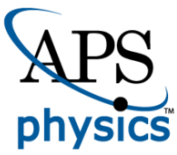


Phase diagram of matrix compressed sensing

Christophe Schülke, Philip Schniter, and Lenka Zdeborová

Phys. Rev. E **94**, 062136 — Published 27 December 2016

DOI: 10.1103/PhysRevE.94.062136

Phase diagram of matrix compressed sensing

Christophe Schülke*

Laboratoire de Physique Statistique, CNRS,

PSL Universités et Ecole Normale Supérieure, 75005, Paris, France and

Institut de Physique Théorique, CNRS, CEA,

Université Paris-Saclay, 91191, Gif-sur-Yvette, France

Philip Schniter†

Department of Electrical and Computer Engineering,

The Ohio State University, Columbus, OH 43210, USA

Lenka Zdeborová‡

Institut de Physique Théorique, CNRS, CEA,

Université Paris-Saclay, 91191, Gif-sur-Yvette, France

Abstract

In the problem of matrix compressed sensing, we aim to recover a low-rank matrix from a few noisy linear measurements. In this contribution, we analyze the asymptotic performance of a Bayes-optimal inference procedure for a model where the matrix to be recovered is a product of random matrices. The results that we obtain using the replica method describe the state evolution of the P-BiG-AMP algorithm, recently introduced in [J. T. Parker and P. Schniter, IEEE Journal of Selected Topics in Signal Processing **10**, 795 (2016)]. We show the existence of different two types of phase transition and their implications for the solvability of the problem, and we compare the results of our theoretical analysis to the numerical performance reached by P-BiG-AMP. Remarkably, the asymptotic replica equations for matrix compressed sensing are the same as those for a related but formally different problem of matrix factorization.

* christophe.schulke@espci.fr

† schniter@ece.osu.edu

‡ lenka.zdeborova@cea.fr

I. INTRODUCTION

Recovering a sparse or a low-rank signal from as few observations as possible is a class of problems that attracted considerable attention in statistics and signal processing. Well-known members of this class include compressed sensing [1] and or matrix completion [2]. Another important member of this class is the problem of matrix compressed sensing, where one aims to recover a low-rank matrix $\mathbf{X}$ from a few noisy linear measurements. We give a formal definition of the problem in Sec. I A. The matrix compressed sensing problem has a range of interesting applications, including quantum state tomography [3], face recognition [4], sensor localization [5], and many others [6]. We briefly discuss the first three applications below.

In *quantum state tomography*, a mixed quantum state is represented as a square positive semidefinite matrix $\mathbf{X}$ with unit trace. A pure state yields a rank 1 matrix, and an approximately pure state yields a low-rank matrix. An important practical problem is that of recovering $\mathbf{X}$ from a set of linear measurements. Since the size of $\mathbf{X}$ grows exponentially with the number of particles in the system, compressed sensing is useful to reduce the number of measurements [3]. In *face recognition*, one can exploit the fact that, ideally, all images of a face under varying illumination live in a 9-dimensional subspace (and this would be exactly true if faces were convex Lambertian bodies) [7]. Matrix compressive sensing thus makes it possible to recover a representation of a given face from a relatively small set of linear measurements, each under different (and unknown) illumination conditions [4]. Such representations can then be used directly in compressed-sensing based face recognition [8]. For *sensor localization*, one can exploit the fact that the matrix $\mathbf{X}$ of pairwise distances between sensors in a D-dimensional space has a rank of at most $D+2$. Thus, to save bandwidth and energy, sensors could transmit a few random combinations of the distances to their neighbors, rather than the full distance vectors, to a gateway node that uses compressive sensing to reconstruct the full $\mathbf{X}$ [5]. Several other applications of matrix compressed sensing are discussed in [6].

The main line of theoretical work related to matrix compressed sensing is based on minimizing the nuclear norm of the matrix (i.e. the sum of its singular values) subject to the constraint that a set of linear measurements agree with the measured values [9, 10]. Nuclear norm minimization is algorithmically tractable and provably recovers the unknown matrix

for an interesting range of parameters. The nuclear norm is a common type of regularization that encourages low-rank solutions. A rank R matrix $\mathbf{X}$ of dimension $M \times P$ can be written as a product of two matrices $\mathbf{X} = \mathbf{U}\mathbf{V}^\top$ of sizes $M \times R$ and $R \times P$. However, the nuclear norm minimization approach does not handle straightforwardly the case when there are additional structures (such as sparsity) on the factors $\mathbf{U}$ and $\mathbf{V}$.

In the present paper we study the generalized matrix compressed sensing problem, where general linear projections of $\mathbf{X}$ are observed through a noisy, and possibly non-linear, scalar output channel. Our analysis is restricted to a probabilistic setting where the components of the ground-truth factors $\mathbf{U}$ and $\mathbf{V}$ are i.i.d. random variables of known probability distribution, and where the probabilistic nature of the scalar output channel is known. Under such assumptions, the model is amenable to exact analysis via the replica method developed in statistical physics [11, 12]. The results stemming from the replica method are in general known to be in one-to-one correspondence with the analysis of message passing algorithms designed to solve the problem in an optimal way, as illustrated for the compressed sensing problem in [13], for matrix factorization in [14] and a number of other related problems. For the matrix compressed sensing problem, such a message-passing algorithm, called P-BiG-AMP, was derived and tested recently in [6]. Results of the replica method can hence also be viewed as an asymptotic analysis of the performance of this algorithm for the assumed model. We compare the results of the replica analysis to the performance of P-BiG-AMP and indeed observe excellent agreement, as expected from previous results for other models.

Our analysis reveals a striking connection between the matrix compressed sensing problem and the problem of matrix factorization as studied in [14–16]. These are two different inference problems. In matrix compressed sensing we observe a set of linear projections of the matrix $\mathbf{X}$, whereas in matrix factorization we observe the elements of the matrix $\mathbf{X}$ directly. Yet the replica analysis of the two problems yields equivalent equations and hence the asymptotic behaviors of the two problems, including their phase transitions, are closely linked. Similar links were already noticed between the nuclear-norm minimization approaches to matrix compressed sensing and matrix denoising in [10], and between matrix compressed sensing and matrix completion in [17].

Another main result of our work is establishing the existence of a large "hard but possible" phase corresponding to very sparse $\mathbf{U}$ and $\mathbf{V}$. This may come as a surprise, because in compressed sensing, perfect recovery from very sparse signals is achievable from random

initializations at very low measurement rates. We show that, in matrix compressed sensing, only *informed* initializations allow perfect reconstruction at low measurement rates, even with very sparse signals.

A. Definition of the problem

Let $\mathbf{X} \in \mathbb{R}^{M\times P}$ be a matrix of low rank $R < \min(M, P)$. It can thus be written as a product of two smaller matrices: $\mathbf{U} \in \mathbb{R}^{M\times R}$ and $\mathbf{V} \in \mathbb{R}^{P\times R}$,

$$\mathbf{X} = \mathbf{U}\mathbf{V}^\top. \tag{1}$$

The low-rank matrix compressed sensing problem consists in recovering $\mathbf{X}$ from a few noisy linear combinations of its entries. We call $\mathcal{A} : \mathbb{R}^{M\times P} \to \mathbb{R}^L$ the linear operator, where

$$\mathbf{Z} = \mathcal{A}(\mathbf{X}) \in \mathbb{R}^L \tag{2}$$

and we call $\mathbf{Y}$ the measured version of $\mathbf{Z}$ after passing through a component-wise measurement channel:

$$\mathbf{Y} \sim p^0_{Y|Z}(\mathbf{Y}|\mathbf{Z}). \tag{3}$$

This setting is shown in Fig. 1. The goal is to reconstruct $\mathbf{U}$ and $\mathbf{V}$ (or sometimes only $\mathbf{X}$) from the knowledge of $\mathbf{Y}$.

We can rewrite (2) in the component-wise manner

$$\forall l \in [1, L], \quad z_l = \sum_{\mu=1}^{M}\sum_{p=1}^{P} A_l^{\mu p} x_{\mu p}, \tag{4}$$

where the coefficients $A_l^{\mu p}$ parameterize $\mathcal{A}$. Notice that these coefficients define a 3-way tensor $\mathbf{A}$.

1. *The probabilistic model and assumptions of our analysis.*

In order to enable the asymptotic analysis (i.e. when $M, P, L \to \infty$) via the replica method we introduce the following probabilistic model for matrix compressed sensing.

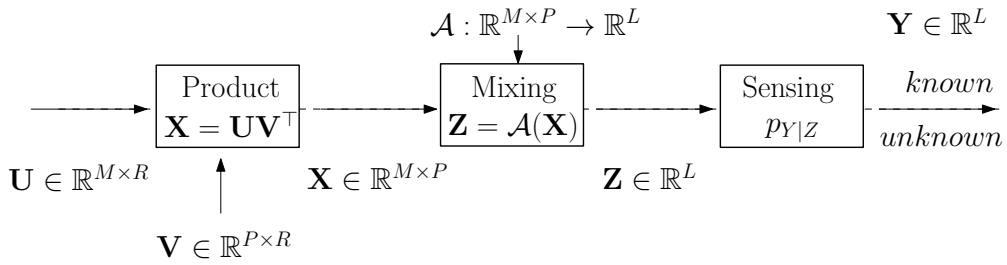


FIG. 1. The setting of generalized matrix compressed sensing. A low-rank matrix $\mathbf{X}$ can be decomposed into a product of two smaller matrices $\mathbf{U}$ and $\mathbf{V}$. A linear operator $\mathcal{A}$ is applied to $\mathbf{X}$, producing an intermediary variable $\mathbf{Z}$. A measurement $\mathbf{Y}$ of $\mathbf{Z}$ is obtained through a noisy channel. The problem is closely linked to other inference problems: dropping the "mixing" block, one recovers a generalized matrix factorization problem. Dropping the "product" block, one recovers a generalized linear model.

- We assume that elements of $\mathbf{U}$ and $\mathbf{V}$ are sampled independently at random such that

$$\mathbf{U} \sim \prod_{\mu s} p^0_U(u_{\mu s}), \qquad\qquad \mathbf{V} \sim \prod_{ps} p^0_V(v_{ps}). \tag{5}$$

  We assume the distributions p^0_U and p^0_V to have zero mean and respective variances Q^0_u and Q^0_v of order one. These distributions might not be known exactly: instead, we use zero-mean priors p_U and p_V believed to be close to p^0_U and p^0_V (in terms of Kullback-Leibler divergence).

- We assume the output distribution $p^0_{Y|Z}$ to be separable

$$p^0_{Y|Z} = \prod_l p^0_{Y|Z}(y_l|z_l)\,. \tag{6}$$

  In the inference we use a distribution $p_{Y|Z}$ we believe to be close to $p^0_{Y|Z}$ (in terms of Kullback-Leibler divergence).

- We assume the tensor $\mathbf{A}$ of the linear operator $\mathcal{A}$ to be normally distributed i.i.d. elements with zero mean and variance $1/(RMP)$, such that the elements of $\mathbf{Z}$ have zero mean and variance $Q^0_u Q^0_v$. A similar assumption is often made in compressed sensing, which differentiates the problem from matrix factorization, in which $\mathcal{A}$ is the identity.

- We assume the dimensions M, P and L to be large, but their following ratios to be of order one

$$\alpha_U = \frac{L}{RM}, \qquad \alpha_V = \frac{L}{RP}. \tag{7}$$

  On the other hand, R can be small.

2. Measures of recovery and symmetries of the problem

Given the estimates $(\hat{\mathbf{U}}, \hat{\mathbf{V}}, \hat{\mathbf{X}})$ that an algorithm returns for $(\mathbf{U}, \mathbf{V}, \mathbf{X})$, the following mean squared errors quantify how close the estimates are from the real values:

$$\text{MSE}_u = \frac{\|\mathbf{U} - \hat{\mathbf{U}}\|_F^2}{MR}, \qquad \text{MSE}_v = \frac{\|\mathbf{V} - \hat{\mathbf{V}}\|_F^2}{PR}, \qquad \text{MSE}_x = \frac{\|\mathbf{X} - \hat{\mathbf{X}}\|_F^2}{LR}, \tag{8}$$

where $||\cdot||_F$ is the Frobenius norm of a matrix. Note that as in matrix factorization, there is an inherent ill-posedness when it comes to recovering the couple $(\mathbf{U}, \mathbf{V})$. As a matter of fact, for any $R \times R$ invertible matrix $\mathbf{C}$, the couple $(\mathbf{UC}, \mathbf{V}\left(\mathbf{C}^{-1}\right)^\top)$ generates the same $\mathbf{X}$ as $(\mathbf{U}, \mathbf{V})$. In some case, this symmetry can be lifted thanks to the distributions p_U^0 and p_V^0, but this is not always the case and might nevertheless be cause of trouble. In that case, it is possible to have a very low MSE_x but high MSE_u and MSE_v.

In the setting where $R = 1$, $\mathbf{U}$ and $\hat{\mathbf{U}}$ are vectors and we can consider the following definitions of normalized mean squared errors

$$\text{nMSE}_u = 1 - \frac{\left|\mathbf{U}^\top \hat{\mathbf{U}}\right|}{||\mathbf{U}||_2 ||\hat{\mathbf{U}}||_2}, \qquad \text{nMSE}_v = 1 - \frac{\left|\mathbf{V}^\top \hat{\mathbf{V}}\right|}{||\mathbf{V}||_2 ||\hat{\mathbf{V}}||_2} \tag{9}$$

that take values between 0 and 1 and take into account all invariances of the problem: an nMSE of 0 indicates perfect reconstruction up to the scaling invariance.

Note that the mentioned symmetry in the $R > 1$ case has to be taken into account in the theoretical analysis of the problem (see (B6)) in order to obtain the best achievable MSEs. In contrast, in the P-BiG-AMP algorithm the symmetry is broken spontaneously by the choice of the initialization.

