How to Design Message Passing Algorithms for Compressed Sensing

David L. Donoho[*], Arian Maleki[†] and Andrea Montanari[*,†]

February 17, 2011

Abstract

Finding fast first order methods for recovering signals from compressed measurements is a problem of interest in applications ranging from biology to imaging. Recently, the authors proposed a class of low-complexity algorithms called approximate message passing or AMP. The new algorithms were shown, through extensive simulations and mathematical analysis, to exhibit very fast convergence rate and high phase transition. This paper provides a step-by-step explanation of how these algorithms can be constructed, leveraging on the ideas from the graphical models theory, and message passing.

1 Introduction

There has been much recent effort on finding new algorithms for recovering sparse solutions of underdetermined system of linear equations. The ℓ_1 minimization, also known as the basis pursuit [7], has attracted attention as an efficient recovery algorithm. It solves the following optimization problem

$$\left\{ \begin{array}{ll} \text{minimize} & \|s\|_1 \, , \\ \text{subject to} & y = As \, , \end{array} \right. \tag{1}$$

where $s \in \mathbb{R}^N$ is the vector to be recovered, A is an $n \times N$ measurement matrix and $y \in \mathbb{R}^n$ includes the measurements. The solution of this problem can be obtained by generic linear programming or LP algorithms. However, high computational complexity of LP presents an obstacle for large problem sizes that occur very often in applications. Low computational complexity of iterative thresholding algorithms has made them an appealing alternative. Therefore many variations of this approach have been proposed in the literature [32, 19, 34, 10, 8, 17, 21, 16, 18, 4, 20, 27, 37, 36, 5, 6, 31, 2, 30, 23]. A generic form of iterative thresholding algorithms is,

$$\begin{aligned} x^{t+1} &= \eta_t(x^t + A^* z^t) \\ z^t &= y - Ax^t, \end{aligned} \tag{2}$$

where $x^t \in \mathbb{R}^N$ is the current estimate. $z^t \in \mathbb{R}^n$ is the residual. A^* is the Hermitian of the matrix and finally η_t is a nonlinear function that imposes the sparsity or the structure of the signal on the estimation. Popular examples of such thresholding functions are soft thresholding where $\eta(x; b) = \text{sign}(x)(x - b)_+$ and hard thresholding where $\eta(x; b) = x\mathbf{I}(|x| \geq b)$. The threshold level b in these two cases may depend on the iteration t and there are different heuristics for setting this value. Whenever the arguments of these

[*] Department of Statistics, Stanford University

[†] Department of Electrical Engineering, Stanford University

functions are vectors, it means that the function is applied component-wise to the elements of the vector. Unfortunately the iterative thresholding algorithms suffer either from slow convergence rate or low phase transition compared to ℓ_1.

In a recent paper [13], we proposed an algorithm that exhibits both the low computational complexity of iterative thresholding algorithms and the reconstruction power of the basis pursuit. The algorithm was dubbed AMP, for 'approximate message passing', and was inspired by the ideas from graphical models theory, message passing algorithms, and statistical physics. Starting from $x^0 = 0$, the algorithm proceeds according to the following iteration,

$$\begin{aligned} x^{t+1} &= \eta(x^t + A^* z^t; \beta\hat{\sigma}^t), \\ z^t &= y - Ax^t + \langle \eta'(x^{t-1} + A^* z^{t-1}; \beta\hat{\sigma}^{t-1}) \rangle. \end{aligned} \tag{3}$$

η is the soft thresholding function and z^t and x^t are similar to the corresponding notation in the generic iterative thresholding. We have also chosen the special value $\beta\hat{\sigma}^t$ for the threshold levels. $\hat{\sigma}^t$ is an estimate of the standard deviation of the 'noise' ($x^t + A^* z^t - x_o$) at time t and β is a constant parameter that shall be tuned optimally [13]. This noise is called multiple access interference noise. The only discrepancy from the generic iterative thresholding algorithms is the extra term $\langle \eta'(A^* z^{t-1} + x_i^{t-1}; \hat{\tau}^{t-1}) \rangle$ in the calculation of the residual. This term has a significant effect on the performance of the algorithm [13] while slightly increases the computational complexity.

In this paper, it will be shown that this algorithm is equivalent to the sum product belief propagation algorithm if a suitable joint distribution is considered for the variables $s_1, s_2, \dots, s_N$. Remarkably, AMP update rules are much simpler than the justifying sum-product rules. Such a striking simplification emerge in the large system limit. This is one instance of the "blessings of dimensionality" [11]. Apart from justifying the algorithms studied in [13], the unified derivation provided here allows to develop (1) an automatic way to set the threshold without any need for optimal tuning of parameter β and (2) other versions of the algorithm that are better suited for specific objectives. In particular, an important alternative of the problem (1) is

$$\text{minimize} \quad \lambda \|s\|_1 + \frac{1}{2}\|y - As\|_2^2 \,, \tag{4}$$

also known as Basis Pursuit De-Noising (BPDN), or Lasso. We derive an iterative AMP algorithm for this problem that has exactly the same structure as the AMP with different threshold values. Finally, the approach presented here allows us to systematically incorporate further information concerning the distribution of the signal s, thus bridging the gap with a Baysian approach to compressed sensing reconstruction.

