $\boldsymbol{D}_i$ and $\boldsymbol{J}_i$ are sparse and $\boldsymbol{F}$ has a fast FFT-based implementation, the overall matrix $\boldsymbol{A}$ has a fast implementation.

To eliminate the need for the expensive matrix multiplications with the elementwise-squared versions of $\boldsymbol{A}$ and $\boldsymbol{A}^{\mathsf{H}}$, as specified in steps (S1) and (S6) of Table 3.4, GAMP was run in "uniform variance" mode, meaning that $\{\nu^p_m(t)\}_{m=1}^M$ were approximated by $\nu^p(t) \triangleq \frac{1}{M}\sum_{m'=1}^M \nu^p_{m'}(t)$; similar was done with $\{\underline{\nu}^s_m(t)\}_{m=1}^M$, $\{\nu^r_n(t)\}_{n=1}^N$, and $\{\nu^x_n(t)\}_{n=1}^N$. The result is that lines (S1)-(S2) in Table 3.4 become $\nu^p(t) = \beta\|\boldsymbol{A}\|_F^2\nu^x(t)/M + (1-\beta)\nu^p(t-1) = \alpha(t)$ and line (S6) becomes $\underline{\nu}^r(t) = \big(\|\boldsymbol{A}\|_F^2\underline{\nu}^s(t)/N\big)^{-1}$.

As before, the observations took the form $\boldsymbol{y} = |\boldsymbol{A}\boldsymbol{x}+\boldsymbol{w}|$, but now the noise variance was adjusted to yield a nontrivial $\mathsf{SNR} = 30$ dB. To demonstrate *compressive* phase retrieval, only $M = N = 65536$ intensity measurements were used. Running PR-GAMP on 100 problem realizations (each with different random $\boldsymbol{A}$ and $\boldsymbol{w}$, and allowing at most 10 restarts per realization), a **99% success rate** was observed, where for this noisy problem "success" was defined as $\mathsf{NMSE} < \mathsf{SNR}^{-1} = -30$ dB. Furthermore, PR-GAMP's median runtime over these realizations was only **8.4 seconds**. The right subplot in Fig. 10 shows a typical PR-GAMP recovery.

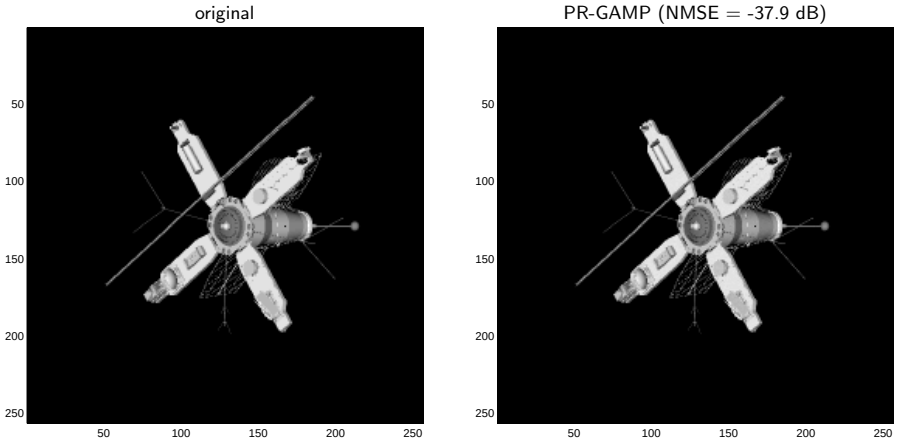

Fig. 10. Original image (left) and a typical PR-GAMP-recovery (right) from $M = N$ masked-Fourier intensity measurements at $\mathsf{SNR} = 30$ dB, which took 2.2 seconds.

For the second set of experiments, we "blurred" the masked-Fourier outputs to further randomize $\boldsymbol{A}$, which allowed us to achieve similar recovery performance using *half* the intensity measurements, i.e., $M = N/2 = 32768$. In particular, we used a linear transformation $\boldsymbol{A} \in \mathbb{C}^{M \times N}$ of the form

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{B}_1 \boldsymbol{F} \boldsymbol{D}_1 \\ \boldsymbol{B}_2 \boldsymbol{F} \boldsymbol{D}_2 \end{bmatrix}, \tag{20}$$

where $\boldsymbol{F}$ and $\boldsymbol{D}_i$ were as before[13] and $\boldsymbol{B}_i$ were banded[14] matrices of size $\frac{M}{2} \times N$ with 10 nonzero i.i.d circular-Gaussian entries per column. The use of blurring to enhance phase retrieval was discussed in [47]. As with (19), the $\boldsymbol{A}$ in (20) has a fast implementation. Running PR-GAMP as before on 100 problem realizations at $\mathsf{SNR} = 30$ dB, a **99% success rate** was observed with a median runtime of only **12.6 seconds**.