B. Notations

We use bold letters for vectors and matrices and non-bold letters for scalars. The elements of a vector $\mathbf{x}$ are noted $[\mathbf{x}]_i$ or x_i. The operator $\odot$ is used for component-wise multiplication

of vectors or matrices. $\mathbf{x}^{-1}$, $\mathbf{x}^2$ and $\mathbf{x}^\top$ refer respectively to the component-wise inverse, the component-wise square and the transpose of the vector (or matrix or tensor) $\mathbf{x}$. If $\mathcal{A}$ is a linear operator and $\mathbf{A}$ its tensor, we write $\mathcal{A}^2$ for the linear operator associated to $\mathbf{A}^2$. We use the notation $\imath \equiv \sqrt{-1}$. Estimators $\hat{X}$ and $\hat{x}$ of a variable X are the minimal mean squared error (MMSE) estimators of estimated probability distribution functions $\hat{P}(x)$ and $\hat{p}(x)$. We note $\bar{X}$ and $\bar{x}$ the variances of these distributions and refer to them as *uncertainties*, as they are a measure of the uncertainty of the estimators $\hat{X}$ and $\hat{x}$.

Using the tensor $\mathbf{A}$, we can define two auxiliary linear operators $\mathcal{A}_U : \mathbb{R}^P \to \mathbb{R}^{L\times M}$ and $\mathcal{A}_V : \mathbb{R}^M \to \mathbb{R}^{L\times P}$ such that

$$[\mathcal{A}_U(\mathbf{v})]_{l\mu} \equiv \sum_p A_l^{\mu p} v_p, \tag{10}$$

$$[\mathcal{A}_V(\mathbf{u})]_{lp} \equiv \sum_\mu A_l^{\mu p} u_\mu. \tag{11}$$

We note $x \sim p_X(x)$ a random variable x following the probability distribution p_X. This holds also for vectors and matrices: $\mathbf{x} \sim p_X(\mathbf{x})$. In that case, we say that $p_X(\mathbf{x})$ is separable if each component x_i of $\mathbf{x}$ is sampled independently from the others: $\forall i,\ x_i \sim p_{X_i}(x_i)$, which we will note p_X as well if the components are identically distributed.

We write $f(x) \propto g(x)$ when the functions f and g are equal up to a multiplying constant that does not depend on x. We write $K = O(1)$ (respectively $K = O(M)$) in order to signify that K is of order 1 (respectively M).

Let us introduce some useful functions that will be used throughout the paper. We note $\mathcal{N}(x; \hat{x}, \bar{x})$ the normalized Gaussian with mean $\hat{x}$ and variance $\bar{x}$:

$$\mathcal{N}(x; \hat{x}, \bar{x}) = \frac{1}{\sqrt{2\pi\bar{x}}} e^{-\frac{(x-\hat{x})^2}{2\bar{x}}}. \tag{12}$$

In integrals, we note $\mathrm{D}t$ the integration over a variable t with a standard normal distribution:

$$\mathrm{D}t = \mathrm{d}t\, \mathcal{N}(t; 0, 1). \tag{13}$$

For any function h and integer i, we define the i-th moment of the product of h multiplied by a Gaussian:

$$f_i^h(\hat{x}, \bar{x}) = \int \mathrm{d}x\, x^i h(x) \mathcal{N}(x; \hat{x}, \bar{x}). \tag{14}$$

With (14), we define the mean and the variance of the distribution $\frac{h(x)\mathcal{N}(x;\hat{x},\bar{x})}{f_0^h(\hat{x},\bar{x})}$:

$$\hat{f}^h(\hat{x},\bar{x}) = \frac{f_1^h(\hat{x},\bar{x})}{f_0^h(\hat{x},\bar{x})}, \tag{15}$$

$$\bar{f}^h(\hat{x},\bar{x}) = \frac{f_2^h(\hat{x},\bar{x})}{f_0^h(\hat{x},\bar{x})} - \hat{f}^h(\hat{x},\bar{x})^2, \tag{16}$$

It can be verified that following relations hold:

$$\frac{\partial}{\partial\hat{x}} f_i^h(\hat{x},\bar{x}) = \frac{1}{\bar{x}}\left(f_{i+1}^h(\hat{x},\bar{x}) - \hat{x} f_i^h(\hat{x},\bar{x})\right), \tag{17}$$

$$\frac{\partial}{\partial\bar{x}} f_i^h(\hat{x},\bar{x}) = \frac{1}{2\bar{x}^2}\left(f_{i+2}^h(\hat{x},\bar{x}) - 2\hat{x} f_{i+1}^h(\hat{x},\bar{x}) - (\bar{x}-\hat{x}^2) f_i^h(\hat{x},\bar{x})\right), \tag{18}$$

$$\frac{\partial}{\partial s} f_i^h(\sqrt{s}t,\rho-s) = -\frac{e^{\frac{t^2}{2}}}{2s}\frac{\partial}{\partial t}\left(e^{-\frac{t^2}{2}}\frac{\partial}{\partial t} f_i^h(\sqrt{s}t,\rho-s)\right). \tag{19}$$

Finally, we introduce two further useful auxiliary functions:

$$\hat{g}^h(\hat{x},\bar{x}) = \frac{\hat{f}^h(\hat{x},\bar{x}) - \hat{x}}{\bar{x}}, \qquad \bar{g}^h(\hat{x},\bar{x}) = \frac{\bar{f}^h(\hat{x},\bar{x}) - \bar{x}}{\bar{x}^2}. \tag{20}$$

II. MESSAGE-PASSING ALGORITHM

In this paper, we will focus on an approximate message passing (AMP) algorithm. AMP algorithms originated in studies of problems related to linear estimation [18–20]. For the above probabilistic model of matrix compressed sensing, AMP was derived and called P-BiG-AMP in [6]. In the following, we explain its principle and expose the main steps of its derivation.

In Bayesian inference, one seeks to produce estimators $\hat{\mathbf{U}}$ and $\hat{\mathbf{V}}$ of $\mathbf{U}$ and $\mathbf{V}$ using the following posterior probability:

$$p(\mathbf{U},\mathbf{V}|\mathbf{Y},\mathcal{A}) \propto p_U(\mathbf{U}) p_V(\mathbf{V}) p_{Y|Z}\left(\mathbf{Y}|\mathcal{A}(\mathbf{U}\mathbf{V}^\top)\right). \tag{21}$$

As explained above, the probability distributions used in (21) ideally match the distributions (3, 5) used for the generation of the problem, in which case the inference is said to be Bayes-optimal. However, it is often the case that these distributions are not known exactly: in this case, the distributions used in (21) are assumptions that we make on the signals' distributions and on the measurement channel. Inference is in that case suboptimal. However, in similar problems it has turned out that the results can still be satisfying despite the mismatch between the priors and the actual probability distributions. Furthermore, it

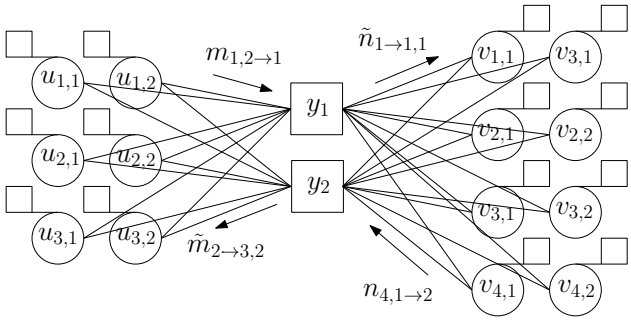


FIG. 2. Factor graph associated to the probability distribution (22). Here, we used $R = 2$, $M = 3$, $P = 4$, $L = 2$. Circle represent variables, squares represent constraints. The small squares represent the priors on the variables u and v. Messages $(m, \tilde{m}, n, \tilde{n})$ are sent along each edge of the factor graph.

is possible to parametrize the priors and learn the parameters during inference, for example using an expectation maximization procedure [21], which has proven to give satisfying results [13].

Starting from the posterior probability distribution (21), the two interesting questions are how to evaluate this quantity and how to obtain estimators $(\hat{\mathbf{U}}, \hat{\mathbf{V}})$ from it. For the second point, we will use the minimal mean squared error (MMSE) estimator, as our goal is to obtain low MSEs for (8). Concerning the first point, the problem in estimating (21) is that it is a distribution in a high-dimensional space. Though it is possible to sample from such a distribution using a Monte Carlo Markov chain, the procedure is very time consuming. Therefore we resort to loopy belief propagation (BP) to estimate the marginals of (21). Though not guaranteed to converge on this type of problems, BP has proven to be very successful in a variety of similar inference problems [13, 22].

In order to derive the BP algorithm, we first rewrite (21) to make all variables appear individually:

$$p(\mathbf{U}, \mathbf{V}, |\mathbf{Y}, \mathcal{A}) \propto \prod_{\mu s} p_U(u_{\mu s}) \prod_{ps} p_V(v_{ps}) \int \prod_{l} \mathrm{d}z_l p_{Y|Z}(y_l|z_l) \delta \left(z_l - \sum_{p=1}^{P} \sum_{\mu=1}^{M} A_l^{p\mu} \sum_{s=1}^{R} u_{ps} v_{\mu s} \right). \tag{22}$$

This probability distribution can be represented by the factor graph in figure 2. On it, two types of message pairs $(m, \tilde{m})$ and $(n, \tilde{n})$ are sent to and from the u and v variables respectively. As the roles of u and v are completely symmetric, we will only treat explicitly

the pair $(m, \tilde{m})$: the result can be generalized straightforwardly to $(n, \tilde{n})$. The message-passing update equations read:

$$m^{t+1}_{\mu s\to l}(u_{\mu s}) \propto p_U(u_{\mu s}) \prod_{l'\neq l} \tilde{m}^t_{l'\to \mu s}(u_{\mu s}), \tag{23}$$

$$\tilde{m}^{t+1}_{l\to \mu s}(u_{\mu s}) \propto \int \left(\prod_{ps'} \mathrm{d}v_{ps'} n^{t+1}_{ps'\to l}(v_{ps'}) \prod_{(s',\mu')\neq(s,\mu)} \mathrm{d}u_{\mu s'} m^{t+1}_{\mu s'\to l}(u_{\mu s'}) \right) \mathrm{d}z p_{Y|Z}(y_l|z) \delta(z - \mathcal{A}(\mathbf{U}\mathbf{V}^\top)), \tag{24}$$

where the $\propto$ sign stands because $(m, \tilde{m})$ are probability distributions and must therefore be normalized. These equations can be seen as fixed point equations or as iterative equations that constitute an algorithm. For notational lightness, we will do the following calculations without time indices. However, the correct time indices are crucial for the final algorithm to converge.

A first simplification can be made by replacing the $R(M+P)$ integrals in (24) by a single one over the variable z, which is the sum of $R(M+P)-1$ random variables. In BP, we assume these random variables to be independent, which allows us to use the central limit theorem. Calling $\hat{u}_{\mu s\to l}$ and $\bar{u}_{\mu s\to l}$ respectively the means and variances of the variable $u_{\mu s}$ distributed according to the distribution $m_{\mu s\to l}$ (and similarly for the variables v_{ps}), the variable $z_l = \sum_{\mu p} A^{\mu p}_l \sum_s u_{\mu s} v_{ps}$ is a Gaussian variable with mean and variance:

$$\hat{Z}_l = \sum_{\mu p s} A^{\mu p}_l \hat{u}_{\mu s\to l} \hat{v}_{ps\to l}, \tag{25}$$

$$\begin{aligned} \bar{Z}_l = &\sum_{\mu p s} (A^{\mu p}_l)^2 \left[\bar{u}_{\mu s\to l} \bar{v}_{ps\to l} + (\hat{u}_{\mu s\to l})^2 \bar{v}_{ps\to l} + \bar{u}_{\mu s\to l} (\hat{v}_{ps\to l})^2 \right] \\ &+ \sum_{ps} \sum_{\mu\neq\mu'} A^{\mu p}_l A^{\mu' p}_l \bar{v}_{ps\to l} \hat{u}_{\mu s\to l} \hat{u}_{\mu s'\to l} \\ &+ \sum_{\mu s} \sum_{p\neq p'} A^{\mu p}_l A^{\mu p'}_l \bar{u}_{\mu s\to l} \hat{v}_{ps\to l} \hat{v}_{ps'\to l}. \end{aligned} \tag{26}$$

However, in eq. (24), $u_{\mu s}$ is fixed and thus $(\hat{u}_{\mu s\to l}, \bar{u}_{\mu s\to l})$ has to be replaced by $(u_{\mu s}, 0)$

in (25,26). Defining $(\hat{Z}_{l\to\mu s}, \bar{Z}_{l\to\mu s})$ to be $(\hat{Z}_l, \bar{Z}_l)$ with $(\hat{u}_{\mu s\to l}, \bar{u}_{\mu s\to l}) = (0, 0)$ and

$$F_{l\mu s} = \sum_p A_l^{\mu p} \hat{v}_{ps\to l}, \tag{27}$$

$$H_{l\mu s} = 2 \sum_p \sum_{\mu' \neq \mu} A_l^{\mu p} A_l^{\mu' p} \hat{u}_{\mu s'\to l} \bar{v}_{ps\to l}, \tag{28}$$

$$G_{l\mu s} = \sum_p (A_l^{\mu p})^2 \bar{v}_{ps\to l}, \tag{29}$$

one can rewrite (24) with a single integral over a variable z following a Gaussian distribution. Using the definition (14), the message (24) can be expressed as a simple function of the mean and variance of this Gaussian:

$$\tilde{m}_{l\to\mu s}(u_{\mu s}) \propto f_0^Y \left(\hat{Z}_{l\to\mu s} + F_{l\mu s} u_{\mu s}, \bar{Z}_{l\to\mu s} + H_{l\mu s} u_{\mu s} + G_{l\mu s} u_{\mu s}^2 \right). \tag{30}$$

Here, we use the simplified notation $f_i^Y \equiv f_i^{p_{Y|Z}}$. In appendix A, we show how by making a Taylor expansion of this equation, we can express the message (23) as

$$m_{\mu s\to l}(u_{\mu s}) \propto p(u_{\mu s}) \mathcal{N}\left(u_{\mu s}; \hat{U}_{\mu s\to l}, \bar{U}_{\mu s\to l} \right), \tag{31}$$

with

$$\bar{U}_{\mu s\to l} = -\left(\sum_{l' \neq l} \left(F_{l'\mu s}^2 + G_{l'\mu s} \right) \bar{g}_{l'\to\mu s} + G_{l'\mu s} \hat{g}_{l'\to\mu s}^2 \right)^{-1}, \tag{32}$$

$$\hat{U}_{\mu s\to l} = \bar{U}_{\mu s\to l} \sum_{l' \neq l} F_{l'\mu s} \hat{g}_{l'\to\mu s}, \tag{33}$$

where

$$\hat{g}_{l'\to\mu s} = \hat{g}^Y(\hat{Z}_{l'\to\mu s}, \bar{Z}_{l'\to\mu s}), \qquad \bar{g}_{l'\to\mu s} = \bar{g}^Y(\hat{Z}_{l'\to\mu s}, \bar{Z}_{l'\to\mu s}), \tag{34}$$

and $(\hat{g}^Y(\cdot,\cdot), \bar{g}^Y(\cdot,\cdot))$ are simplified notations for the functions $(\hat{g}^{p_{Y|Z}}(\cdot,\cdot), \bar{g}^{p_{Y|Z}}(\cdot,\cdot))$ defined in (20).