2 Contribution and organization of the paper

In this section we explain the notation that will be used throughout the paper and will then explain our main contributions. Finally we compare the ideas presented in this paper with the related work in the literature.

2.1 Notation

Let s_o be a vector in $\mathbb{R}^N$. We observe $n < N$ linear measurements of this vector through the matrix A, $y = As_o$. The goal is to recover s_o from (y, A). Although the number of measurements is much smaller than the dimension of the signal, the problem can still be solved since the underlying signal is 'structured' in an appropriate sense. A specific notion of 'simplicity' amounts to postulating that s is either exactly or approximately sparse. The columns of A are assumed to have unit ℓ_2 norm. $a, b, c, \dots$ and and $i, j, k, \dots$ denote the indices in $[n] \equiv \{1, 2, \dots, n\}$ and $[N] \equiv \{1, 2, \dots, N\}$ respectively. The a, i element of the matrix A will be indicated as A_{ai}. We are following the standard notation in graphical models where $a, b, c, \dots$ represent the factor nodes and $i, j, k, \dots$ are used for the variable nodes[26]. The elements of the vectors y, s, x, s_o are indicated by $y_a, s_i, x_i, s_{o,i}$ respectively. Let $\delta = n/N$ be a measure of indeterminacy of the measurement system. Whenever we refer to the *large system limit* we consider the case where $N \to \infty$ while δ is fixed. Since in most of the applications of compressed sensing such as the magnetic resonance imaging the problem of interest has millions of variables with tens of thousands of measurements, the large system limits of the algorithms are of particular interest. In addition large system limit provides very sharp and exact sampling theorems that can then be used for comparing compressed sensing algorithms [23]. It is worth mentioning that in practice, the algorithms we develop in this paper perform well even in the medium size problems where there are just thousands of variables and hundreds of measurements [13]. In the rest of this section we explain the original problems we want to solve and the corresponding iterative thresholding algorithms.

2.2 Basis pursuit problem

Consider the following distribution over the variables $s_1, s_2, \dots, s_N$

$$\mu(\mathrm{d}s) = \frac{1}{Z} \prod_{i=1}^{N} \exp\left(-\beta |s_i|\right) \prod_{a=1}^{n} \delta_{\{y_a = (As)_a\}}\,, \tag{5}$$

where $\delta_{\{y_a=(As)_a\}}$ denotes a Dirac distribution on the hyperplane $y_a = (Ax)_a$. It is clear that as $\beta \to \infty$, the mass of this distribution concentrates around the solution of (1). If the minimizer is unique and the marginals of μ are known, the solution of (1) will be immediate. Belief propagation provides a low-complexity heuristic for approximating such marginals. In order to introduce belief propagation, we consider the factor graph $G = (V, F, E)$ with variable nodes $V = [N]$, factor nodes $F = [n]$ and edges $E = [N] \times [n] = \{(i, a) : i \in [N],\, a \in [n]\}$. Hence G is the complete bipartite graph with N variable nodes and n functional nodes. It is easy to see that the joint distribution (5) is structured according to this factor graph.

The state variables of the belief propagation are the messages $\{\nu_{i\to a}\}_{i\in V, a\in F}$, $\{\hat{\nu}_{a\to i}\}_{i\in V, a\in F}$ associated with the edges of this graph. In the present case, messages are probability measures over the real line. Throughout this paper $\nu_{i\to a}$, $\hat{\nu}_{a\to i}$ denote densities. The update rules for the densities are

$$\nu^{t+1}_{i\to a}(s_i) \;\cong\; e^{-\beta|s_i|} \prod_{b\neq a} \hat{\nu}^t_{b\to i}(s_i)\,, \tag{6}$$

$$\hat{\nu}^t_{a\to i}(s_i) \;\cong\; \int \prod_{j\neq i} \nu^t_{j\to a}(s_i)\;\, \delta_{\{y_a-(As)_a\}} \mathrm{d}s\,. \tag{7}$$

Here and below a superscript denotes the iteration number. Moreover, the symbol $\cong$ denotes identity between probability distributions up to a normalization constant[1]. Unfortunately this message passing algorithm has two problems. First, the messages are density functions over the real line and unless they have certain structure keeping track of these messages will be very difficult. Second, since the graph is dense the number of messages are $2nN$ and therefore the algorithm is computationally expensive. In section 3 we will prove that in the large system limit and as $\beta \to \infty$ this complicated message passing algorithm is equivalent to the following simple iterative algorithm.