To our knowledge, no existing algorithms are able to perform compressive phase retrieval on images of this size and sparsity with such high speed and accuracy. To put our results in perspective, we recall the image recovery experiment in [25], which shows an example of GESPAR taking 80 seconds to recover a $K = 15$-sparse image whose support was effectively constrained to $N = 225$ pixels from $M = 38025$ 2D Fourier intensity measurements. In contrast, Fig. 10 shows PR-GAMP taking 2.2 seconds to recover a $K = 6678$-sparse image whose support was constrained to $N = 65536$ pixels from $M = 65536$ masked 2D Fourier intensity measurements.

5 Conclusions

In this paper, we proposed a novel approach to compressive phase retrieval based on the generalized approximate message passing (GAMP) algorithm. Numerical results showed that the proposed PR-GAMP algorithm has excellent phase transition behavior, noise robustness, and runtime. In particular, for successful recovery of synthetic K-sparse signals PR-GAMP requires approximately 4 times the number of measurements as phase-oracle GAMP and achieves NMSE that is only 3 dB worse than phase-oracle GAMP. For recovery of a real-valued 65532-pixel image from 32768 pre-masked and post-blurred Fourier intensities, PR-GAMP was successful 99% of the time with a median runtime of only 12.6 seconds. Comparison to the recently proposed CPRL, sparse-Fienup, and GESPAR algorithms revealed PR-GAMP's superior phase transitions and orders-of-magnitude faster runtimes at large K.

[13] Here, since we used only two masks, we ensured invertibility by constructing the diagonal of $\boldsymbol{D}_1$ using exactly $N/2$ unit-valued entries positioned uniformly at random and constructing the diagonal of $\boldsymbol{D}_2$ as its complement, so that $\boldsymbol{D}_1 + \boldsymbol{D}_2 = \boldsymbol{I}$.

[14] Since each $\boldsymbol{B}_i$ was a wide matrix, its nonzero band was wrapped from bottom to top when necessary.

A Output Thresholding Rules

In this appendix, we derive the expressions (10) and (12) that are used to compute the functions $g_{\mathsf{out},m}$ and $g'_{\mathsf{out},m}$ defined in lines (D2) and (D3) of Table 2.

To facilitate the derivations in this appendix,[15] we first rewrite $p_{Y|Z}(y|z)$ in a form different from (8). In particular, recalling that—under our AWGN assumption—the noisy transform outputs $u = z + w$ are conditionally distributed as $p(u|z) = \mathcal{N}(u; z, \nu^w)$, we first transform $u = ye^{j\theta}$ from rectangular to polar coordinates to obtain

$$p(y, \theta|z) = 1_{y\geq 0} 1_{\theta\in[0,2\pi)}\, \mathcal{N}(ye^{j\theta}; z, \nu^w)\, y \tag{21}$$

where y is the Jacobian of the transformation, and then integrate out the unobserved phase θ to obtain

$$p_{Y|Z}(y|z) = 1_{y\geq 0}\, y \int_0^{2\pi} \mathcal{N}(ye^{j\theta}; z, \nu^w)\, d\theta, \tag{22}$$

We begin by deriving the integration constant

$$\begin{aligned} C(y, \nu^w, \widehat{p}, \nu^p) &\triangleq \int_{\mathbb{C}} p_{Y|Z}(y|z)\, \mathcal{N}(z; \widehat{p}, \nu^p) dz \\ &= y\, 1_{y\geq 0} \int_0^{2\pi} \int_{\mathbb{C}} \mathcal{N}(ye^{j\theta}; z, \nu^w) \mathcal{N}(z; \widehat{p}, \nu^p) dz d\theta \qquad (23) \\ &= y\, 1_{y\geq 0} \int_0^{2\pi} \mathcal{N}(ye^{j\theta}; \widehat{p}, \nu^w + \nu^p) d\theta, \qquad (24) \end{aligned}$$

where we used the Gaussian-pdf multiplication rule[16] in (24). Noting the similarity between (24) and (22), the equivalence between (22) and (8) implies that

$$C(y, \nu^w, \widehat{p}, \nu^p) = \frac{2y}{\nu^w + \nu^p} \exp\left(-\frac{y^2 + |\widehat{p}|^2}{\nu^w + \nu^p} \right) I_0\left(\frac{2y|\widehat{p}|}{\nu^w + \nu^p} \right) 1_{y\geq 0}. \tag{25}$$

In the sequel, we make the practical assumption that $y > 0$, allowing us to drop the indicator "$1_{y\geq 0}$" and invert C.