This allows us to have a simple expression for the previously introduced mean and variance $\hat{u}_{\mu s\to l}$ and $\bar{u}_{\mu s\to l}$ of the message (31). Using the notations (15, 16),

$$\hat{u}_{\mu s\to l} = \hat{f}^U \left(\hat{U}_{\mu s\to l}, \bar{U}_{\mu s\to l} \right), \qquad \bar{u}_{\mu s\to l} = \bar{f}^U \left(\hat{U}_{\mu s\to l}, \bar{U}_{\mu s\to l} \right), \tag{35}$$

where as before, we introduce the simplifying notation $f^U \equiv f^{p_U}$. As noted previously, the exact same thing can be done for the messages $(n, \tilde{n})$. The result is an iterative set of equations on a set of means and variances

$$\left(\hat{Z}_{\cdot\to\cdot}^t, \bar{Z}_{\cdot\to\cdot}^t, \hat{g}_{\cdot\to\cdot}^t, \bar{g}_{\cdot\to\cdot}^t, \hat{U}_{\cdot\to\cdot}^t, \bar{U}_{\cdot\to\cdot}^t, \hat{u}_{\cdot\to\cdot}^t, \bar{u}_{\cdot\to\cdot}^t, \hat{V}_{\cdot\to\cdot}^t, \bar{V}_{\cdot\to\cdot}^t, \hat{v}_{\cdot\to\cdot}^t, \bar{v}_{\cdot\to\cdot}^t \right) \tag{36}$$

that constitute the message-passing algorithm.

Algorithm 1 P-BiG-AMP for matrix compressed sensing

Initialization:

Initialize the means $(\hat{\mathbf{u}}_0, \hat{\mathbf{v}}_0, \hat{\mathbf{g}}_0)$ and the variances $(\bar{\mathbf{u}}_0, \bar{\mathbf{v}}_0)$ at random according to the distributions p_U^0 and p_V^0.

Main loop: while $t < t_{\max}$, calculate following quantities:

$$\bar{\mathbf{X}}_{t+1} = \bar{\mathbf{u}}_t \bar{\mathbf{v}}_t^\top + \bar{\mathbf{u}}_t (\hat{\mathbf{v}}_t^2)^\top + \hat{\mathbf{u}}_t^2 \bar{\mathbf{v}}_t^\top$$

$$\hat{\mathbf{X}}_{t+1} = \hat{\mathbf{u}}_t \hat{\mathbf{v}}_t^\top$$

$$\bar{\mathbf{Z}}_{t+1} = \mathcal{A}^2(\bar{\mathbf{X}}_{t+1})$$

$$\hat{\mathbf{Z}}_{t+1} = \mathcal{A}(\hat{\mathbf{X}}_{t+1}) - \hat{\mathbf{g}}_t \odot \left(\bar{\mathbf{u}}_t \left(\mathcal{A}_U(\hat{\mathbf{v}}_t) \odot \mathcal{A}_U(\hat{\mathbf{v}}_{t-1}) \right)^\top + \left(\mathcal{A}_V(\hat{\mathbf{u}}_t) \odot \mathcal{A}_V(\hat{\mathbf{u}}_{t-1}) \right) \bar{\mathbf{v}}_t^\top \right)$$

$$\bar{\mathbf{g}}_{t+1} = \bar{g}^Y(\hat{\mathbf{Z}}_{t+1}, \bar{\mathbf{Z}}_{t+1})$$

$$\hat{\mathbf{g}}_{t+1} = \hat{g}^Y(\hat{\mathbf{Z}}_{t+1}, \bar{\mathbf{Z}}_{t+1})$$

$$\bar{\mathbf{U}}_{t+1} = -\left(\left[\mathcal{A}_U(\hat{\mathbf{v}}_t)^2 + \mathcal{A}_U^2(\bar{\mathbf{v}}_t) \right] \bar{\mathbf{g}}_{t+1} + \mathcal{A}_U^2(\bar{\mathbf{v}}_t) \hat{\mathbf{g}}_{t+1}^2 \right)^{-1}$$

$$\hat{\mathbf{U}}_{t+1} = \bar{\mathbf{U}}_{t+1} \odot \left(\mathcal{A}_U(\hat{\mathbf{v}}_t) \hat{\mathbf{g}}_{t+1} - \hat{\mathbf{u}}_t \odot \mathcal{A}_U(\hat{\mathbf{v}}_t)^2 \bar{\mathbf{g}}_{t+1} - \hat{\mathbf{u}}_{t-1} \odot \mathcal{A}_U^2(\bar{\mathbf{v}}_{t-1}) \hat{\mathbf{g}}_{t+1} \odot \hat{\mathbf{g}}_t \right)$$

$$\bar{\mathbf{u}}_{t+1} = \bar{f}^U(\hat{\mathbf{U}}_{t+1}, \bar{\mathbf{U}}_{t+1})$$

$$\hat{\mathbf{u}}_{t+1} = \hat{f}^U(\hat{\mathbf{U}}_{t+1}, \bar{\mathbf{U}}_{t+1})$$

$$\bar{\mathbf{V}}_{t+1} = -\left(\left[\mathcal{A}_V(\hat{\mathbf{u}}_t)^2 + \mathcal{A}_V^2(\bar{\mathbf{u}}_t) \right] \bar{\mathbf{g}}_{t+1} + \mathcal{A}_V^2(\bar{\mathbf{u}}_t) \hat{\mathbf{g}}_{t+1}^2 \right)^{-1}$$

$$\hat{\mathbf{V}}_{t+1} = \bar{\mathbf{V}}_{t+1} \odot \left(\mathcal{A}_V(\hat{\mathbf{u}}_t) \hat{\mathbf{g}}_{t+1} - \hat{\mathbf{v}}_t \odot \mathcal{A}_V(\hat{\mathbf{u}}_t)^2 \bar{\mathbf{g}}_{t+1} - \hat{\mathbf{v}}_{t-1} \odot \mathcal{A}_V^2(\bar{\mathbf{u}}_{t-1}) \hat{\mathbf{g}}_{t+1} \odot \hat{\mathbf{g}}_t \right)$$

$$\bar{\mathbf{v}}_{t+1} = \bar{f}^V(\hat{\mathbf{V}}_{t+1}, \bar{\mathbf{V}}_{t+1})$$

$$\hat{\mathbf{v}}_{t+1} = \hat{f}^V(\hat{\mathbf{V}}_{t+1}, \bar{\mathbf{V}}_{t+1})$$

Result : $(\hat{\mathbf{U}}, \hat{\mathbf{V}}, \hat{\mathbf{X}}, \hat{\mathbf{Z}})$ are the estimates for $(\mathbf{U}, \mathbf{V}, \mathbf{X}, \mathbf{Z})$ and $(\bar{\mathbf{U}}, \bar{\mathbf{V}}, \bar{\mathbf{X}}, \bar{\mathbf{Z}})$ are variances of these estimates.

This algorithm can be further simplified using the so-called Thouless-Andersen-Palmer (TAP) approximation introduced in the study of spin glasses [23]. We refer the reader to other works in which these simplifications are treated in details [6, 14] and only give the resulting algorithm 1, in which only local quantities and no messages are updated. This algorithm is a special case of the "P-BiG-AMP" algorithm, introduced in [6].

As its counterparts for generalized linear models (GAMP [20]) or matrix factorization [14,

24], algorithm 1 needs some adaptations that improve its convergence. One very simple damping scheme that allows to improve convergence (though not guaranteeing it) consists in damping a single variable:

$$\hat{\mathbf{U}}_{t+1} \leftarrow \beta \hat{\mathbf{U}}_{t+1} + (1-\beta)\hat{\mathbf{U}}_t, \tag{37}$$

with $\beta = 0.3$, applied right after the calculation of $\hat{\mathbf{U}}_{t+1}$. A more involved and better performing, adaptive damping strategy is presented in [25]. Notice that we defined the operators $\mathcal{A}_U$ and $\mathcal{A}_V$ used in algorithm 1 as linear applications $\mathcal{A}_U : \mathbb{R}^P \to \mathbb{R}^{L\times M}$ and $\mathcal{A}_V : \mathbb{R}^M \to \mathbb{R}^{L\times P}$ in (10,11): In the algorithm, we apply them row-wise on the matrices they act on.

III. ASYMPTOTIC ANALYSIS

The problem of low-rank matrix compressed sensing can be analyzed with statistical physics methods in the thermodynamic limit, i.e. when the dimensions of the signals M and P and of the measurements L go to infinity. R can remain finite or go to infinity as well. On the other hand, the ratios defined in (7) have to be fixed and finite. As in related inference problems, the analysis is done with the replica method. The resulting state evolution equations describe the behavior of the corresponding message-passing algorithm. In this section, we will focus on the derivation of the replica analysis that results in a simple set of state evolution equations. The analysis is very similar to the one of related inference problems [12–14, 26].

A. Replica analysis: free entropy

Treating an inference problem as a statistical physics problem reduces to writing an energy function corresponding to the problem and studying the free energy of the system. We are thus interested in calculating a partition function. Here, the relevant partition function is the normalization constant of the probability distribution (21):

$$\mathcal{Z}(\mathbf{Y},\mathbf{A}) = \int \mathrm{d}\mathbf{U}\, p_U(\mathbf{U}) \int \mathrm{d}\mathbf{V}\, p_V(\mathbf{V}) \int \mathrm{d}\mathbf{z} p_{Y|Z}\left(\mathbf{Y}|\mathbf{z}\right) \delta\left[\mathbf{z} - \mathcal{A}(\mathbf{U}\mathbf{V}^\top)\right]. \tag{38}$$

The free entropy $\log \mathcal{Z}(\mathbf{Y},\mathbf{A})$ of a given instance can be calculated from the marginals calculated by the belief propagation equations.

However, one can also be interested in the average free entropy of this problem. In order to do this, one needs to average $\log \mathcal{Z}(\mathbf{Y}, \mathbf{A})$ over all possible realizations of $\mathbf{A}$ and $\mathbf{Y}$, for which we use the replica method [12, 22]. It uses the identity

$$\langle \log \mathcal{Z} \rangle = \lim_{n \to 0} \frac{\partial}{\partial n} \langle \mathcal{Z}^n \rangle \tag{39}$$

where $\langle \cdot \rangle$ denotes the average over $\mathbf{A}$ and $\mathbf{Y}$, and relies on the fact that an expression for $\mathcal{Z}^n$ can be found for integer n. This expression is then used for calculating the $n \to 0$ limit in (39). Though not rigorous, this method has proven to give correct results in a wide range of problems [12, 22].

Let us therefore start by calculating

$$\mathcal{Z}(\mathbf{Y}, \mathbf{A})^n = \int \prod_{a=1}^{n} \left\{ \mathrm{d}\mathbf{U}^a \, p_U(\mathbf{U}^a) \mathrm{d}\mathbf{V}^a \, p_V(\mathbf{V}^a) \mathrm{d}\mathbf{z}^a p_{Y|Z}\left(\mathbf{Y}|\mathbf{z}^a\right) \delta\left[\mathbf{z}^a - \mathcal{A}(\mathbf{U}^a(\mathbf{V}^a)^\top)\right] \right\} \tag{40}$$

and its average with respect to the realizations of $\mathbf{Y}$, generated by $\mathbf{U}^0$, $\mathbf{V}^0$ and $\mathcal{A}$:

$$\begin{aligned} \langle \mathcal{Z}^n \rangle = \int & \mathrm{d}\mathbf{U}^0 \, p_U^0(\mathbf{U}^0) \mathrm{d}\mathbf{V}^0 \, p_V^0(\mathbf{V}^0) \mathrm{d}\mathbf{A} \, p_A^0(\mathbf{A}) \mathrm{d}\mathbf{Y} \\ & \mathrm{d}\mathbf{z}^0 p_{Y|Z}(\mathbf{Y}|\mathbf{z}^0) \delta\left[\mathbf{z}^0 - \mathcal{A}(\mathbf{U}^0(\mathbf{V}^0)^\top)\right] \mathcal{Z}(\mathbf{Y}, \mathbf{A})^n. \end{aligned} \tag{41}$$

The indices a represent so-called replicas of the system and are initially independent from each other. Carrying on the calculation requires to couple them. To be more precise, each variable $z_l^a = [\mathcal{A}(\mathbf{U}^a(\mathbf{V}^a)^\top)]_l$ is the sum of a large number of independent random variables and can therefore be approximated as a Gaussian random variable. This was done in section II already and allows again to considerably reduce the number of integrals caused by the averaging over $\mathbf{A}$. However, z_l^a and z_l^b are not independent, as they are produced with the same operator $\mathcal{A}$. We show in appendix B that $\mathbf{z}_l \equiv (z_l^0 \dots z_l^n)$ is a multivariate random Gaussian variable with mean 0 and covariance matrix $\mathbf{Q}_z \equiv \mathbf{Q}_u \odot \mathbf{Q}_v$, where the elements of the matrices $\mathbf{Q}_u$ and $\mathbf{Q}_v$ are given by:

$$Q_u^{ab} \equiv \frac{1}{M} \sum_\mu u_\mu^a u_\mu^b, \qquad\qquad Q_v^{ab} \equiv \frac{1}{P} \sum_p v_p^a v_p^b. \tag{42}$$

As in (41), these quantities can be anything, we have to integrate over them, such that

$$\langle \mathcal{Z}^n \rangle = \int \mathrm{d}\mathbf{Q}_u \left[\int \prod_a \mathrm{d}\mathbf{U}^a \, p_U^a(\mathbf{U}^a) \prod_{\substack{s \\ a \leq b}} \delta \left(M Q_u^{ab} - \sum_\mu u_{\mu s}^a u_{\mu s}^b \right) \right]$$
$$\int \mathrm{d}\mathbf{Q}_v \left[\int \prod_a \mathrm{d}\mathbf{V}^a \, p_V^a(\mathbf{V}^a) \prod_{\substack{s \\ a \leq b}} \delta \left(P Q_v^{ab} - \sum_p v_{ps}^a v_{ps}^b \right) \right]$$
$$\prod_{l=1}^{L} \left[\int \mathrm{d}\mathbf{z}_l \mathcal{N}(\mathbf{z}_l; 0, \mathbf{Q}_z) \int \mathrm{d}y_l p_{Y|Z}^0(y_l|z_l^0) \prod_{a=1}^{n} p_{Y|Z}(y_l|z_l^a) \right]. \tag{43}$$