[1] More precisely, given two non-negative functions p, $q : \Omega \to \mathbb{R}$ over the same space, we write $p(s) \cong q(s)$ if there exists a positive constant a such that $p(s) = a\, q(s)$ for every $s \in \Omega$.

Starting from $x^0 = 0$ and $\hat{\tau}^0 = 1$ the resulting iterative algorithm proceeds according to

$$\begin{aligned} x^{t+1} &= \eta(A^* z^t + x^t; \hat{\tau}^t)\,, \\ z^t &= y - Ax^t + \frac{1}{\delta} z^{t-1} \langle \eta'(A^* z^{t-1} + x_i^{t-1}; \hat{\tau}^{t-1}) \rangle\,, \\ \hat{\tau}^t &= \frac{\hat{\tau}^{t-1}}{\delta} \langle \eta'(A^* z^{t-1} + x^t; \hat{\tau}^{t-1}) \rangle\,. \end{aligned} \tag{8}$$

where $\eta(x; b) = \text{sign}(x)(|x| - b)_+$ is the soft thresholding function applied entry-wise. η' is the first derivative of η with respect to the first argument and the notation $\langle \cdot \rangle$ is the averaging operator. Intuitively speaking the x_i^t in the iterative algorithm corresponds to the mean of the message $\nu_{i \to a}^t$, z_a^t corresponds to the mean of the message $\hat{\nu}_{a \to i}^t$ and finally $\hat{\tau}^t$ corresponds to the variance of the message $\nu_{i \to a}^t$. For more careful definition and analysis of these terms refer to section 3. We will call this algorithm $AMP.0$.

2.3 BPDN problem

Now consider the following density function over the variables $s = (s_1, \dots s_N)$.

$$\mu(\mathrm{d}s) = \frac{1}{Z} \prod_{i=1}^{N} \exp(-\beta\lambda|s_i|) \prod_{a=1}^{n} \exp\Big\{ -\frac{\beta}{2}(y_a - (As)_a)^2 \Big\}\, \mathrm{d}s\,. \tag{9}$$

Notice that the mode of this distribution coincides with the solution of BPDN and the distribution concentrates on its mode as $\beta \to \infty$.

In section 4 we will show that each iteration of the sum-product message passing algorithm is equivalent to the following AMP algorithm.

$$\begin{aligned} x^t &= \eta(x^t + A^* z^t; \lambda + \gamma^t), \\ z^{t+1} &= y - Ax^t + \frac{1}{\delta} z^t \langle \eta'(x^{t-1} + A^* z^{t-1}), \rangle \\ \gamma^{t+1} &= \frac{\lambda + \gamma^t}{\delta} \langle \eta'(Az^t + x^t; \gamma^t + \lambda) \rangle\,. \end{aligned} \tag{10}$$

The only difference between this algorithm and AMP.0 is in the way the threshold parameter is set. We call this algorithm AMP.A where A stands for the automatic threshold selection.

2.4 Theoretical Prediction

Statistical properties of approximate message passing algorithms allow us to accurately analyze the asymptotical performance of the algorithm. The state evolution framework introduced in [13] will be briefly reviewed in section 5. Based on this framework we derive the following equations that predict the evolution of AMP.0 and AMP.A algorithms. Assuming that the empirical distribution of s_o converges weakly to p_s the state evolution equations are

$$\begin{aligned} \tau_{t+1}^2 &= \sigma^2 + \frac{1}{\delta} \mathbb{E}[\eta(X_0 + \tau_t Z; \lambda + \gamma^t) - X_0]^2, \\ \gamma^{t+1} &= \frac{\gamma^t + \lambda}{\delta} \mathbb{E}[\eta'(X_0 + \tau_t Z; \lambda + \gamma^t)]. \end{aligned}$$

In these two equations (τ^t, γ^t) are called the states of the system at time t. X_0 and Z are two independent random variables with density function p_s and $N(0, 1)$ respectively. σ is the standard deviation of the measurement noise. In the above equations $\lambda = 0$, corresponds to the AMP.0 algorithm.

2.5 Extensions

The method we will propose for deriving the above AMP algorithms, enables us to incorporate more complicated priors (if available on the data). To demonstrate this we consider two more complicated priors in the extension section and develop the corresponding message passing algorithms. First, we will see how one can add a positivity constraint. Second, we will consider an arbitrary product distribution on the variables and will derive a simple iterative algorithm that is equivalent to the sum product belief propagation.