Next, we derive the conditional mean

$$\mathrm{E}_{Z|Y,P}\{Z|y, \widehat{p}; \nu^p\} = C(y, \nu^w, \widehat{p}, \nu^p)^{-1} \int_{\mathbb{C}} z\, p_{Y|Z}(y|z; \nu^w) \mathcal{N}(z; \widehat{p}, \nu^p) dz. \tag{26}$$

Plugging (22) into (26) and applying the Gaussian-pdf multiplication rule,

[15] For notational brevity, the subscript "m" is omitted throughout this appendix for brevity.

[16] $\mathcal{N}(z; a, A)\mathcal{N}(z; b, B) = \mathcal{N}\Big(z; \frac{\frac{a}{A} + \frac{b}{B}}{\frac{1}{A} + \frac{1}{B}}, \frac{1}{\frac{1}{A} + \frac{1}{B}}\Big)\mathcal{N}(a; b, A+B)$.

$$
\begin{aligned}
&\mathrm{E}_{Z|Y,P}\{Z|y,\widehat{p};\nu^p\} \\
&= C^{-1} y \int_0^{2\pi} \int_{\mathbb{C}} z\,\mathcal{N}(z; ye^{j\theta}, \nu^w)\mathcal{N}(z;\widehat{p},\nu^p)dzd\theta && (27)\\
&= C^{-1} y \int_0^{2\pi} \int_{\mathbb{C}} z\,\mathcal{N}\Big(z; \frac{ye^{j\theta}/\nu^w + \widehat{p}/\nu^p}{1/\nu^w + 1/\nu^p}, \frac{1}{1/\nu^w+1/\nu^p}\Big) \\
&\quad \times \mathcal{N}(ye^{j\theta};\widehat{p},\nu^w+\nu^p)dzd\theta && (28)\\
&= C^{-1} y \int_0^{2\pi} \frac{ye^{j\theta}/\nu^w + \widehat{p}/\nu^p}{1/\nu^w+1/\nu^p}\mathcal{N}(ye^{j\theta};\widehat{p},\nu^w+\nu^p)d\theta && (29)\\
&= \frac{y/\nu^w}{1/\nu^w+1/\nu^p} C^{-1} y \int_0^{2\pi} e^{j\theta}\mathcal{N}(ye^{j\theta};\widehat{p},\nu^w+\nu^p)d\theta \\
&\quad + \frac{\widehat{p}/\nu^p}{1/\nu^w+1/\nu^p} C^{-1} y \int_0^{2\pi} \mathcal{N}(ye^{j\theta};\widehat{p},\nu^w+\nu^p)d\theta && (30)\\
&= \frac{y}{\nu^w/\nu^p+1} C^{-1} y \int_0^{2\pi} e^{j\theta}\mathcal{N}(ye^{j\theta};\widehat{p},\nu^w+\nu^p)d\theta + \frac{\widehat{p}}{\nu^p/\nu^w+1}. && (31)
\end{aligned}
$$

Expanding the $\mathcal{N}$ term, the integral in (31) becomes

$$
\begin{aligned}
&\int_0^{2\pi} e^{j\theta}\mathcal{N}(ye^{j\theta};\widehat{p},\nu^w+\nu^p)d\theta \\
&= \frac{1}{\pi(\nu^w+\nu^p)} \exp\Big(-\frac{y^2+|\widehat{p}|^2}{\nu^w+\nu^p}\Big) \int_0^{2\pi} e^{j\theta} \exp\Big(\frac{2y|\widehat{p}|}{\nu^w+\nu^p}\cos(\theta-\psi)\Big)d\theta && (32)\\
&= \frac{1}{\pi(\nu^w+\nu^p)} \exp\Big(-\frac{y^2+|\widehat{p}|^2}{\nu^w+\nu^p}\Big) e^{j\psi} \int_0^{2\pi} e^{j\theta'} \exp\Big(\frac{2y|\widehat{p}|}{\nu^w+\nu^p}\cos(\theta')\Big)d\theta' && (33)\\
&= \frac{2e^{j\psi}}{\nu^w+\nu^p} \exp\Big(-\frac{y^2+|\widehat{p}|^2}{\nu^w+\nu^p}\Big) I_1\Big(\frac{2y|\widehat{p}|}{\nu^w+\nu^p}\Big) && (34)
\end{aligned}
$$

where ψ denotes the phase of $\widehat{p}$, and where the integral in (33) was resolved using the expression in [48, 9.6.19]. Plugging (34) into (31) gives

$$
\mathrm{E}_{Z|Y,P}\{Z|y,\widehat{p};\nu^p\} = \frac{\widehat{p}}{\nu^p/\nu^w+1} + \frac{ye^{j\psi}}{\nu^w/\nu^p+1}\frac{I_1\big(\frac{2y|\widehat{p}|}{\nu^w+\nu^p}\big)}{I_0\big(\frac{2y|\widehat{p}|}{\nu^w+\nu^p}\big)}, \tag{35}
$$

which agrees with (10).