Here, we use the convention that $p_U^a = p_U$ if $a \neq 0$. We now see that the different replicas are coupled via $\mathbf{Q}_u$ and $\mathbf{Q}_v$ in the first two lines. As we did with $\mathbf{z}_l$, we now introduce the vector $\mathbf{u}_{ps} = (u_{ps}^0 \dots u_{ps}^n)$ (similarly for $\mathbf{v}_{\mu s}$) and we use the integral representation of the δ function, introducing the conjugate variables $\hat{\mathbf{Q}}_u$ and $\hat{\mathbf{Q}}_v$ (details in appendix B), which leads to

$$\langle \mathcal{Z}^n \rangle = \int \mathrm{d}\mathbf{Q}_u \mathrm{d}\hat{\mathbf{Q}}_u e^{-\frac{MR}{2}\mathrm{Tr}(\mathbf{Q}_u \hat{\mathbf{Q}}_u)} \left[\prod_{\mu s} \mathrm{d}\mathbf{u}_{\mu s} p_u(\mathbf{u}_{\mu s}) e^{\frac{1}{2}\mathbf{u}_{\mu s}^\top \hat{\mathbf{Q}}_u \mathbf{u}_{\mu s}} \right]$$
$$\int \mathrm{d}\mathbf{Q}_v \mathrm{d}\hat{\mathbf{Q}}_v e^{-\frac{PR}{2}\mathrm{Tr}(\mathbf{Q}_v \hat{\mathbf{Q}}_v)} \left[\prod_{ps} \mathrm{d}\mathbf{v}_{ps} p_v(\mathbf{v}_{ps}) e^{\frac{1}{2}\mathbf{v}_{ps}^\top \hat{\mathbf{Q}}_v \mathbf{v}_{ps}} \right]$$
$$\prod_{l=1}^{L} \left[\int \mathrm{d}\mathbf{z}_l \mathcal{N}(\mathbf{z}_l; 0, \mathbf{Q}_z) \int \mathrm{d}y_l p_{Y|Z}^0(y_l|z_l^0) \prod_{a=1}^{n} p_{Y|Z}(y_l|z_l^a) \right]. \tag{44}$$

Finally, we assume the distributions of $u_{\mu s}$'s, v_{ps}'s and y_l's are the same for every coordinate. Using the notations

$$p_u(\mathbf{u}) = p_U^0(u^0) \prod_{a>0} p_U(u^a), \quad p_v(\mathbf{v}) = p_V^0(v^0) \prod_{a>0} p_V(v^a), \quad p_{y|\mathbf{z}}(y|\mathbf{z}) = p_{Y|Z}^0(y|z^0) \prod_{a>0} p_{Y|Z}(y|z^a), \tag{45}$$

this leads to:

$$\langle \mathcal{Z}^n \rangle = \int \mathrm{d}\mathbf{Q}_u \mathrm{d}\hat{\mathbf{Q}}_u e^{-\frac{MR}{2}\mathrm{Tr}(\mathbf{Q}_u \hat{\mathbf{Q}}_u)} \left[\mathrm{d}\mathbf{u} p_u(\mathbf{u}) e^{\frac{1}{2}\mathbf{u}^\top \hat{\mathbf{Q}}_u \mathbf{u}} \right]^{RM}$$
$$\int \mathrm{d}\mathbf{Q}_v \mathrm{d}\hat{\mathbf{Q}}_v e^{-\frac{PR}{2}\mathrm{Tr}(\mathbf{Q}_v \hat{\mathbf{Q}}_v)} \left[\mathrm{d}\mathbf{v} p_v(\mathbf{v}) e^{\frac{1}{2}\mathbf{v}^\top \hat{\mathbf{Q}}_v \mathbf{v}} \right]^{RP}$$
$$\left[\int \mathrm{d}\mathbf{z} \mathcal{N}(\mathbf{z}; 0, \mathbf{Q}_z) \int \mathrm{d}y p_{y|\mathbf{z}}(y|\mathbf{z}) \right]^L. \tag{46}$$

In the "thermodynamic" limit, we take M, P and L going to infinity with constant ratios. This motivates us to rewrite the last equation under the form

$$\langle \mathcal{Z}^n \rangle = \int \mathrm{d}\mathbf{Q}_u \hat{\mathbf{Q}}_u \mathbf{Q}_v \hat{\mathbf{Q}}_v e^{-MR\left[S_n(\mathbf{Q}_u, \hat{\mathbf{Q}}_u, \mathbf{Q}_v, \hat{\mathbf{Q}}_v)\right]} \tag{47}$$

and to use the saddle point method, according to which

$$\log\left(\langle \mathcal{Z}^n \rangle\right) = -MR \min_{\mathbf{Q}_u, \hat{\mathbf{Q}}_u, \mathbf{Q}_v, \hat{\mathbf{Q}}_v} S_n(\mathbf{Q}_u, \hat{\mathbf{Q}}_u, \mathbf{Q}_v, \hat{\mathbf{Q}}_v). \tag{48}$$

We are therefore left with a minimization problem over the space of the matrices $\mathbf{Q}_u, \hat{\mathbf{Q}}_u, \mathbf{Q}_v$ and $\hat{\mathbf{Q}}_v$, representing $2(n+1)(n+2)$ parameters (as the matrices are symmetric).

B. Replica symmetric assumption

The idea of the replica symmetric assumption is that the n replicas introduced in (40) are all equivalent, as they are purely a mathematical manipulation. Based on this, we make the assumption that a sensible matrix $\mathbf{Q}_u$ does not make any distinction between the n introduced replicas. We therefore parametrize $\mathbf{Q}_u$ and $\hat{\mathbf{Q}}_u$ in the following way:

$$\mathbf{Q}_u = \left(\begin{array}{c|ccc} Q_u^0 & m_u & \cdots & m_u \\ \hline m_u & Q_u & \cdots & q_u \\ \vdots & \vdots & \ddots & \vdots \\ m_u & q_u & \cdots & Q_u \end{array}\right) \qquad \hat{\mathbf{Q}}_u = \left(\begin{array}{c|ccc} \hat{Q}_u^0 & \hat{m}_u & \cdots & \hat{m}_u \\ \hline \hat{m}_u & \hat{Q}_u & \cdots & \hat{q}_u \\ \vdots & \vdots & \ddots & \vdots \\ \hat{m}_u & \hat{q}_u & \cdots & \hat{Q}_u \end{array}\right) \tag{49}$$

and similarly for $\mathbf{Q}_v$, allowing to be left with 16 instead of $2(n+1)(n+2)$ parameters over which to perform the extremization (48). Furthermore, Q_u^0 and Q_v^0 are in fact known, as they are the second moments of the priors p_U^0 and p_V^0, and therefore we set

$$\hat{Q}_u^0 = 0, \qquad \hat{Q}_v^0 = 0, \tag{50}$$

and thus the extremization is only over 12 variables: $(m_u, \hat{m}_u, q_u, \hat{q}_u, Q_u, \hat{Q}_u)$ and $(m_v, \hat{m}_v, q_v, \hat{q}_v, Q_v, \hat{Q}_v)$.

Let us now look in more details at the function S_n to extremize:

$$\begin{aligned} S_n(\mathbf{Q}_u, \mathbf{Q}_v, \hat{\mathbf{Q}}_u, \hat{\mathbf{Q}}_v) \equiv & \left[\frac{1}{2}\mathrm{Tr}\mathbf{Q}_u\hat{\mathbf{Q}}_u - \log\left(\int \mathrm{d}\mathbf{u} p_u(\mathbf{u}) e^{\frac{1}{2}\mathbf{u}^\top \hat{\mathbf{Q}}_u \mathbf{u}}\right)\right] \\ & + \frac{M}{P}\left[\frac{1}{2}\mathrm{Tr}\mathbf{Q}_v\hat{\mathbf{Q}}_v - \log\left(\int \mathrm{d}\mathbf{v} p_u(\mathbf{v}) e^{\frac{1}{2}\mathbf{v}^\top \hat{\mathbf{Q}}_v \mathbf{v}}\right)\right] \\ & - \frac{L}{RP}\log\left(\int \mathrm{d}\mathbf{z}\mathcal{N}(\mathbf{z}; 0, \mathbf{Q}_z)\int \mathrm{d}y p_{y|\mathbf{z}}(y|\mathbf{z})\right). \end{aligned} \tag{51}$$

Thanks to the parametrization (49), the different terms have simple expressions. The traces can simply be written as

$$\mathrm{Tr}\mathbf{Q}_u\hat{\mathbf{Q}}_u = 2nm\hat{m}_u + nQ_u\hat{Q}_u + n(n-1)q_u\hat{q}_u, \tag{52}$$

while we can use that

$$\mathbf{u}^\top\hat{\mathbf{Q}}_u\mathbf{u} = \hat{Q}_u^0(u^0)^2 + (\hat{Q}_u - \hat{q}_u)\sum_{a>0}(u^a)^2 + \hat{q}_u(\sum_{a>0}u^a)^2 + 2\hat{m}_u u^0\sum_{a>0}u^a \tag{53}$$

and the Gaussian transformation $e^{\lambda\alpha^2} = \int \mathrm{D}x\, e^{\alpha\sqrt{2\lambda}x}$ in order to rewrite the integral $\int \mathrm{d}\mathbf{u}P_u(\mathbf{u})e^{\frac{1}{2}\mathbf{u}^\top\hat{\mathbf{Q}}_u\mathbf{u}}$ as

$$\mathcal{I}_U^n = \int \mathrm{D}t \int \mathrm{d}u^0\, p_U^0(u^0)\left[\int \mathrm{d}u\, p_U(u)e^{\frac{\hat{Q}_u-\hat{q}_u}{2}u^2+(t\sqrt{\hat{q}_u}+\hat{m}_u u^0)u}\right]^n. \tag{54}$$

The third line in (51) can be simplified as well. The first step consists in writing the coupled Gaussian random variables $z^0\dots z^n$ as a function of n independent, standard Gaussian random variables x^a ($a\in[1,n]$) and one additional standard random variable t that couples them all:

$$z^0 = \sqrt{Q_z^0 - \frac{m_z^2}{q_z}}\,x^0 + \frac{m_z}{\sqrt{q_z}}\,t, \qquad\qquad z^a = \sqrt{Q_z - q_z}\,x^a + \sqrt{q_z}\,t. \tag{55}$$

Making the change of variables in the integral, we obtain the following expression for $\int \mathrm{d}\mathbf{z}\mathcal{N}(\mathbf{z};0,\mathbf{Q}_z)\int \mathrm{d}y P_{y|z}(y|\mathbf{z})$:

$$\mathcal{I}_Z^n = \int \mathrm{d}y \int \mathrm{D}t \left[\int \mathrm{D}x^0\, p_{Y|Z}^0(y|\sqrt{Q_z^0 - \frac{m_z^2}{q_z}}\,x^0 + \frac{m_z}{\sqrt{q_z}}\,t)\right]\left[\int \mathrm{D}x\, p_{Y|Z}(y^0|\sqrt{Q_z - q_z}\,x + \sqrt{q_z}\,t)\right]^n. \tag{56}$$

Looking back at the replica trick (39), we have to study the quantity $\lim_{n\to 0}\frac{\partial}{\partial n}S_n$ and therefore the quantities

$$\mathcal{I}_U(\hat{\mathbf{Q}}) = \lim_{n\to 0}\frac{\partial}{\partial n}\log\mathcal{I}_U^n = \int \mathrm{D}t\left[\int \mathrm{d}u^0\, p_U^0(u^0)\log\left[\int \mathrm{d}u\, p_U(u)e^{\frac{\hat{Q}-\hat{q}}{2}u^2+(t\sqrt{\hat{q}}+\hat{m}u^0)u}\right]\right], \tag{57}$$

as well as its equivalent $\mathcal{I}_V$ (obtained by replacing all us by vs in (57)) and

$$\mathcal{I}_Z(\mathbf{Q}) = \lim_{n\to 0}\frac{\partial}{\partial n}\log\mathcal{I}_Z^n = \int \mathrm{d}y \int \mathrm{D}t\, f_0^{Y,0}(\frac{m}{\sqrt{q}}t, Q^0 - \frac{m^2}{q})\log\left(f_0^Y(\sqrt{q}t, Q-q)\right), \tag{58}$$

where $f_i^{Y,0} \equiv f_i^{p^0_{Y|Z}}$. In the end, we obtain the free entropy ϕ as an extremum

$$\phi = -\operatorname{extr}\left\{\left(m_u\hat{m}_u + \frac{1}{2}Q_u\hat{Q}_u - \frac{1}{2}q_u\hat{q}_u - \mathcal{I}_U(\hat{\mathbf{Q}}_u)\right)\right.$$
$$\left. + \frac{M}{P}\left(m_v\hat{m}_v + \frac{1}{2}Q_v\hat{Q}_v - \frac{1}{2}q_v\hat{q}_v - \mathcal{I}_V(\hat{\mathbf{Q}}_v)\right) - \frac{L}{RP}\mathcal{I}_Z(\mathbf{Q}_u \odot \mathbf{Q}_v)\right\} \tag{59}$$

over a set of 12 variables. Note that the shift from a minimum in (48) to an extremum in the equation above is a consequence to the hazardous $n \to 0$ limit in the replica method.

1. *Equivalence to generalized matrix factorization*

It is interesting to notice that if $L = MP$ and $R = O(M)$, this free entropy is the same as in generalized matrix factorization [14]. This is not an entirely obvious fact, as the two problems are different and that they are identical only if $\mathcal{A}$ is the identity: in generalized matrix factorization, $\mathbf{Z} = \mathbf{X}$.