2.6 Comparison with other work

2.6.1 First order methods

As mentioned in the introduction finding fast first order methods for ℓ_1 minimization is an active area of research and numerous approaches have been proposed [32, 19, 34, 10, 8, 17, 21, 16, 18, 4, 20, 27, 37, 36, 5, 6, 31, 2, 30, 23]. Here we just emphasize on the main differences between the algorithms constructed in this paper with those proposals. For more formal comparison the reader is referred to [25].

(1) The AMP algorithm is derived from the statistical point of view rather than linear algebraic or convex analysis view point. This makes the accurate analysis of the algorithm on compressed sensing problem possible. The linear algebraic analysis of the convergence rate may provide lower bounds that are far from the reality of compressed sensing problems. For instance, we are able to prove linear convergence of the estimate of AMP to the final solution, while the best result known for linear algebraic methods is strong convergence without any specific bound on the rate [10, 2, 8].

(2) As a result of the statistical analysis all the free parameters can be tuned optimally. Therefore the algorithms we propose are parameter free. Also the theoretical framework of this algorithm allows us to analyze different continuation strategies [21] which is considered as a difficult problem for other approaches.

2.6.2 Message passing algorithms

The use of message passing algorithms for compressed sensing problems was suggested before, see for instance [33]. However such a proposal faces two major difficulties.

(1) According to the standard prescription, messages used in the the sum-product algorithm should be probability measures over the real line $\mathbb{R}$, cf. Eqs. (12), (13). This is impractical from a computational point of view. That's why simpler models such as mixture models are sometimes considered in these cases.

(2) The factor graph on which the sum-product algorithm is run is the complete bipartite graph with N variable nodes, and n function nodes. In other words, unless the underlying matrix is sparse, the graphical model is very dense. This requires to update Nn messages per iteration, and each message update depend on N or n input messages. Again this is very expensive computationally.

(3) The use of belief propagation requires to define a prior on the vector s_o. For most applications, no good prior is available.

2.6.3 State evolution and replica calculations

In the context of coding theory, message passing algorithms are analyzed through density evolution [29]. The common justification for density evolution is that the underlying graph is random and sparse, and hence converges locally to a tree in the large system limit. In the case of trees density evolution is exact, hence it is asymptotically exact for sparse random graphs.

State evolution is the analog of density evolution in the case of dense graphs. For definitions and results on state evolution we refer to the [13, 12]. The success of state evolution cannot be ascribed to the locally tree-like structure of the graph, and calls for new mathematical ideas.

The fixed points of state evolution describe the output of the corresponding AMP, when the latter is run for a sufficiently large number of iterations (independent of the dimensions n, N). It is well known, within statistical mechanics [26], that the fixed point equations do indeed coincide with the equations obtained through a completely different non-rigorous approach, the *replica method* (in its replica-symmetric form). This is indeed an instance of a more general equivalence between replica and cavity methods.

During the last year, several papers investigated compressed sensing problems using the replica method [28, 22, 9]. In view of the discussion above, it is not surprising that these results can be recovered from the state evolution formalism put forward in [13]. Let us mention that the latter has several advantages over the replica method: (1) It is more concrete, and its assumptions can be checked quantitatively through simulations; (2) It is intimately related to efficient message passing algorithms; (3) It actually allows to predict the performances of these algorithms.

2.7 Organization of the Paper

In the interest of clarity, we first present our results on the basis pursuit problem (1) in Section 3. We will then consider problem (4) in Section 4. Section 5 will be devoted to the asymptotic analysis of the algorithm and finally in section 6 we will be discussing more complicated priors.

3 AMP for reconstruction under hard constraints

In this section the basis pursuit optimization problem defined in Eq. (1) is considered. In the concrete derivation, for the sake of simplicity we assume that $A_{ai} \in \{+1/\sqrt{n}, -1/\sqrt{n}\}$. This is not crucial, and only simplifies some of the calculations. The derivation of AMP proceeds in 4 steps:

1. Construct a joint distribution over $(s_1, \dots, s_N)$, parameterized by $\beta \in \mathbb{R}_+$, associated with the problem of interest. The distribution is structured according to a graphical model and it is immediate to write down the corresponding sum-product belief propagation algorithm.

2. Show, by central limit theorem argument, that in the large system limit, the sum product messages are well approximated by families with two scalar parameters. Derive the update rules for these parameters.

3. Find the limit $\beta \to \infty$ (the entire mass of the distribution will concentrate around the mode) and get the appropriate rules for minimization.