Finally, we derive the conditional covariance

$$
\begin{aligned}
\mathrm{var}_{Z|Y,P}\{Z|y,\widehat{p};\nu^p\} = C(y,\nu^w,\widehat{p},\nu^p)^{-1} \int_{\mathbb{C}} |z|^2\, p_{Y|Z}(y|z;\nu^w)\mathcal{N}(z;\widehat{p},\nu^p)dz \\
- |\mathrm{E}_{Z|Y,P}\{Z|y,\widehat{p};\nu^p\}|^2. && (36)
\end{aligned}
$$

Focusing on the first term in (36), if we plug in (22) and apply the Gaussian-pdf multiplication rule, we get

$$
\begin{aligned}
&C(y,\nu^w,\widehat{p},\nu^p)^{-1}\int_{\mathbb{C}}|z|^2\,p_{Y|Z}(y|z;\nu^w)\mathcal{N}(z;\widehat{p},\nu^p)dz\\
&=C^{-1}y\int_0^{2\pi}\int_{\mathbb{C}}|z|^2\,\mathcal{N}\Big(z;\frac{ye^{j\theta}/\nu^w+\widehat{p}/\nu^p}{1/\nu^w+1/\nu^p},\frac{1}{1/\nu^w+1/\nu^p}\Big)dz\\
&\quad\times\mathcal{N}(ye^{j\theta};\widehat{p},\nu^w+\nu^p)d\theta &(37)\\
&=C^{-1}y\int_0^{2\pi}\Big(\big|\frac{ye^{j\theta}/\nu^w+\widehat{p}/\nu^p}{1/\nu^w+1/\nu^p}\big|^2+\frac{1}{1/\nu^w+1/\nu^p}\Big)\mathcal{N}(ye^{j\theta};\widehat{p},\nu^w+\nu^p)d\theta &(38)\\
&=C^{-1}y\int_0^{2\pi}\frac{|y|^2/(\nu^w)^2+|\widehat{p}|^2/(\nu^p)^2+2y|\widehat{p}|/(\nu^w\nu^p)\mathrm{Re}\{e^{j(\theta-\psi)}\}}{(1/\nu^w+1/\nu^p)^2}\\
&\quad\times\mathcal{N}(ye^{j\theta};\widehat{p},\nu^w+\nu^p)d\theta+\frac{1}{1/\nu^w+1/\nu^p} &(39)\\
&=\frac{|y|^2/(\nu^w)^2+|\widehat{p}|^2/(\nu^p)^2}{(1/\nu^w+1/\nu^p)^2}+\frac{1}{1/\nu^w+1/\nu^p}\\
&\quad+\frac{2y|\widehat{p}|/(\nu^w\nu^p)}{(1/\nu^w+1/\nu^p)^2}C^{-1}y\mathrm{Re}\Big\{e^{-j\psi}\int_0^{2\pi}e^{j\theta}\mathcal{N}(ye^{j\theta};\widehat{p},\nu^w+\nu^p)d\theta\Big\} &(40)\\
&=\frac{|y|^2/(\nu^w)^2+|\widehat{p}|^2/(\nu^p)^2}{(1/\nu^w+1/\nu^p)^2}+\frac{1}{1/\nu^w+1/\nu^p}\\
&\quad+\frac{2y|\widehat{p}|/(\nu^w\nu^p)}{(1/\nu^w+1/\nu^p)^2}C^{-1}y\frac{2}{\nu^w+\nu^p}\exp\Big(-\frac{y^2+|\widehat{p}|^2}{\nu^w+\nu^p}\Big)I_1\Big(\frac{2y|\widehat{p}|}{\nu^w+\nu^p}\Big) &(41)\\
&=\frac{|y|^2/(\nu^w)^2+|\widehat{p}|^2/(\nu^p)^2}{(1/\nu^w+1/\nu^p)^2}+\frac{1}{1/\nu^w+1/\nu^p}+\frac{2y|\widehat{p}|/(\nu^w\nu^p)}{(1/\nu^w+1/\nu^p)^2}\frac{I_1\big(\frac{2y|\widehat{p}|}{\nu^w+\nu^p}\big)}{I_0\big(\frac{2y|\widehat{p}|}{\nu^w+\nu^p}\big)} &(42)
\end{aligned}
$$

where (41) used (34) and (42) used (25). By plugging (42) back into (36), we obtain the expression given in (12).