In order to perform the theoretical analysis of generalized matrix *factorization* as in [14], it is important to take the limit $R \to \infty$. In fact, it is this limit that ensures that each entry of $\mathbf{Z}$ is the sum of a large number of random variables, which allows to consider that it has a Gaussian distribution. This is a condition both in the derivation of the message-passing algorithm and in the replica analysis. For that reason, generalized matrix factorization with finite R leads to different algorithms and theoretical bounds [27, 28]. However, in matrix compressed sensing, the mixing of coefficients with $\mathcal{A}$ ensures that even if $R = 1$, each element of $\mathbf{Z}$ can be considered to have a Gaussian distribution. Thanks to this, both the algorithm and the analysis are the same, independently of R. Note that it would be natural to write the free entropy (59) with no explicit R-dependence by introducing a global measurement ratio $\alpha \equiv \frac{L}{R(M+P)}$.

Let us examine the case in which $L = MP$ and $R = O(M)$ and the two problems are strictly equivalent. What differentiates the generalized matrix compressed sensing from the generalized matrix factorization case is that $\mathcal{A}$ is not the identity. However, as $\mathcal{A}$'s coefficients are Gaussian i.i.d. , it is with high probability a bijection when $L = MP$, and in this sense the mixing step does not introduce any further difficulty into the problem compared to matrix factorization. If $L > MP$, matrix compressed sensing is not "compressive" and therefore easier than the corresponding matrix factorization problem, because more measurements are available. If $L < MP$, matrix compressed sensing is "compressive".

C. State evolution equations

In the previous section, we have derived an expression of the free entropy as an extremum of an action function over a set of parameters. In this section, we find self-consistent equations that hold at the values of these parameters extremizing the action. Furthermore, these self-consistent equations can be iterated in order to numerically obtain the extrema of the action.

In order to find the extremum in (59), we simply set all the partial derivatives of ϕ to 0. The difficult part is finding expressions for the derivatives of the integrals $\mathcal{I}_U, \mathcal{I}_V$ and $\mathcal{I}_Z$, which we detail here. First we do the calculation for $\mathcal{I}_U$.

$$\begin{aligned}
\frac{\partial}{\partial \hat{Q}_u}\mathcal{I}_U(\hat{\mathbf{Q}}_u) &= \int \mathrm{D}t \int \mathrm{d}u^0 p_U^0(u^0) \frac{\int \mathrm{d}u\, p_U(u) u^2 e^{\frac{\hat{Q}_u-\hat{q}_u}{2}u^2+(t\sqrt{\hat{q}_u}+\hat{m}_u u^0)u}}{\int \mathrm{d}u\, p_U(u) e^{\frac{\hat{Q}_u-\hat{q}_u}{2}u^2+(t\sqrt{\hat{q}_u}+\hat{m}_u u^0)u}}, \\
\frac{\partial}{\partial \hat{q}_u}\mathcal{I}_U(\hat{\mathbf{Q}}_u) &= \int \mathrm{D}t \int \mathrm{d}u^0 p_U^0(u^0) \frac{\int \mathrm{d}u\, p_U(u) \left(-\frac{u^2}{2}+\frac{tu}{2\sqrt{\hat{q}_u}}\right) e^{\frac{\hat{Q}_u-\hat{q}_u}{2}u^2+(t\sqrt{\hat{q}_u}+\hat{m}_u u^0)u}}{\int \mathrm{d}u\, p_U(u) e^{\frac{\hat{Q}_u-\hat{q}_u}{2}u^2+(t\sqrt{\hat{q}_u}+\hat{m}_u u^0)u}}, \\
\frac{\partial}{\partial \hat{m}_u}\mathcal{I}_U(\hat{\mathbf{Q}}_u) &= \int \mathrm{D}t \int \mathrm{d}u^0 u^0 p_U^0(u^0) \frac{\int \mathrm{d}u\, p_U(u)\, u\, e^{\frac{\hat{Q}_u-\hat{q}_u}{2}u^2+(t\sqrt{\hat{q}_u}+\hat{m}_u u^0)u}}{\int \mathrm{d}u\, p_U(u) e^{\frac{\hat{Q}_u-\hat{q}_u}{2}u^2+(t\sqrt{\hat{q}_u}+\hat{m}_u u^0)u}}.
\end{aligned} \tag{60}$$

If we inject these expressions into the extremization equations of ϕ with respect to $\hat{Q}_u, \hat{q}_u, \hat{m}_u$ and use the update functions defined in (14)-(16), we obtain

$$m_u = \int \mathrm{D}t \int \mathrm{d}u^0\, u^0 p_U^0(u^0) \hat{f}^U\left(\frac{\sqrt{\hat{q}_u}t+\hat{m}_u u^0}{\hat{q}_u-\hat{Q}_u}, \frac{1}{\hat{q}_u-\hat{Q}_u}\right), \tag{61}$$

$$Q_u - q_u = \frac{1}{\sqrt{\hat{q}_u}} \int \mathrm{D}t\; t \int \mathrm{d}u^0 p_U^0(u^0) \hat{f}^U\left(\frac{\sqrt{\hat{q}_u}t+\hat{m}_u u^0}{\hat{q}_u-\hat{Q}_u}, \frac{1}{\hat{q}_u-\hat{Q}_u}\right), \tag{62}$$

$$Q_u = \int \mathrm{D}t \int \mathrm{d}u^0 p_U^0(u^0) \left[\bar{f}^U\left(\frac{\sqrt{\hat{q}_u}t+\hat{m}_u u^0}{\hat{q}_u-\hat{Q}_u}, \frac{1}{\hat{q}_u-\hat{Q}_u}\right) + \left(\hat{f}^U\left(\frac{\sqrt{\hat{q}_u}t+\hat{m}_u u^0}{\hat{q}_u-\hat{Q}_u}, \frac{1}{\hat{q}_u-\hat{Q}_u}\right)\right)^2\right]. \tag{63}$$

These equations can be further simplified by using the transformation $t \leftarrow t + \frac{\hat{m}}{\sqrt{\hat{q}}}u^0$ and integrating by part eq (62):

$$m_u = \sqrt{\frac{\hat{q}_u}{\hat{m}_u^2}} \int \mathrm{d}t\, f_1^{U,0}\left(\frac{\sqrt{\hat{q}_u}}{\hat{m}_u}t, \frac{\hat{q}_u}{\hat{m}_u^2}\right) \hat{f}^U\left(\frac{\sqrt{\hat{q}_u}t}{\hat{q}_u-\hat{Q}_u}, \frac{1}{\hat{q}_u-\hat{Q}_u}\right), \tag{64}$$

$$Q_u - q_u = \sqrt{\frac{\hat{q}_u}{\hat{m}_u^2}} \int \mathrm{d}t\, f_0^{U,0}\left(\frac{\sqrt{\hat{q}_u}}{\hat{m}_u}t, \frac{\hat{q}_u}{\hat{m}_u^2}\right) \bar{f}^U\left(\frac{\sqrt{\hat{q}_u}t}{\hat{q}_u-\hat{Q}_u}, \frac{1}{\hat{q}_u-\hat{Q}_u}\right), \tag{65}$$

$$q_u = \sqrt{\frac{\hat{q}_u}{\hat{m}_u^2}} \int \mathrm{d}t\, f_0^{U,0}\left(\frac{\sqrt{\hat{q}_u}}{\hat{m}_u}t, \frac{\hat{q}_u}{\hat{m}_u^2}\right) \left[\hat{f}^U\left(\frac{\sqrt{\hat{q}_u}t}{\hat{q}_u-\hat{Q}_u}, \frac{1}{\hat{q}_u-\hat{Q}_u}\right)\right]^2, \tag{66}$$

and the same equations hold replacing u by v.

Let us now come to the derivatives of $\mathcal{I}_Z$. To calculate them, we use the identity (19), taking $s = q$ or $s = \frac{m^2}{q}$. After an integration by parts, we obtain

$$\frac{\partial}{\partial m}\mathcal{I}_Z(\mathbf{Q}) = \frac{1}{m}\int \mathrm{d}y \int \mathrm{D}t \, \frac{\left[\frac{\partial}{\partial t} f_0^{Y,0}(\frac{m}{\sqrt{q}}t, Q^0 - \frac{m^2}{q})\right]\left[\frac{\partial}{\partial t} f_0^{Y}(\sqrt{q}t, Q - q)\right]}{f_0^Y(\sqrt{q}t, Q-q)}, \tag{67}$$

$$\frac{\partial}{\partial q}\mathcal{I}_Z(\mathbf{Q}) = -\frac{1}{2q}\int \mathrm{d}y \int \mathrm{D}t \, \left[\frac{\frac{\partial}{\partial t} f_0^{Y}(\sqrt{q}t, Q - q)}{f_0^Y(\sqrt{q}t, Q-q)}\right]^2 f_0^{Y,0}(\frac{m}{\sqrt{q}}t, Q^0 - \frac{m^2}{q}), \tag{68}$$

$$\frac{\partial}{\partial Q}\mathcal{I}_Z(\mathbf{Q}) = \int \mathrm{d}y \int \mathrm{D}t \, f_0^{Y,0}(\frac{m}{\sqrt{q}}t, Q^0 - \frac{m^2}{q}) \left[\frac{\frac{\partial}{\partial Q} f_0^{Y}(\sqrt{q}t, Q - q)}{f_0^Y(\sqrt{q}t, Q-q)}\right]. \tag{69}$$

Injecting these expressions into the extremization equations of ϕ with respect to Q, q, m, we obtain

$$\hat{m} = \frac{1}{m}\int \mathrm{d}y \int \mathrm{D}t \, \frac{\left[\frac{\partial}{\partial t} f_0^{Y,0}(\frac{m}{\sqrt{q}}t, Q^0 - \frac{m^2}{q})\right]\left[\frac{\partial}{\partial t} f_0^{Y}(\sqrt{q}t, Q - q)\right]}{f_0^Y(\sqrt{q}t, Q-q)}, \tag{70}$$

$$\hat{q} = \frac{1}{q}\int \mathrm{d}y \int \mathrm{D}t \, \left[\frac{\frac{\partial}{\partial t} f_0^{Y}(\sqrt{q}t, Q - q)}{f_0^Y(\sqrt{q}t, Q-q)}\right]^2 f_0^{Y,0}(\frac{m}{\sqrt{q}}t, Q^0 - \frac{m^2}{q}), \tag{71}$$

$$\hat{Q} = 2\int \mathrm{d}y \int \mathrm{D}t \, f_0^{Y,0}(\frac{m}{\sqrt{q}}t, Q^0 - \frac{m^2}{q}) \left[\frac{\frac{\partial}{\partial Q} f_0^{Y}(\sqrt{q}t, Q - q)}{f_0^Y(\sqrt{q}t, Q-q)}\right], \tag{72}$$

and remembering that $m = m_u m_v, q = q_u q_v$, $Q = Q_u Q_v$ and the definitions (7):

$$\hat{m}_u = \alpha_U m_v \hat{m}, \qquad \hat{q}_u = \alpha_U q_v \hat{q}, \qquad \hat{Q}_u = \alpha_U Q_v \hat{Q}, \tag{73}$$

$$\hat{m}_v = \alpha_V m_u \hat{m}, \qquad \hat{q}_v = \alpha_V q_u \hat{q}, \qquad \hat{Q}_v = \alpha_V Q_u \hat{Q}. \tag{74}$$

The equations (64,65,66) along with their equivalents for v, the equations (70,71,72) and (73,74) constitute a closed set of equations that hold at the extrema of ϕ in equation (59).

When they are iterated, they constitute the so-called state evolution equations. These can also be obtained by the analysis of the BP algorithm and are known to accurately describe the algorithm's behavior when the replica symmetric hypothesis is indeed correct.

As noted before, if $L = MP$, these state evolution equations are identical to the ones in matrix factorization [14]. Therefore, they reduce to the state evolution of GAMP when U is known, which corresponds to fixing $m_u = q_u = Q_u = Q_u^0$ in the equations.

D. Bayes-optimal analysis

Until now, we have not supposed exact knowledge of the true signal distributions and of the true measurement channel. When this is the case, the state evolution equations greatly simplify because of the so-called Nishimori conditions [29]. In our case, these ensure that following equalities hold:

$$Q = Q^0, \quad \hat{Q} = 0, \quad m = q, \quad \hat{m} = \hat{q} \tag{75}$$

both for u and v. Then, we only need to keep track of the variables $(m_u, \hat{m}_u, m_v, \hat{m}_v)$, and the state evolution is obtained by choosing initial values for (m_u^0, m_v^0) and iterating for $i \geq 0$ the equations

$$\hat{m}^{i+1} = \frac{1}{m_u^i m_v^i} \int \mathrm{d}y \int \mathrm{D}t \, \frac{\left[\frac{\partial}{\partial t} f_0^Y(\sqrt{m_u^i m_v^i}t, Q_u^0 Q_v^0 - m_u^i m_v^i)\right]^2}{f_0^Y(\sqrt{m_u^i m_v^i}t, Q_u^0 Q_v^0 - m_u^i m_v^i)}, \tag{76}$$

$$m_u^{i+1} = \frac{1}{\sqrt{\alpha_U m_v^i \hat{m}^{i+1}}} \int \mathrm{d}t \frac{\left[f_1^U(\frac{t}{\sqrt{\alpha_U m_v^i \hat{m}^{i+1}}}, \frac{1}{\alpha_U m_v^i \hat{m}^{i+1}})\right]^2}{f_0^U(\frac{t}{\sqrt{\alpha_U m_v^i \hat{m}^{i+1}}}, \frac{1}{\alpha_U m_v^i \hat{m}^{i+1}})}, \tag{77}$$

$$m_v^{i+1} = \frac{1}{\sqrt{\alpha_V m_u^i \hat{m}^{i+1}}} \int \mathrm{d}t \frac{\left[f_1^V(\frac{t}{\sqrt{\alpha_V m_u^i \hat{m}^{i+1}}}, \frac{1}{\alpha_V m_u^i \hat{m}^{i+1}})\right]^2}{f_0^V(\frac{t}{\sqrt{\alpha_V m_u^i \hat{m}^{i+1}}}, \frac{1}{\alpha_V m_u^i \hat{m}^{i+1}})}, \tag{78}$$

until convergence. From m_u and m_v, one can simply deduce the mean squared errors by the following relations:

$$\mathrm{MSE}_u = Q_u^0 - m_u, \qquad \mathrm{MSE}_v = Q_v^0 - m_v, \qquad \mathrm{MSE}_x = Q_u^0 Q_v^0 - m_u m_v. \tag{79}$$

The initialization values $(m_u, \hat{m}_u, m_v, \hat{m}_v)$ indicate how close to the solution the algorithm is at initialization. In case of a random initialization of the algorithm, the expected initial overlaps m_u^0 and m_v^0 are of order $1/M$ and $1/P$ respectively, and they should therefore be set to these values (or less) in the state evolution equations.