4. Approximate the message passing rules for large systems with updates of the form (8).

3.1 Construction of the graphical model

We consider the following joint probability distribution over the variables $s_1, s_2, \dots s_N$

$$\mu(\mathrm{d}s) = \frac{1}{Z} \prod_{i=1}^{N} \exp\left(-\beta |s_i|\right) \prod_{a=1}^{n} \mu_{A,y}(\mathrm{d}s)\,,$$

where $\mu_{A,y}$ is the Lebesgue measure on the hyperplane $\{s\,:\, As = y\}$, and Z is a constant that ensures the normalization $\int \mu(\mathrm{d}s) = 1$. In other words, the weights that are assigned to the solutions of the linear system $As = y$, decay exponentially with the ℓ_1 norm of the solutions. This measure can be written more explicitly as

$$\mu(\mathrm{d}s) = \frac{1}{Z}\prod_{i=1}^{N}\exp\left(-\beta|s_i|\right)\prod_{a=1}^{n}\delta_{\{y_a=(As)_a\}}\,. \tag{11}$$

Here and below $\delta_{\{y_a=(As)_a\}}$ denotes a Dirac distribution on the hyperplane $y_a = (Ax)_a$. Products of such distributions associated with distinct hyperplanes yield a well defined measure. As we let $\beta \to \infty$, the mass of the above distribution concentrates around the solution of (1). If the minimizer is unique and the marginals of μ are known, the solution of (1) will be immediate. Belief propagation provides a low-complexity heuristic for approximating marginals.

We consider the factor graph $G = (V, F, E)$ with variable nodes $V = [N]$, factor nodes $F = [n]$, and edges $E = [N] \times [n] = \{(i, a) : i \in [N],\, a \in [n]\}$. The update rules for the sum-product message passing algorithm on this graph are

$$\nu^{t+1}_{i\to a}(s_i) \;\cong\; e^{-\beta|s_i|}\prod_{b\neq a}\hat{\nu}^t_{b\to i}(s_i)\,, \tag{12}$$

$$\hat{\nu}^t_{a\to i}(s_i) \;\cong\; \int\prod_{j\neq i}\nu^t_{j\to a}(s_i)\;\delta_{\{y_a-(As)_a\}}\,, \tag{13}$$

where superscript denotes the iteration number. In the next section we will try to find the form of the messages in the large system limit.

3.2 Large system limit

The main goal in this section is to show that in the large system limit as $N \to \infty$ the messages have very simple forms. More specifically we show that under certain conditions that will be explained later for n, N large, the messages $\hat{\nu}^t_{a\to i}(\,\cdot\,)$ are approximately Gaussian distributions with variances of order N. On the other hand, the densities of messages $\nu^t_{i\to a}(\,\cdot\,)$ are well approximated by the product of a Gaussian and a Laplace density. We state this fact formally below. Recall that, given two distributions μ_1, μ_2, their Kolmogorov distance is

$$||\mu_1 - \mu_2||_{\mathrm{K}} \equiv \sup_{a\in\mathbb{R}}\Big|\int_{-\infty}^{a}\mu_1(\mathrm{d}x) - \int_{-\infty}^{a}\mu_2(\mathrm{d}x)\Big|\,. \tag{14}$$

The first Lemma provides an estimate for the messages $\hat{\nu}^t_{a\to i}$.

Lemma 3.1. *Let $x^t_{j\to a}$ and $(\tau^t_{j\to a}/\beta)$ be, respectively, the mean and variance of the distribution $\nu^t_{j\to a}$. Assume further $\int|s_j|^3\mathrm{d}\nu^t_{j\to a}(s_j) \leq C_t$ uniformly in N, n. Then there exists a constant C'_t such that*

$$\begin{aligned}
||\hat{\nu}^t_{a\to i} - \hat{\phi}^t_{a\to i}||_{\mathrm{K}} &\leq \frac{C'_t}{N^{1/2}(\hat{\tau}^t_{a\to i})^{3/2}}\,,\\
\hat{\phi}^t_{a\to i}(\mathrm{d}s_i) &\equiv \sqrt{\frac{\beta A^2_{ai}}{2\pi\hat{\tau}^t_{a\to i}}}\;\exp\left\{\frac{\beta}{2\hat{\tau}^t_{a\to i}}(A_{ai}s_i - z^t_{a\to i})^2\right\}\,\mathrm{d}s_i\,,
\end{aligned} \tag{15}$$

where the distribution parameters are given by

$$z^t_{a\to i} \equiv y_a - \sum_{j\neq i}A_{aj}x^t_{j\to a}, \qquad \hat{\tau}^t_{a\to i} \equiv \sum_{j\neq i}A^2_{aj}\tau^t_{j\to a}. \tag{16}$$

Proof. By an easy manipulation, we see that, for any Borel set S

$$\hat{\nu}^{t+1}_{a\to i}(S) = \mathbb{P}\Big\{y_a - \sum_{j\neq i} A_{aj}s_j \in A_{ai}S\Big\},$$

where $A_{ai}S = \{A_{ai}x \,:\, x\in S\}$. Here probability is over the random vector $(s_1, s_2, \dots, s_{i-1}, s_{i+1}, \dots, s_N)$, that is distributed according to the product measure $\nu^t_{1\to a}(s_1)\dots\nu^t_{N\to a}(s_N)$.