B EM Update for Noise Variance

Noting that

$$
\begin{aligned}
\ln p(\boldsymbol{y},\boldsymbol{x};\nu^w)&=\ln p(\boldsymbol{y}|\boldsymbol{x};\nu^w)+\ln p(\boldsymbol{x};\nu^w) &(43)\\
&=\sum_{m=1}^{M}\ln p_{Y|Z}(y_m|\boldsymbol{a}_m^{\mathsf{H}}\boldsymbol{x};\nu^w)+\text{const} &(44)\\
&=\sum_{m=1}^{M}\ln\Big(y_m\int_0^{2\pi}\mathcal{N}(y_me^{j\theta_m};\boldsymbol{a}_m^{\mathsf{H}}\boldsymbol{x},\nu^w)d\theta_m\Big)+\text{const}, &(45)
\end{aligned}
$$

where (45) used the expression for $p_{Y|Z}$ from (22), we have

$$\begin{aligned}&\mathrm{E}\big\{\ln p(\boldsymbol{y},\boldsymbol{x};\nu^w)\big|\boldsymbol{y};\widehat{\nu^w}[i]\big\}\\&=\int_{\mathbb{C}^N} p(\boldsymbol{x}|\boldsymbol{y};\widehat{\nu^w}[i])\sum_{m=1}^{M}\ln\Big(\int_0^{2\pi}\mathcal{N}(y_m e^{j\theta_m};\boldsymbol{a}_m^{\mathsf{H}}\boldsymbol{x},\nu^w)d\theta_m\Big)d\boldsymbol{x}.\end{aligned}\tag{46}$$

To circumvent the high-dimensional integral in (46), we use the same large-system-limit approximation used in the derivation of GAMP [29]: for sufficiently dense $\boldsymbol{A}$, as $N\to\infty$, the central limit theorem (CLT) suggests that $\boldsymbol{a}_m^{\mathsf{H}}\boldsymbol{x}=z_m$ is well becomes Gaussian. In particular, when $\boldsymbol{x}\sim p(\boldsymbol{x}|\boldsymbol{y};\widehat{\nu^w}[i])$, the CLT suggests that $\boldsymbol{a}_m^{\mathsf{H}}\boldsymbol{x}\sim\mathcal{N}(\widehat{z}_m,\nu_m^z)$, where

$$\widehat{z}_m[i]\triangleq\sum_{n=1}^{N}a_{mn}\widehat{x}_n[i]\tag{47}$$

$$\nu_m^z[i]\triangleq\sum_{n=1}^{N}|a_{mn}|^2\nu_n^x[i],\tag{48}$$

such that $\widehat{x}_n[i]$ and $\nu_n^x[i]$ are the mean and variance of the marginal posterior pdf $p(x_n|\boldsymbol{y};\widehat{\nu^w}[i])$. In this case,

$$\begin{aligned}&\mathrm{E}\big\{\ln p(\boldsymbol{y},\boldsymbol{x};\nu^w)\big|\boldsymbol{y};\widehat{\nu^w}[i]\big\}\\&=\sum_{m=1}^{M}\int_{\mathbb{C}}\mathcal{N}(z_m;\widehat{z}_m[i],\nu_m^z[i])\,\ln\int_0^{2\pi}\mathcal{N}(y_m e^{j\theta_m};z_m,\nu^w)d\theta_m dz_m.\end{aligned}\tag{49}$$

From (14), we see that any solution $\widehat{\nu^w}[i{+}1]>0$ is necessarily a value of ν^w that zeros the derivative of the expected log-pdf. Thus, using the expected-log-pdf expression from (49),

$$0=\sum_{m=1}^{M}\int_{\mathbb{C}}\mathcal{N}(z_m;\widehat{z}_m[i],\nu_m^z[i])\frac{\int_0^{2\pi}\frac{\partial}{\partial\nu^w}\mathcal{N}(y_m e^{j\theta_m};z_m,\widehat{\nu^w}[i{+}1])d\theta_m}{\int_0^{2\pi}\mathcal{N}(y_m e^{j\theta'_m};z_m,\widehat{\nu^w}[i{+}1])d\theta'_m}dz_m.\tag{50}$$

Plugging the derivative expression (see [39])

$$\begin{aligned}&\frac{\partial}{\partial\nu^w}\mathcal{N}(y_m e^{j\theta_m};z_m,\widehat{\nu^w}[i{+}1])\\&=\frac{\mathcal{N}(y_m e^{j\theta_m};z_m,\widehat{\nu^w}[i{+}1])}{\widehat{\nu^w}[i{+}1]^2}\big(|y_m e^{j\theta_m}-z_m|^2-\widehat{\nu^w}[i{+}1]\big),\end{aligned}\tag{51}$$

into (50) and multiplying both sides by $\widehat{\nu^w}[i{+}1]^2$, we find

$$\widehat{\nu^w}[i+1] = \frac{1}{M}\sum_{m=1}^{M}\int_{\mathbb{C}} \mathcal{N}(z_m;\widehat{z}_m[i],\nu^z_m[i]) \times \frac{\int_0^{2\pi}|y_m e^{j\theta_m}-z_m|^2\mathcal{N}(y_m e^{j\theta_m};z_m,\widehat{\nu^w}[i+1])d\theta_m}{\int_0^{2\pi}\mathcal{N}(y_m e^{j\theta'_m};z_m,\widehat{\nu^w}[i+1])d\theta'_m}dz_m \tag{52}$$