Note that state evolution run with matching priors without imposing the Nishimori conditions (75) should in principle give the exact same results as the Bayes-optimal state evolution analysis presented above, and thus naturally follow the so-called "Nishimori line" defined by (75). However, as shown in [30], the Nishimori line can be unstable: In that case, numerical fluctuations around it will be amplified under iterations of state evolution that will

thus give a different result than its counterpart with imposed Nishimori conditions. This instability of the Nishimori line seems to be the reason why algorithm 1 as well as others of the same type do not converge without damping of the variables.

IV. CASE STUDY

In this section, we focus on one specific setting for which the state evolution equations are practical to implement. An analysis of their fixed points leads to an understanding of different phases and of the phase transitions between them.

We look at the setting in which both $\mathbf{U}$ and $\mathbf{V}$ follow a Bernoulli-Gauss distribution:

$$p_U(u) = (1-\rho_u)\delta(u) + \rho_u \mathcal{N}(u;0,1), \tag{80}$$

$$p_V(v) = (1-\rho_v)\delta(v) + \rho_v \mathcal{N}(v;0,1), \tag{81}$$

and the measurements are taken through an additive white Gaussian noise (AWGN) channel:

$$\forall l \in [1,L], \quad Y_l = [\mathcal{A}(\mathbf{U}\mathbf{V}^T)]_l + \xi_l, \qquad \text{with} \quad \xi_l \sim \mathcal{N}(\xi_l;0,\Delta). \tag{82}$$

Note that most previous works [9, 31–33] consider this channel. For the AWGN channel the equation (76) has a simple analytical expression:

$$\hat{m}^{i+1} = \frac{1}{\Delta + \rho_u\rho_v - m_u^i m_v^i}. \tag{83}$$

Further simplifying the setting to the special case $M = P$ and $\rho_u = \rho_v = \rho$, the Bayes optimal state evolution equations (76-78) can be written as one single equation

$$m = \sqrt{\frac{\Delta+\rho^2-m^2}{\alpha_u m}} \int \mathrm{d}t \frac{\left[f_1^U(\sqrt{\frac{\Delta+\rho^2-m^2}{\alpha_u m}}t, \frac{\Delta+\rho^2-m^2}{\alpha_u m})\right]^2}{f_0(\sqrt{\frac{\Delta+\rho^2-m^2}{\alpha_u m}}t, \frac{\Delta+\rho^2-m^2}{\alpha_u m})}, \tag{84}$$

in which the iteration-time indices of m, i (left hand side) and $i-1$ (right hand side), are left out for better legibility. We can define a global measurement rate

$$\alpha \equiv \frac{L}{2MR} = \frac{\alpha_u}{2}, \tag{85}$$

which is the natural quantity to compare ρ to.

A. Phases and phase transitions

As in compressed sensing or in matrix factorization, the analysis of the free entropy and state evolution equations reveals the existence of different phases in which the difficulty of the problem is different. In our case study, the free entropy ϕ has the following expression:

$$\phi(m) = -m\hat{m} - \frac{\alpha}{4}\log\left(2\pi\left(\Delta + \rho^2 - m^2\right)\right) + \frac{2}{\sqrt{\hat{m}}}\int \mathrm{d}t f_0^U\left(\frac{t}{\sqrt{\hat{m}}}, \frac{1}{\hat{m}}\right)\left[\frac{t^2}{2} + \log\left(\sqrt{\frac{2\pi}{\hat{m}}} f_0^U\left(\frac{t}{\sqrt{\hat{m}}}, \frac{1}{\hat{m}}\right)\right)\right] \tag{86}$$

with

$$\hat{m} = \frac{1}{\Delta + \rho^2 - m^2}. \tag{87}$$

The integral can best be numerically evaluated replacing $\int$ by $2\left(\int_0^{20} + \int_{20}^{20\sqrt{1+\hat{m}}}\right)$, which allows a reliable numerical evaluation for all possible values of $\hat{m}$.

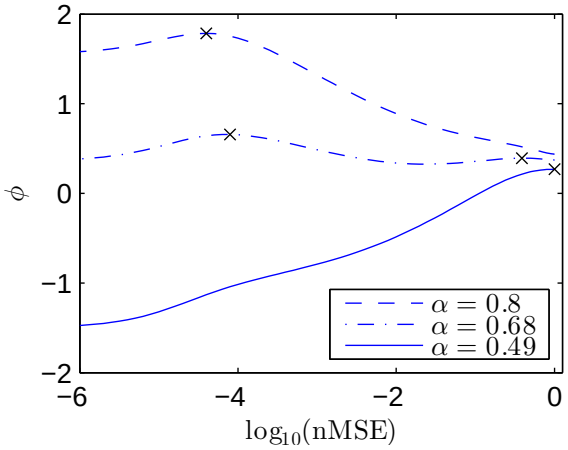


FIG. 3. Free entropy landscapes for $\rho = 0.5$, $\Delta = 10^{-5}$. Crosses represent local maxima. There are three types of them: either at nMSE= 1 (as for $\alpha = 0.49$), or at nMSE $\approx \Delta$, or in an intermediary region. In case there are several local maxima (as for $\alpha = 0.68$), the algorithm will perform sub-optimally, getting stuck in the local maximum of highest nMSE instead of converging to the global maximum ("hard but possible" phase).

Figure 3 shows the free entropy landscapes for $\rho = 0.1$ and different values of α. Instead of using m as x-axes, we use the normalized mean squared error

$$\mathrm{nMSE} = 1 - \frac{m}{\rho}, \tag{88}$$

that is a more natural quantity to measure the quality of reconstruction.

We can define three different phases depending on the positions of the free entropy maxima. In the noiseless setting, these are:

1. An "impossible" phase, in which the global maximum of the free entropy is not at nMSE= 0. In that phase, no algorithm can find the correct solution.

2. A "hard but possible" phase, in which the free entropy has its global maximum at nMSE= 0, but also a *local* maximum at non-zero nMSE. In that phase, it is possible to find the correct solution, by correctly sampling from the posterior distribution (21). However, algorithms such as P-BiG-AMP get stuck in the local free entropy maximum instead of finding the global maximum.

3. An "easy" phase, in which the free entropy function has a single maximum at nMSE= 0.

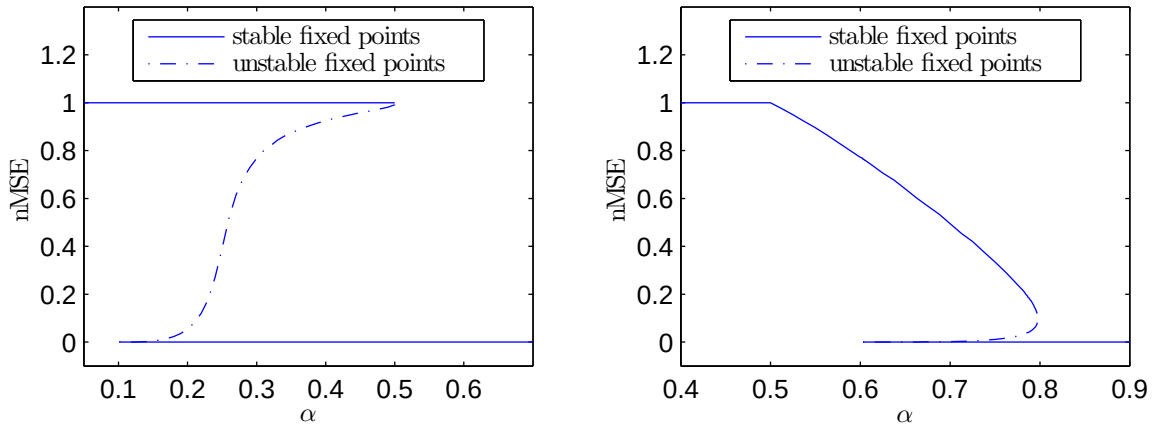
In a noisy setting as in figure 3, the lowest achievable nMSE is of the order of the AWGN variance Δ instead of 0.

1. State evolution fixed points

The state evolution equation (84) can either be iterated or considered as a fixed point equation. Figure 4 shows the fixed points of (84), which are all local extrema of the free entropy ϕ. The iterated state evolution equation converges to one of the local maxima. Since the state evolution for the matrix compressed sensing problem and the dictionary learning problem are the same (provided $L = MP$ and $R = O(M)$) these diagrams and their analysis are equivalent to those presented in previous work on the dictionary learning [14]. Notably [16] presented analogous diagrams depicting the fixed points for the dictionary learning problem.

The plots allow to see more clearly the "impossible", "hard but possible" and "easy" phases. In the "hard but possible" phase, the state evolution has an unstable fixed point, which corresponds to a local minimum of the free entropy. Three interesting facts can be noticed:

1. In the noiseless setting, the impossible/possible phase transition (the apparition of the low nMSE fixed point) takes place at $\alpha = \rho$. This can be expected because it is the



(a) $\rho = 0.1$ (b) $\rho = 0.6$

FIG. 4. Fixed points of the state evolution equation (84) for two different sparsities ρ. For values of α for which two stable fixed points exist, the iterated state evolution equation converges to the one of higher nMSE if the initial nMSE is higher than the unstable fixed point, and to the one of lower nMSE if not.

critical α at which the number of available equations is equal to the total number of non-zero components of the unknowns, just as in compressed sensing.

2. The fixed point at nMSE=1 always exists and is stable for $\alpha \in [0, 1/2]$. This is a rather remarkable fact that does not appear in compressed sensing. A consequence of this is the existence of a "hard but possible" phase that even for very small values of ρ extends at least up to $\alpha = 1/2$. This radically differs from the low-ρ regime in compressed sensing, in which the measurement rate α necessary for tractable recovery goes to zero as $\rho \to 0$.

3. Increasing α starting below $1/2$ and following the high-nMSE branch, two successive phase transitions are encountered. First, the nMSE $= 1$ fixed point disappears at $\alpha = 1/2$ and turns into an nMSE < 1 fixed point in a second order (i.e. continuous) phase transition. Second, the upper branch disappears and the discontinuity of the nMSE of the fixed point, jumping down to the lower branch, marks a first order phase transition. While these two transitions of different types are clearly visible in Figure 4b, they are too close together in Figure 4a to be distinguished. They are separated nonetheless, the easy/hard (first order) phase transition always takes place

at $\alpha > 1/2$.

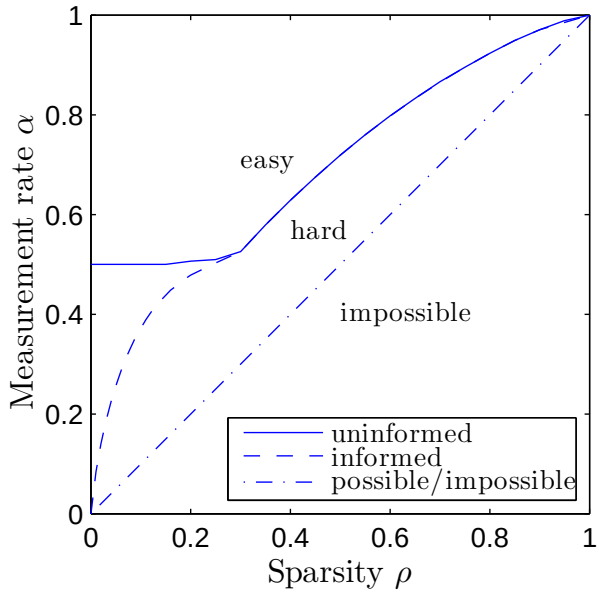


FIG. 5. Phase diagram for the considered case-study obtained from the state evolution, eq. (84). Noise variance is $\Delta = 10^{-12}$ and success is defined by a final nMSE$< 10^{-10}$. The disappearing of the state evolution fixed point (or equivalently, of a free entropy maximum) with nMSE of order 1 marks the frontier between the "hard" and the "easy" phase (full line). The dashed line marks the easy/hard phase boundary when an "informed" initialization is provided (see text). The possible/impossible frontier represented corresponds to the noiseless case.

Figure 5 shows the full phase diagram for the case-study problem, with the easy, hard and impossible phases. The "uninformed" line is obtained by starting the state evolution starting from nMSE$= 1 - \epsilon$, with an infinitesimally small ϵ, and defines the transition between the "easy" and the "hard" phase. Interestingly, the entire region with $\alpha < 0.5$ is in the hard phase, even at low values of ρ, due to the existence of the stable fixed point at nMSE $= 1$. In the "hard" phase, inference is possible provided a good estimation of the signal is already known. The effect of such a partial knowledge can be simulated by running the state evolution equation (84) starting with nMSE$= 0.9$, leading to the "informed" line, for which $\alpha \to 0$ when $\rho \to 0$. The position of this line depends strongly on the starting nMSE.

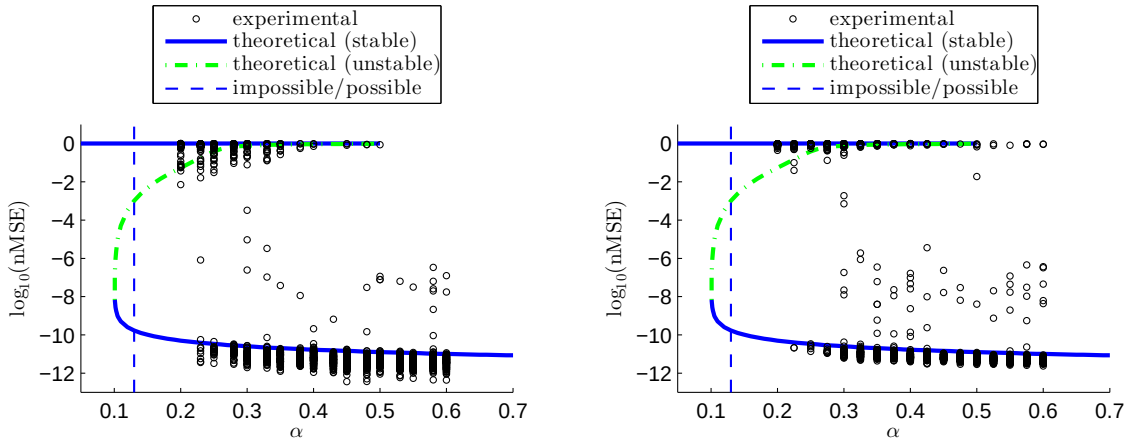
B. Comparison with algorithmic performances

Figures 6 and 7 presents a comparison of the theoretical fixed point analysis performed above with the actual performances of P-BiG-AMP.