Consider the random variable $Z = y_a - \sum_{j\neq i} A_{aj}s_j$. According to the assumptions and the central limit theorem, Z is approximately normal. Clearly

$$\begin{aligned} \mathbb{E}(Z) &= y_a - \sum_{j\neq i} A_{aj}x^t_{j\to a}, \\ \mathrm{Var}(Z) &= \sum_{j\neq i} A^2_{aj}\tau^t_{j\to a}\,. \end{aligned}$$

The statement follows from Berry-Esseen Central limit theorem. □

Motivated by this lemma, we consider the computation of means and variances of the messages $\nu^{t+1}_{i\to a}(s_i)$. To state the result, it is convenient to introduce the family of densities

$$f_\beta(s;x,b) \equiv \frac{1}{z_\beta(x,b)}\,\exp\Big\{-\beta|s| - \frac{\beta}{2b}(s-x)^2\Big\}\,. \tag{17}$$

We also denote as follows its mean and variance (here Z has density $f_\beta(\,\cdot\,;x,b)$)

$$\mathsf{F}_\beta(x;b) \equiv \mathbb{E}_{f_\beta(\,\cdot\,;x,b)}(Z)\,, \qquad \mathsf{G}_\beta(x;b) \equiv \mathrm{Var}_{f_\beta(\,\cdot\,;x,b)}(Z)\,. \tag{18}$$

Notice that, because of Eq. (16), $\hat{\tau}^t_{i\to a}$ is expected to concentrate tightly, and we will therefore assume that it is independent of the edge (i,a).

Lemma 3.2. *Suppose that at iteration t, the messages from factor nodes to the variable nodes are set to $\hat{\nu}^t_{a\to i} = \hat{\phi}^t_{a\to i}$, with $\hat{\phi}^t_{a\to i}$ defined as in Eq. (15) with parameters $z^t_{a\to i}$ and $\hat{\tau}^t_{a\to i} = \hat{\tau}^t$. Then at the next iteration we have*

$$\nu^{t+1}_{i\to a}(s_i) = \phi^{t+1}_{i\to a}(s_i)\,\{1 + O(s_i^2/n)\}\,, \qquad \phi^{t+1}_{i\to a}(s_i) \equiv f_\beta(s_i; \sum_{b\neq a} A_{bi}z^t_{b\to i}, \hat{\tau}^t)\,. \tag{19}$$

In particular, the mean and variances of these messages are given by

$$x^{t+1}_{i\to a} = \mathsf{F}_\beta(\sum_{b\neq a} A_{bi}z^t_{b\to i}; \hat{\tau}^t), \qquad \tau^t_{i\to a} = \beta\,\mathsf{G}_\beta\Big(\sum_{b\neq a} A_{bi}z^t_{b\to i}; \hat{\tau}^t\Big).$$

Proof. Equation (19) is simply obtained by pointwise multiplication of the densities $\hat{\phi}^t_{a\to i}$ in Eq. (15), according to the general sum-product rule (12). More precisely, we obtain

$$\begin{aligned} \nu^{t+1}_{i\to a}(s_i) &\cong e^{-\beta|s_i|}\prod_{b\neq a}\hat{\nu}^t_{b\to i}(s_i) = \exp\Big\{-\beta|s_i| - \sum_{b\neq a}\frac{\beta}{2\hat{\tau}^t}(A_{ai}s_i - z^t_{b\to i})^2\Big\} \\ &\cong \exp\Big\{-\beta|s_i| - \frac{\beta}{2\hat{\tau}^t}\Big(\frac{n-1}{n}s_i^2 - 2s_i\sum_{b\neq a}A_{bi}z^t_{b\to i}\Big)\Big\}\,, \end{aligned}$$

which coincides with $\phi^{t+1}_{i\to a}(s_i)$ up to terms of order s_i^2/n. Finally the formulae for $x^{t+1}_{i\to a}$ and $\tau^t_{i\to a}$ follow directly from the definitions of F_β and G_β. □

Summarizing the above discussion, and approximating $\hat{\tau}^t_{a\to i}$ with an edge-independent quantity $\hat{\tau}^t$, we reach to the following algorithm.

$$x^{t+1}_{i\to a} = \mathsf{F}_\beta\Big(\sum_{b\neq a} A_{bi} z^t_{b\to i}; \hat{\tau}^t\Big)\,, \qquad z^t_{a\to i} \equiv y_a - \sum_{j\neq i} A_{aj} x^t_{j\to a}, \tag{20}$$

$$\hat{\tau}^{t+1} = \frac{\beta}{n}\sum_{i=1}^N \mathsf{G}_\beta\Big(\sum_b A_{bi} z^t_{b\to i}; \hat{\tau}^t\Big). \tag{21}$$