$$= \frac{1}{M}\sum_{m=1}^{M}\int_{\mathbb{C}} \mathcal{N}(z_m;\widehat{z}_m[i],\nu^z_m[i]) \times \int_0^{2\pi}|y_m e^{j\theta_m}-z_m|^2 p(\theta_m;z_m,\widehat{\nu^w}[i+1])d\theta_m dz_m \tag{53}$$

with the newly defined pdf

$$p(\theta_m;z_m,\widehat{\nu^w}[i+1]) \triangleq \frac{\mathcal{N}(y_m e^{j\theta_m};z_m,\widehat{\nu^w}[i+1])}{\int_0^{2\pi}\mathcal{N}(y_m e^{j\theta'_m};z_m,\widehat{\nu^w}[i+1])d\theta'_m} \tag{54}$$

$$\propto \exp\Big(-\frac{|z_m-y_m e^{j\theta_m}|^2}{\widehat{\nu^w}[i+1]}\Big) \tag{55}$$

$$\propto \exp\big(\kappa_m\cos(\theta_m-\phi_m)\big) \text{ for } \kappa_m \triangleq \frac{2|z_m|y_m}{\widehat{\nu^w}[i+1]}, \tag{56}$$

where ϕ_m is the phase of z_m (recall (5)). The proportionality (56) identifies this pdf as a von Mises distribution [49], which can be stated in normalized form as

$$p(\theta_m;z_m,\widehat{\nu^w}[i+1]) = \frac{\exp(\kappa_m\cos(\theta_m-\phi_m))}{2\pi I_0(\kappa_m)}. \tag{57}$$

Expanding the quadratic in (53) and plugging in (57), we get

$$\widehat{\nu^w}[i+1] = \frac{1}{M}\sum_{m=1}^{M}\int_{\mathbb{C}}\mathcal{N}(z_m;\widehat{z}_m[i],\nu^z_m[i])\bigg(y_m^2+|z_m|^2 - 2y_m|z_m|\int_0^{2\pi}\cos(\theta_m-\phi_m)\,\frac{\exp(\kappa_m\cos(\theta_m-\phi_m))}{2\pi I_0(\kappa_m)}d\theta_m\bigg)dz_m \tag{58}$$

$$= \frac{1}{M}\sum_{m=1}^{M}\int_{\mathbb{C}}\mathcal{N}(z_m;\widehat{z}_m[i],\nu^z_m[i]) \times \bigg(y_m^2+|z_m|^2-2y_m|z_m|R_0\Big(\frac{2|z_m|y_m}{\widehat{\nu^w}[i+1]}\Big)\bigg)dz_m \tag{59}$$

where $R_0(\cdot)$ is the modified Bessel function ratio defined in (13) and (59) follows from [48, 9.6.19]. To proceed further, we make use of the expansion $R_0(\kappa) = 1-\frac{1}{2\kappa}-\frac{1}{8\kappa^2}-\frac{1}{8\kappa^3}+o(\kappa^{-3})$ from [50, Lemma 5] to justify the high-SNR approximation

$$R_0(\kappa) \approx 1 - \frac{1}{2\kappa}, \tag{60}$$

which, when applied to (59), yields

$$\widehat{\nu^w}[i+1] \approx \frac{2}{M} \sum_{m=1}^{M} \int_{\mathbb{C}} \big(y_m - |z_m|\big)^2 \mathcal{N}(z_m; \widehat{z}_m[i], \nu_m^z[i]) dz_m. \tag{61}$$

Although (61) can be reduced to an expression that involves the mean of a Rician distribution, our empirical experience suggests that it suffices to assume $\nu_m^z[i] \approx 0$ in (61), after which we obtain the much simpler expression given in (15).