For the experiments, rank $R = 1$ was used. In this setting, the only invariance left is a scaling invariance: if $(\mathbf{U}, \mathbf{V})$ is the true solution, then for every $\gamma \neq 0$, $(\gamma\mathbf{U}, \frac{1}{\gamma}\mathbf{V})$ is a solution as well. The final nMSE returned by the algorithm takes this invariance into account and is the average of the error on $\mathbf{U}$ and the error on $\mathbf{V}$:

$$\text{nMSE} = \frac{1}{2}\left(\text{nMSE}_\text{u} + \text{nMSE}_\text{v}\right) \tag{89}$$

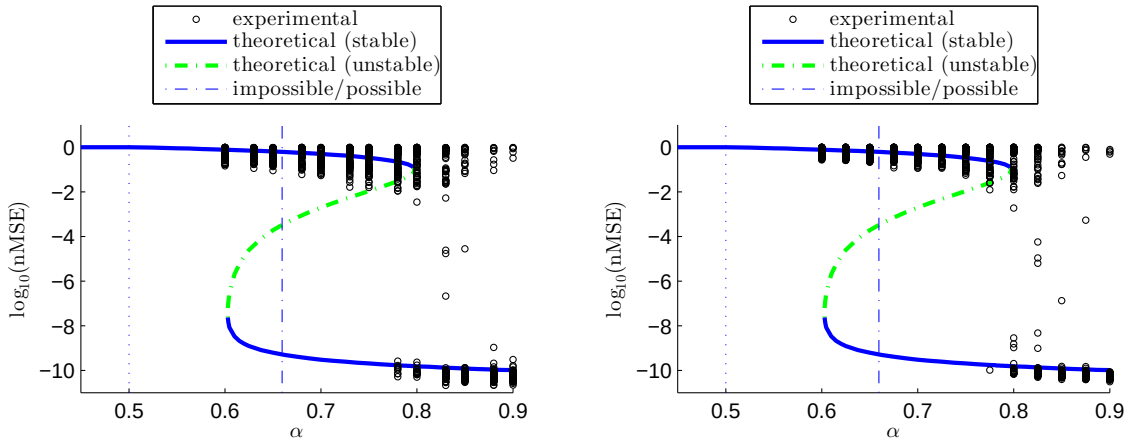
which will be compared to the results obtained by the theoretical expression (88). For each instance of the problem, the algorithm was allowed up to 20 restarts from different random initializations to reach a nMSE smaller than 10^{-6}, and the lowest of the reached nMSE was kept.



(a) $\rho = 0.1$, $M = 50$ (b) $\rho = 0.1$, $M = 200$

FIG. 6. Comparison of fixed points obtained by the state evolution and and values reached by the P-BiG-AMP algorithm. Parameters are $\rho = 0.1$, $\Delta = 10^{-12}$ with (a): $M = 50$, (b): $M = 200$. For each α there are 100 experimental points. The experimental fixed points are relatively close to the fixed points of the state evolution. Note that the spreading around the theoretical line diminishes with growing M. In the thermodynamic limit $M \to \infty$, all experimental points would be on the fixed point of *highest* nMSE. At finite M, the probability to initialize the algorithm *below* the unstable fixed point allows some instances to converge to the low-nMSE fixed point.

The results show that there is a good agreement between the theory and the performance of P-BiG-AMP: most of the nMSEs reached by P-BiG-AMP correspond to a stable fixed point of the state evolution. The agreement with the theory becomes better with increasing system size. For smaller sizes, the experimental points are more spread around the theoretical fixed points. This can be well understood by analyzing the case of fixed points with nMSE=1. The "meaning" of such fixed points is that the algorithm is unable to estimate the true signals better than at random. In the $M \to \infty$ limit, the nMSE between the true signals and random signals is 1 with probability 1. For finite values of M however, the nMSE between true and random signals follows a distribution on $[0, 1]$ that gets more peaked on 1 as M increases. This explains the narrowing of the spread of experimental points around the fixed points as M increases.



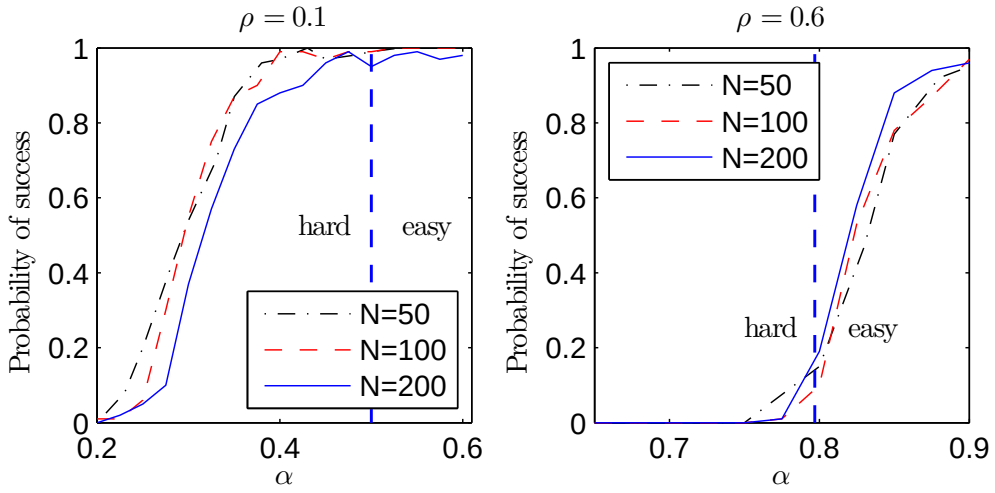
(a) $\rho = 0.6$, $M = 50$ (b) $\rho = 0.6$, $M = 200$

FIG. 7. Comparison of fixed points obtained by the state evolution and values reached by the P-BiG-AMP algorithm. Parameters are $\rho = 0.6$, $\Delta = 3.6 \times 10^{-11}$ with (a): $M = 50$, (b): $M = 200$. For each α there are 100 experimental points. Unlike for the $\rho = 0.1$ case on figure 6,the algorithm fails for an important fraction of instances in the "easy" phase. This phenomenon is not explained by the state evolution analysis and might be a finite size effect. However, as α grows the probability of success goes to 1 (see figure 8b). Unlike for $\rho = 0.1$, the probability of recovery inside the "hard" phase is much smaller, due to the lower nMSE of the unstable fixed point. The thin dotted line marks the position of the second order phase transition, at which the nMSE stops being strictly equal to 1.

1. Succeeding in the hard phase: importance of the initialization

An interesting consequence of this finite size effect is that for small M, parts of the "hard" phase are quite easy. The reason is that if the random initialization of the algorithm is such that the nMSE is *smaller* than the nMSE of the *unstable* fixed point, the algorithm naturally converges to the low-nMSE solution. Therefore, running the algorithm from a few different initializations can allow to converge to the correct solution even in the "hard" phase, provided that M is small enough and that the unstable fixed point has a high enough nMSE.

Figure 8 shows that this effect is quite important for $\rho = 0.1$, but nearly inexistent for $\rho = 0.6$. The reason for this is the much higher nMSE of the unstable fixed point for $\rho = 0.1$ than for $\rho = 0.6$.



(a) $\rho = 0.1$ (b) $\rho = 0.6$

FIG. 8. Empirical probability of success (defined by nMSE $< 10^{-6}$), for the experiments presented on figures 6 and 7. Due to the finite size, the position of the curves slightly vary for different values of M. Finite size effects allow a fraction of successful instances inside the hard phase for $\rho = 0.1$, but much less for $\rho = 0.6$.

Remember that in P-BiG-AMP, the initial estimates of $\mathbf{U}$ and $\mathbf{V}$ are random. While in some regions of the phase diagram and with small signal sizes, running the algorithm from several of those random initial estimates might be sufficient, in general it would be preferable to have a procedure that systematically produces good initializations. Previous works stress

this fact as well and often rely on an initialization from spectral methods [9, 31–33]. In addition to restarts from random initializations, P-BiG-AMP uses a damping scheme that is non-trivial to analyze. For this reason, we could not check if the results presented on figure 8 are quantitatively in agreement with the hypothesis that the finite effect described above is the only reason for success in the hard phase.

As a matter of fact, other finite size effects seem to exist as well: another difference between figures 8a and 8b is that in the latter, the algorithm fails for a significant fraction of instances inside the "easy" phase, which is not the case in the former. The fact that the fraction of such failed instances decreases with increasing signal size M seems to indicate that this is as well a finite size effect. Unlike the previously examined finite size effect, this one cannot be explained from the state evolution, as it has a unique fixed point in the "easy" phase.

V. CONCLUSION

In this paper, we provide an asymptotic analysis of Bayesian low-rank matrix compressed sensing. We employ the replica method of statistical physics to obtain the so-called state evolution equations, whose fixed points allow us to determine if inference is easy, hard or impossible. The state evolution equations describe the behavior of the associated message passing algorithm P-BiG-AMP that was derived and studied previously in [6]. This work inscribes in a line of work where approximate message passing was derived and analyzed on related estimation problems such as compressed sensing [19, 20], or matrix factorization [14, 24, 34].

An interesting point concerning the saddle point equations and the resulting state evolution equations and phase diagrams is that they are the same as those for the matrix factorization problem derived in [14]. Related observations were made in [10].

Our analysis, just as the algorithm, is written for a generic separable prior and output channel. We analyze in detail the phase diagram for Gaussian noise on the output and Gauss-Bernoulli prior on both the factors. A striking point in the phase diagram is that the α (eq. (85)) needed for the recovery to be tractable does not go to zero as the factors become very sparse. This is a remarkable difference between the matrix and the linear compressed sensing. We show numerically that there is an excellent agreement between the

theoretical analysis and the performances of the P-BiG-AMP algorithm. We observe that for the simulated system sizes, the algorithm performs better than what could be expected from the asymptotic theoretical analysis. However, we explain this as a finite size effect in terms of state evolution fixed points and stress the importance of a good initial estimate in order to perform inference outside of the easy phase. Our analysis quantifies how "good" the initialization needs to be for large systems to allow tractable recovery.

The results obtained in compressed sensing using so-called *spatial coupling* matrices have shown that for certain types of carefully designed measurement matrices, perfect signal recovery is possible in the hard but possible phase despite uninformed initialization [13]. We expect that this is the case as well for matrix compressed sensing. Verifying this would be an interesting direction for further research, especially in order to overcome the large hard phase for low ρ. Another interesting type of measurement matrices are *structured* measurement matrices, such as Fourier or Hadamard matrices. Although these matrices are not random, it has been shown that they lead to very similar results while allowing a considerable speedup of the algorithm [35, 36], while they stay analyzable by the replica method as in [37, 38].

ACKNOWLEDGEMENT

Philip Schniter's work on this project was supported in part by the National Science Foundation under grant CCF-1527162. Christophe Schülke's work was supported in part by Université franco-italienne and in part by the ERC under the European Union's 7th Framework Programme Grant Agreement 307087-SPARCS.

Appendix A: Details for the derivation of the message-passing algorithm

Here, we complete the derivation of the message-passing algorithm starting with equation (30):

$$\tilde{m}_{l\to\mu s}(u_{\mu s}) \propto f_0^Y\left(\hat{Z}_{l\to\mu s}+F_{l\mu s}u_{\mu s}, \bar{Z}_{l\to\mu s}+H_{l\mu s}u_{\mu s}+G_{l\mu s}u_{\mu s}^2\right). \tag{A1}$$

We first make a Taylor expansion of this message at order 2 around $u_{\mu s} = 0$. We drop all indices for this calculation and use simplified notations $f = f_0^Y(\hat{Z}, \bar{Z})$, $\partial_1 = \frac{\partial}{\partial \hat{Z}}$, $\partial_2 = \frac{\partial}{\partial \bar{Z}}$:

$$\begin{aligned}\tilde{m}(u) \propto\ & f + u\left(F\partial_1 f + H\partial_2 f\right) \\ & + \frac{1}{2}u^2\left(F^2\partial_1^2 f + H^2\partial_2^2 f + 2FH\partial_1\partial_2 f + 2G\partial_2 f\right) + o(u^2).\end{aligned} \tag{A2}$$

We can rewrite $\tilde{m}$ as a Gaussian

$$\tilde{m}(u) \propto \mathcal{N}(u; \hat{p}, \bar{p}) + o(u^2) \tag{A3}$$

by identifying the coefficients of the Taylor expansion above with the Taylor expansion of a Gaussian

$$\mathcal{N}(x; \frac{a}{b}, -\frac{1}{b}) \propto 1 - ax + \frac{b + a^2}{2}x^2 + o(x^2). \tag{A4}$$

Note that the form (A3) is only valid around $u = 0$: $\tilde{m}$ is not Gaussian. However this form makes calculations easier. Identification of the coefficients in (A2) and (A4) leads to

$$\begin{aligned}\bar{p} = -\Bigg[& F^2\left(\frac{\partial_1^2 f}{f} - \left(\frac{\partial_1 f}{f}\right)^2\right) + 2G\frac{\partial_2 f}{f} \\ & + H^2\left(\frac{\partial_2^2 f}{f} - \left(\frac{\partial_2 f}{f}\right)^2\right) + 2FH\left(\frac{\partial_1\partial_2 f}{f} - \frac{\partial_1 f}{f}\frac{\partial_2 f}{f}\right)\Bigg]^{-1}\end{aligned} \tag{A5}$$

$$\hat{p} = -\bar{p}\left(F\frac{\partial_1 f}{f} + H\frac{\partial_2 f}{f}\right) \tag{A6}$$

We can now treat the m-messages from eq. (23). The product is easy to handle as it is a product of Gaussians

$$\prod_{l' \neq l} \tilde{m}_{l' \to \mu s}(u_{\mu s}) \propto \prod_{l' \neq l} \mathcal{N}(u_{\mu s}; \hat{p}_{l' \to \mu s}, \bar{p}_{l' \to \mu s}) \propto \mathcal{N}(u_{\mu s}; \hat{U}_{\mu s \to l}, \bar{U}_{\mu s \to l}), \tag{A7}$$

which allows us to write

$$\bar{U}_{\mu s \to l} = \left(\sum_{l' \neq l} \bar{p}_{l' \to \mu s}^{-1}\right)^{-1} \tag{A8}$$

$$\hat{U}_{\mu s \to l} = \bar{U}_{\mu s \to l} \sum_{l' \neq l}\left(\frac{\hat{p}_{l' \to \mu s}}{\bar{p}_{l' \to \mu s}}\right) \tag{A9}$$

In the sums above, some of the non-leading order terms stemming from (A5,A6) have a vanishing contribution in the limit where $(M, P, L) \to \infty$ and will therefore be neglected. The

table below analyzes the orders of magnitude and possible signs of all quantities in (A5,A6). In the third and fourth line, we use this to analyze the order of magnitude of a sum of L of those terms, as appears in (A8,A9) and what this leads to when $L \propto RM$, which is the scaling we are interested in.