3.3 Large β limit

Although we gave simplified belief propagation formulas for a general value of β in the last section, but the special case $\beta\to\infty$ is of particular interest since the mode of the distribution introduced in Eq. (11) is the same as the Basis pursuit solution. The goal of this section is to derive explicit and simple formulas for the two functions F_β and G_β in the large β limit. Consider the soft threshold function

$$\eta(x;b) = \begin{cases} x-b & \text{if } b<x, \\ 0 & \text{if } -b\le x\le b, \\ x+b & \text{if } x<-b. \end{cases} \tag{22}$$

It is easy to confirm that,

$$\eta(x;b) = \arg\min_{s\in\mathbb{R}} \Big\{ |s| + \frac{1}{2b}(s-x)^2 \Big\}\,. \tag{23}$$

In the $\beta\to\infty$ limit, the integral that defines $\mathsf{F}_\beta(x;b)$ is dominated by the maximum value of the exponent, that corresponds to $s_* = \eta(x;b)$. Therefore $\mathsf{F}_\beta(x;b)\to\eta(x;b)$ as $\beta\to\infty$. The variance (and hence the function $\mathsf{F}_\beta(x;b)$) can be estimated by approximating the density $f_\beta(s;x,b)$ near s_*. Two cases can occur. If $s_*\neq 0$, then at this point the derivative of the exponent is equal to zero and therefore the density can well be approximate with a Gaussian distribution and $\mathsf{G}_\beta(x;b) = \Theta(1/\beta)$. On the other hand if $s_*=0$, $f_\beta(s;x,b)$ can be approximated by a Laplace distribution, leading to $\mathsf{G}_\beta(x;b) = \Theta(1/\beta^2)$. We summarize this discussion in the following lemma:

Lemma 3.3. *For bounded x, b, we have*

$$\begin{aligned} &\lim_{\beta\to\infty} \mathsf{F}_\beta(x;\beta) = \eta(x;b)\,, \\ &\lim_{\beta\to\infty} \beta\,\mathsf{G}_\beta(x;\beta) = b\,\eta'(x;b)\,. \end{aligned} \tag{24}$$

We are therefore led to the following message passing algorithm:

$$x^{t+1}_{i\to a} = \eta\Big(\sum_{b\neq a} A_{bi} z^t_{b\to i}; \hat{\tau}^t\Big)\,, \qquad z^t_{a\to i} \equiv y_a - \sum_{j\neq i} A_{aj} x^t_{j\to a}, \tag{25}$$

$$\hat{\tau}^{t+1} = \frac{\hat{\tau}^t}{N\delta}\sum_{i=1}^N \eta'\Big(\sum_b A_{bi} z^t_{b\to i}; \hat{\tau}^t\Big)\,. \tag{26}$$

3.4 From message passing to AMP

The updates in Eqs. (25), (26) are easy to implement but nevertheless the overall algorithm is still computationally expensive since it requires tracking of $2nN$ messages. The goal of this section is to further simplify the message passing update equations. The modification we introduce is expected to become negligible in the large system limit, but reduces the computation cost dramatically.

In order to justify approximation we assume that the messages can be approximated in the following way.

$$\begin{aligned} x^t_{i\to a} &= x^t_i + \delta x^t_{i\to a} + O(1/N), \\ z^t_{a\to i} &= z^t_a + \delta z^t_{a\to i} + O(1/N), \end{aligned} \tag{27}$$

with $\delta x^t_{i\to a}, \delta z^t_{a\to i} = O(\frac{1}{\sqrt{N}})$ (here the $O(\,\cdot\,)$ errors are uniform in the choice of the edge). We also consider a general message passing algorithms of the form

$$x^{t+1}_{i\to a} = \eta_t\Big(\sum_{b\neq a} A_{bi} z^t_{b\to i}\Big)\,, \qquad z^t_{a\to i} \equiv y_a - \sum_{j\neq i} A_{aj} x^t_{j\to a}, \tag{28}$$

with $\{\eta_t(\,\cdot\,)\}_{t\in\mathbb{N}}$ a sequence of differendiable nonlinear functions with bounded derivatives. Notice that the algorithm derived at the end of the previous section, cf. Eqs. (25), Eqs. (26), is indeed of this form, albeit with η_t non-differentiable at 2 points. This does not change the result, as long as the nonlinear functions are Lipschitz continuous. In the interest of simplicity, we shall stick to the differentiable model.