References

1. J. R. Fienup, "Phase retrieval algorithms: A comparison," *Appl. Optics*, vol. 21, pp. 2758–2769, Aug. 1982.
2. R. P. Millane, "Recent advances in phase retrieval," *The Int. Soc. Optical Eng.*, vol. 6316, 2006.
3. O. Bunk, A. Diaz, F. Pfeiffer, C. David, B. Schmitt, D. K. Satapathy, and J. F. Veen, "Diffractive imaging for periodic samples: Retrieving one-dimensional concentration profiles across microfluidic channels," *Acta Crystallographica Section A: Foundations of Crystallography*, vol. 63, no. 4, pp. 306–314, 2007.
4. R. W. Harrison, "Phase problem in crystallography," *J. Optical Soc. America A*, vol. 10, no. 5, pp. 1046–1055, 1993.
5. R. P. Millane, "Phase retrieval in crystallography and optics," *J. Optical Soc. America A*, vol. 7, pp. 394–411, Mar. 1990.
6. A. Chai, M. Moscoso, and G. Papanicolaou, "Array imaging using intensity-only measurements," *Inverse Problems*, vol. 27, no. 1, pp. 1–16, 2011.
7. A. Walther, "The question of phase retrieval in optics," *Optica Acta*, vol. 10, no. 1, pp. 41–49, 1963.
8. J. C. Dainty and J. R. Fienup, "Phase retrieval and image construction for astronomy," in *Image Recovery: Theory and Application* (H. Stark, ed.), ch. 7, pp. 231–275, New York: Academic Press, 1987.
9. J. Miao, T. Ishikawa, Q. Shen, and T. Earnest, "Extending x-ray crystallography to allow the imaging of noncrystalline materials, cells, and single protein complexes," *Annu. Rev. Phys. Chem.*, vol. 59, pp. 387–410, 2008.
10. R. Balan, P. G. Casazza, and D. Edidin, "On signal reconstruction without phase," *Appl. Computational Harmonic Anal.*, vol. 20, pp. 345–356, May 2006.
11. L. Demanet and V. Jugnon, "Convex recovery from interferometric measurements," *arXiv:1307.6864*, July 2013.
12. J. V. Corbett, "The pauli problem, state reconstruction and quantum-real numbers," *Reports on Mathematical Physics*, vol. 57, no. 1, 2006.
13. T. Heinosaari, L. Mazzarella, and M. M. Wolf, "Quantum tomography under prior information," *arXiv:1109.5478*, Sept. 2011.
14. B. G. Bodmann and N. Hammen, "Stable phase retrieval with low-redundancy frames," *arXiv:1302.5487*, Feb. 2013.

15. S. Gazit, A. Szameit, Y. C. Eldar, and M. Segev, "Super-resolution and reconstruction of sparse sub-wavelength images," *Opt. Express*, vol. 17, no. 26, pp. 23920–23946, 2009.
16. A. Szameit, Y. Shechtman, E. Osherovich, E. Bullkich, P. Sidorenko, H. Dana, S. Steiner, E. B. Kley, S. Gazit, T. Cohen-Hyams, S. Shoham, M. Zibulevsky, I. Yavneh, Y. C. Eldar, O. Cohen, and M. Segev, "Sparsity-based single-shot subwavelength coherent diffractive imaging," *Nature Materials*, vol. 11, pp. 455–459, 2012.
17. S. Marchesini, "Ab initio compressive phase retrieval," *arXiv:0809.2006*, Sept. 2008.
18. Y. Shechtman, E. Small, Y. Lahini, M. Verbin, Y. Eldar, Y. Silberberg, and M. Segev, "Sparsity-based super-resolution and phase-retrieval in waveguide arrays," *Opt. Express*, vol. 21, no. 20, pp. 24015–24024, 2013.
19. X. Li and V. Voroninski, "Sparse signal recovery from quadratic measurements via convex programming," *arXiv:1209.4785*, Sept. 2012.
20. M. L. Moravec, J. K. Romberg, and R. Baraniuk, "Compressive phase retrieval," in *SPIE Conf. Series*, vol. 6701, (San Diego, CA), Aug. 2007.
21. S. Mukherjee and C. S. Seelamantula, "An iterative algorithm for phase retrieval with sparsity constraints: Application to frequency domain optical coherence tomography," in *Proc. IEEE Int. Conf. Acoust. Speech & Signal Process.*, (Kyoto, Japan), pp. 553–556, Mar. 2012.
22. H. Ohlsson, A. Y. Yang, R. Dong, and S. S. Sastry, "CPRL – An extension of compressive sensing to the phase retrieval problem," in *Proc. Neural Inform. Process. Syst. Conf.*, 2012. (Full version at arXiv:1111.6323).
23. E. J. Candès, T. Strohmer, and V. Voroninski, "Phase lift: Exact and stable signal recovery from magnitude measurements via convex programming," *Commun. Pure & Appl. Math*, vol. 66, pp. 1241–1274, 2011.
24. P. Netrapalli, P. Jain, and S. Sanghavi, "Phase retrieval using alternating minimization," in *Proc. Neural Inform. Process. Syst. Conf.*, 2013. (See also arXiv:1306.0160).
25. Y. Shechtman, A. Beck, and Y. C. Eldar, "GESPAR: Efficient phase retrieval of sparse signals," *IEEE Trans. Signal Process.*, vol. 62, pp. 928–938, Feb. 2014.
26. C. Papadimitriou and K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity*. New York: Dover, 1998.
27. P. Schniter and S. Rangan, "Compressive phase retrieval via generalized approximate message passing," in *Proc. Allerton Conf. Commun. Control Comput.*, (Monticello, IL), Oct. 2012.
28. P. Schniter, "Compressive phase retrieval via generalized approximate message passing." presented at the *February Fourier Talks (FFT) Workshop on Phaseless Reconstruction*, (College Park, MD), Feb. 2013.
29. S. Rangan, "Generalized approximate message passing for estimation with random linear mixing," in *Proc. IEEE Int. Symp. Inform. Thy.*, (Saint Petersburg, Russia), pp. 2168–2172, Aug. 2011. (Full version at *arXiv:1010.5141*).
30. D. L. Donoho, A. Maleki, and A. Montanari, "Message passing algorithms for compressed sensing," *Proc. Nat. Acad. Sci.*, vol. 106, pp. 18914–18919, Nov. 2009.
31. D. L. Donoho, A. Maleki, and A. Montanari, "Message passing algorithms for compressed sensing: I. Motivation and construction," in *Proc. Inform. Theory Workshop*, (Cairo, Egypt), pp. 1–5, Jan. 2010.