	F	G	H	F^2	H^2	FH
scales as:	$\frac{1}{\sqrt{RP}}$	$\frac{1}{RP}$	$\frac{1}{R\sqrt{MP}}$	$\frac{1}{RP}$	$\frac{1}{R^2MP}$	$\frac{1}{R^{3/2}P}$
sign:	$\pm$	$+$	$\pm$	$+$	$+$	$\pm$
sum over L	$\frac{\sqrt{L}}{\sqrt{RP}}$	$\frac{L}{RP}$	$\frac{\sqrt{L}}{R\sqrt{MP}}$	$\frac{L}{RP}$	$\frac{L}{R^2MP}$	$\frac{\sqrt{L}}{R^{3/2}P}$
$L \propto RP$	1	1	$\frac{1}{\sqrt{RM}}$	1	$\frac{1}{RM}$	$\frac{1}{R\sqrt{P}}$

This analysis is based on the fact that:

- $\mathbf{A}$ has random i.i.d. elements of mean 0 and variance $1/(RMP)$
- $\mathbf{U}$, $\mathbf{V}$ and $\mathbf{z}$ have zero-mean elements of order 1, therefore all estimators of type $\hat{\mathbf{u}}, \hat{\mathbf{U}}$, etc. are of order 1 as well, either positive or negative
- variances of type $\bar{\mathbf{u}}, \bar{\mathbf{U}}$, etc. are positive and of order 1
- all quantities of the type $\frac{\partial_i f}{f}$ are of order 1.

With the help of the table, we can neglect all terms that have a vanishing contribution. Furthermore,using the relations (17,18) and the definition of the g functions (20), it can be shown that

$$\frac{\partial_1 f^Y}{f^Y} = \hat{g}^Y, \tag{A10}$$

$$\frac{\partial_1^2 f^Y}{f^Y} - \left(\frac{\partial_1 f^Y}{f^Y}\right)^2 = \bar{g}^Y, \tag{A11}$$

$$\frac{\partial_2 f^Y}{f^Y} = \frac{1}{2}\left(\bar{g}^Y - (\hat{g}^Y)^2\right). \tag{A12}$$

In the end, the resulting expressions for (A8,A9) are given in (32,33).

Appendix B: Details for the replica calculation

a. Covariance matrix of $\mathbf{z}_l$ We treat $z_l^a = [\mathcal{A}(\mathbf{U}^a(\mathbf{V}^a)^\top)]_l$ as a random variable of $\mathcal{A}$ and look at the covariance between two of those variables:

$$\langle z_l^a z_{l'}^b \rangle = \left\langle \left(\sum_{\mu p} A_l^{\mu p} \sum_s u_{\mu s}^a v_{ps}^a \right) \left(\sum_{\mu' p'} A_{l'}^{\mu' p'} \sum_{s'} u_{\mu' s'}^b v_{p' s'}^b \right) \right\rangle \tag{B1}$$

$$= \left\langle \sum_{\mu\mu'} \sum_{pp'} A_l^{\mu p} A_{l'}^{\mu' p'} \sum_{ss'} u_{\mu s}^a u_{\mu' s'}^b v_{ps}^a v_{p' s'}^b \right\rangle \tag{B2}$$

$$= \sum_{\mu\mu'} \sum_{pp'} \langle A_l^{\mu p} A_{l'}^{\mu' p'} \rangle \sum_{ss'} u_{\mu s}^a u_{\mu' s'}^b v_{ps}^a v_{p' s'}^b \tag{B3}$$

As the elements of $\mathbf{A}$ are i.i.d. with zero mean and variance $1/(RMP)$, we have $\langle A_l^{\mu p} A_{l'}^{\mu' p'} \rangle = \delta_{l,l'} \delta_{\mu,\mu'} \delta_{p,p'} \frac{1}{RMP}$ and thus

$$\langle z_l^a z_{l'}^b \rangle = \delta_{l,l'} \frac{1}{RMP} \sum_{ss'} \left(\left(\sum_\mu u_{\mu s}^a u_{\mu s'}^b \right) \left(\sum_p v_{ps}^a v_{ps'}^b \right) \right) \tag{B4}$$

$$= \frac{\delta_{l,l'}}{R} \sum_{ss'} \left(\left(\frac{1}{M} \sum_\mu u_{\mu s}^a u_{\mu s'}^b \right) \left(\frac{1}{P} \sum_p v_{ps}^a v_{ps'}^b \right) \right) \tag{B5}$$

We now make the following assumption:

$$\frac{1}{M} \sum_\mu u_{\mu s}^a u_{\mu s'}^b = \begin{cases} Q_u^{ab} = O(1) & \text{if } s = s' \\ (Q_u^{ab})_{ss'} = O(\frac{1}{\sqrt{M}}) & \text{if } s \neq s' \end{cases} \tag{B6}$$

This assumption corresponds to breaking the column-permutation symmetry and more generally the rotational symmetry between different replicas. We thus assume that the s-th column of $\mathbf{U}^a$ is correlated to the s-th column of $\mathbf{U}^b$ and to none of the others. We make the same assumption for $\mathbf{V}$. Then,

$$\langle z_l^a z_{l'}^b \rangle = \frac{\delta_{l,l'}}{R} \left(\sum_s Q_u^{ab} Q_v^{ab} + \sum_{s \neq s'} (Q_u^{ab})_{ss'} (Q_v^{ab})_{ss'} \right). \tag{B7}$$

Due to the hypothesis (B6), the second term vanishes, and

$$\langle z_l^a z_{l'}^b \rangle = \delta_{s,s'} Q_u^{ab} Q_v^{ab}. \tag{B8}$$

Note that by definition of Q_u^{ab} in (B6), $Q_u^{ab} = Q_u^{ba}$.

b. Introducing $\hat{\mathbf{Q}}_u$ In equation (43), Dirac δ functions enforce the relations (B6). We use the integral representation of these δ functions to carry on the calculation:

$$\delta\left(MQ_u^{ab} - \sum_\mu u_{\mu s}^a u_{\mu s}^b\right) = \frac{1}{2\pi\imath}\int \mathrm{d}\tilde{Q}_U^{ab} e^{-\tilde{Q}_U^{ab}\left(MQ_u^{ab} - \sum_\mu u_{\mu s}^a u_{\mu s}^b\right)}. \tag{B9}$$

The product of all these δ functions thus gives

$$\prod_{a\leq b}\delta\left(MQ_u^{ab} - \sum_\mu u_{\mu s}^a u_{\mu s}^b\right) \propto \int \mathrm{d}\tilde{\mathbf{Q}}_U e^{-M\sum_{a\leq b}\tilde{Q}_U^{ab}Q_u^{ab}} e^{\sum_\mu \sum_{a\leq b}\tilde{Q}_U^{ab}u_{\mu s}^a u_{\mu s}^b}. \tag{B10}$$

Note that because $Q_u^{ab} = Q_u^{ba}$, the replica indices in the sum are $a \leq b$. Finally, we make a change of variables

$$\forall a, \quad \hat{Q}_U^{aa} = 2\tilde{Q}_U^{aa} \tag{B11}$$

$$\forall (a,b) \text{ with } a \neq b, \quad \hat{Q}_U^{ab} = 4\tilde{Q}_U^{ab} \tag{B12}$$

which allows us to obtain the following formulas

$$\sum_{a\leq b}\tilde{Q}_U^{ab}Q_u^{ab} = \frac{1}{2}\mathrm{Tr}(\mathbf{Q}_u\hat{\mathbf{Q}}_u), \tag{B13}$$

$$\sum_{a\leq b}\tilde{Q}_U^{ab}u_{\mu s}^a u_{\mu s}^b = \frac{1}{2}\mathbf{u}_{\mu s}^\top\hat{\mathbf{Q}}_u\mathbf{u}_{\mu s}, \tag{B14}$$

where we introduced the vector $\mathbf{u}_{\mu s} = (u_{\mu s}^0 \dots u_{\mu s}^n)^\top$. We change the integration variable from $\tilde{\mathbf{Q}}_U$ to $\hat{\mathbf{Q}}_u$, and we obtain the expression (44).

---

[1] D. L. Donoho, IEEE Transactions on Information Theory **52**, 1289 (2006).

[2] E. J. Candès and B. Recht, Foundations of Computational mathematics **9**, 717 (2009).

[3] D. Gross, Y.-K. Liu, S. T. Flammia, S. Becker, and J. Eisert, Phy. Rev. Lett. **105**, 1 (2010).

[4] E. J. Candès, X. Li, Y. Ma, and J. Wright, J. ACM **58**, 11 (2011).

[5] E. J. Candès and Y. Plan, IEEE Trans. Info. Thy. **57**, 2342 (2011).

[6] J. T. Parker and P. Schniter, IEEE Journal of Selected Topics in Signal Processing **10**, 795 (2016).

[7] R. Basri and D. W. Jacobs, IEEE Trans. Pattern Anal. Mach. Intell. **25**, 218 (2003).

[8] J. Wright, A. Yang, A. Ganesh, S. Sastry, and Y. Ma, IEEE Trans. Pattern Anal. Mach. Intell. **31**, 210 (2009).

[9] B. Recht, M. Fazel, and P. A. Parrilo, SIAM Review **52**, 471 (2010), http://dx.doi.org/10.1137/070697835.

[10] D. L. Donoho, M. Gavish, and A. Montanari, Proceedings of the National Academy of Sciences **110**, 8405 (2013), http://www.pnas.org/content/110/21/8405.full.pdf.

[11] M. Mézard, G. Parisi, and M.-A. Virasoro, *Spin glass theory and beyond.* (World Scientific Publishing Co., Inc., Pergamon Press, 1990).

[12] H. Nishimori, *Statistical physics of spin glasses and information processing: an introduction*, 111 (Oxford University Press, 2001).

[13] F. Krzakala, M. Mézard, F. Sausset, Y. F. Sun, and L. Zdeborová, Phys. Rev. X **2**, 021005 (2012).

[14] Y. Kabashima, F. Krzakala, M. Mézard, A. Sakata, and L. Zdeborová, IEEE Transactions on Information Theory **62**, 4228 (2016).

[15] F. Krzakala, M. Mézard, and L. Zdeborová, in *Information Theory Proceedings (ISIT), 2013 IEEE International Symposium on* (IEEE, 2013) pp. 659–663.

[16] A. Sakata and Y. Kabashima, in *Information Theory Proceedings (ISIT), 2013 IEEE International Symposium on* (2013) pp. 669–673.

[17] E. Riegler, D. Stotz, and H. Bolcskei, in *Information Theory (ISIT), 2015 IEEE International Symposium on* (2015) pp. 1836–1840.

[18] Y. Kabashima, Journal of Physics A: Mathematical and General **36**, 11111 (2003).

[19] D. L. Donoho, A. Maleki, and A. Montanari, Proceedings of the National Academy of Sciences **106**, 18914 (2009).

[20] S. Rangan, in *Information Theory Proceedings (ISIT), 2011 IEEE International Symposium on* (2011) pp. 2168–2172.

[21] D. B. R. A. P. Dempster, N. M. Laird, Journal of the Royal Statistical Society. Series B (Methodological) **39**, 1 (1977).

[22] M. Mezard and A. Montanari, *Information, physics, and computation* (Oxford University Press, 2009).

[23] D. J. Thouless, P. W. Anderson, and R. G. Palmer, Philosophical Magazine **35**, 593 (1977), http://dx.doi.org/10.1080/14786437708235992.

[24] J. Parker, P. Schniter, and V. Cevher, Signal Processing, IEEE Transactions on **62**, 5839 (2014).

[25] J. Vila, P. Schniter, S. Rangan, F. Krzakala, and L. Zdeborová, in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on* (IEEE, 2015) pp. 2021–2025.

[26] Y. Kabashima, in *Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks and Workshops, 2008. WiOPT 2008. 6th International Symposium on* (2008) pp. 620–624.

[27] R. Matsushita and T. Tanaka, in *Advances in Neural Information Processing Systems 26*, edited by C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger (Curran Associates, Inc., 2013) pp. 917–925.

[28] T. Lesieur, F. Krzakala, and L. Zdeborová, in *Information Theory (ISIT), 2015 IEEE International Symposium on* (2015) pp. 1635–1639.

[29] L. Zdeborová and F. Krzakala, arXiv preprint arXiv:1511.02476 (2015).

[30] F. Caltagirone, L. Zdeborová, and F. Krzakala, in *Information Theory (ISIT), 2014 IEEE International Symposium on* (2014) pp. 1812–1816.

[31] K. Lee, Y. Wu, and Y. Bresler, arXiv preprint arXiv:1312.0525 (2013).

[32] P. Jain, P. Netrapalli, and S. Sanghavi, in *Proceedings of the Forty-fifth Annual ACM Symposium on Theory of Computing*, STOC '13 (ACM, New York, NY, USA, 2013) pp. 665–674.

[33] Q. Zheng and J. Lafferty, in *Advances in Neural Information Processing Systems 28*, edited by C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett (Curran Associates, Inc., 2015) pp. 109–117.

[34] J. Parker, P. Schniter, and V. Cevher, Signal Processing, IEEE Transactions on **62**, 5854 (2014).

[35] A. Javanmard and A. Montanari, in *Information Theory Proceedings (ISIT), 2012 IEEE International Symposium on* (2012) pp. 2431–2435.

[36] J. Barbier, C. Schülke, and F. Krzakala, Journal of Statistical Mechanics: Theory and Experiment **2015**, P05013 (2015).

[37] Y. Kabashima, M. Vehkaperä, and S. Chatterjee, Journal of Statistical Mechanics: Theory and Experiment **2012**, P12003 (2012).

[38] M. Vehkaperä, Y. Kabashima, and S. Chatterjee, IEEE Transactions on Information Theory **62**, 2100 (2016).