32. J. Pearl, *Probabilistic Reasoning in Intelligent Systems*. San Mateo, CA: Morgan Kaufman, 1988.
33. F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inform. Theory*, vol. 47, pp. 498–519, Feb. 2001.
34. B. J. Frey and D. J. C. MacKay, "A revolution: Belief propagation in graphs with cycles," in *Proc. Neural Inform. Process. Syst. Conf.*, (Denver, CO), pp. 479–485, 1997.
35. M. Bayati and A. Montanari, "The dynamics of message passing on dense graphs, with applications to compressed sensing," *IEEE Trans. Inform. Theory*, vol. 57, pp. 764–785, Feb. 2011.
36. M. Bayati, M. Lelarge, and A. Montanari, "Universality in polytope phase transitions and iterative algorithms," in *Proc. IEEE Int. Symp. Inform. Thy.*, (Boston, Ma), pp. 1–5, June 2012. (see also arXiv:1207.7321).
37. S. O. Rice, "Mathematical analysis of random noise," *Bell Syst. Tech. J.*, vol. 24, no. 1, pp. 46–156, 1945.
38. A. Dempster, N. M. Laird, and D. B. Rubin, "Maximum-likelihood from incomplete data via the EM algorithm," *J. Roy. Statist. Soc.*, vol. 39, pp. 1–17, 1977.
39. J. P. Vila and P. Schniter, "Expectation-maximization Gaussian-mixture approximate message passing," *IEEE Trans. Signal Process.*, vol. 61, pp. 4658–4672, Oct. 2013.
40. G. F. Cooper, "The computational complexity of probabilistic inference using Bayesian belief networks," *Artificial Intelligence*, vol. 42, pp. 393–405, 1990.
41. P. Schniter, "Turbo reconstruction of structured sparse signals," in *Proc. Conf. Inform. Science & Syst.*, (Princeton, NJ), pp. 1–6, Mar. 2010.
42. M. Borgerding and P. Schniter, "Generalized approximate message passing for the cosparse analysis model," *arXiv:1312.3968*, 2013.
43. J. Vila and P. Schniter, "An empirical-Bayes approach to recovering linearly constrained non-negative sparse signals," in *Proc. IEEE Workshop Comp. Adv. Multi-Sensor Adaptive Process.*, (Saint Martin), pp. 5–8, Dec. 2013. (Full version at *arXiv:1310.2806*).
44. S. Som, L. C. Potter, and P. Schniter, "On approximate message passing for reconstruction of non-uniformly sparse signals," in *Proc. National Aerospace and Electronics Conf.*, (Dayton, OH), July 2010.
45. S. Rangan, P. Schniter, and A. Fletcher, "On the convergence of generalized approximate message passing with arbitrary matrices," in *Proc. IEEE Int. Symp. Inform. Thy.*, (Honolulu, Hawaii), July 2014, to appear. (Full version at *arXiv:1402.3210*).
46. E. J. Candès, X. Li, and M. Soltanolkotabi, "Phase retrieval from coded diffraction patterns," *arXiv:1310.3240*, Oct. 2013.
47. G. Zheng, R. Horstmeyer, and C. Yang, "Wide-field, high-resolution Fourier ptychographic microscopy," *Nature Photonics*, vol. 7, pp. 739–745, 2013.
48. M. Abramowitz and I. A. Stegun, eds., *Handbook of Mathematical Functions*. New York: Dover, 1964.
49. K. V. Mardia and P. E. Jupp, *Directional Statistics*. New York: Wiley, 2009.
50. C. Robert, "Modified Bessel functions and their applications in probability and statistics," *Stat. Prob. Lett.*, vol. 9, pp. 155–161, 1990.