Lemma 3.4. *Suppose that the asymptotic behavior (27) holds for the message passing algorithm (28). Then x^t_i and z^t_a satisfy the following equations*

$$\begin{aligned} x^{t+1}_i &= \eta_t\Big(\sum_a A_{ia} z^t_a + x^t_i\Big) + o_N(1), \\ z^t_a &= y_a - \sum_j A_{aj} x^t_j + \frac{1}{\delta} z^{t-1}_a \langle \eta'_{t-1}(A^* z^{t-1} + x^{t-1})\rangle + o_N(1), \end{aligned}$$

where the $o_N(1)$ terms vanish as $N, n \to \infty$.

Proof. To prove the lemma we substitute (27) in the general equations (28) and write the Taylor expansion of the latter. The update equation for $z^t_{a\to i}$ yields

$$z^t_{a\to i} = \underbrace{y_a - \sum_{j\in[N]} A_{aj} x^t_j - \sum_{j\in[N]} A_{aj}\delta x^t_{j\to a}}_{z^t_a} + \underbrace{A_{ai} x^t_i}_{\delta z^t_{a_i}} + O(1/N).$$

For $x^{t+1}_{i\to a}$ we have

$$x^{t+1}_{i\to a} = \underbrace{\eta_t(\sum_{b\in[n]} A_{bi} z^t_b + \sum_{b\in[n]} A_{bi}\delta z^t_{b\to i})}_{x^t_i} \underbrace{- A_{ai} z^t_a \eta'_t(\sum_{b\in\partial i} A_{bi} z^t_b + \sum_{b\in\partial i} A_{bi}\delta z^t_{b\to i})}_{\delta x^t_{i\to a}} + O(1/N)\,.$$

In underbraces we have identified the various contributions. Substituting the expression indicated for $\delta x^t_{i\to a}$, $\delta z^t_{a\to i}$ we obtain the recursion for x^t_i and z^t_a. In particular x^t_i is updated according to

$$\begin{aligned} x^{t+1}_i &= \eta_t(\sum_{b\in[n]} A_{bi} z^t_b + \sum_{b\in[n]} A_{bi}\delta z^t_{b\to i}) + o(1) \\ &= \eta_t(\sum_{b\in[n]} A_{bi} z^t_b + \sum_{b\in[n]} A^2_{bi} x^t_i) + o(1) \\ &= \eta_t(\sum_{b\in[n]} A_{bi} z^t_b + x^t_i) + o(1)\,. \end{aligned}$$

For z_a^t we get

$$\begin{aligned} z_a^t &= y_a - \sum_{j\in[N]} A_{aj}x_j^t + \sum_{j\in[N]} A_{aj}^2 z_a^{t-1}\eta_{t-1}'(\sum_{b\in[n]} A_{bj}z_b^{t-1} + \sum_{b\in[n]} A_{aj}\delta z_{a\to j}^{t-1}) + o(1) \\ &= y_a - \sum_{j\in[N]} A_{aj}x_j^t + \frac{1}{n}z_a^{t-1}\sum_{j\in[N]}\eta'(\sum_{b\in[n]} A_{bi}z_b^{t-1} + x_i^{t-1}) + o(1) \\ &= y_a - \sum_{j\in[N]} A_{aj}x_j^t + \frac{1}{\delta}z_a^{t-1}\langle\eta_{t-1}(\sum_{b\in[n]} A_{bi}z_b^{t-1} + x_i^{t-1}) + o(1)\rangle\,. \end{aligned}$$

This conclues the proof. □

This theorem naturally suggest a simplified form of the iterations (25), (26). The resulting algorithm can be written in the vector notation as

$$\begin{aligned} x^{t+1} &= \eta(A^*z^t + x^t; \hat{\tau}^t)\,, \\ z^t &= y - Ax^t + \frac{1}{\delta}z^{t-1}\langle\eta'(A^*z^{t-1} + x_i^{t-1}; \hat{\tau}^{t-1})\rangle\,, \end{aligned} \tag{29}$$

where $\langle\,\cdot\,\rangle$ denotes the average entry of a a vector.

The recursion for $\hat{\tau}$ is also as follows.

$$\hat{\tau}^t = \frac{\hat{\tau}^{t-1}}{\delta}\langle\eta'(A^*z^{t-1} + x^t; \hat{\tau}^{t-1})\rangle\,. \tag{30}$$

4 AMP for reconstruction under soft constraints

Another popular reconstruction procedure in compressed sensing is the following optimization problem

$$\text{minimize } \lambda\|s\|_1 + \frac{1}{2}\|y - As\|_2^2. \tag{31}$$

In this section we describe another approximate message passing algorithm for solving this optimization problem. We will follow closely the four-step procedure already outlined in the previous section. The algorithm that is derived is very similar to the AMP algorithm introduced in the previous section. The only difference is in the update rule for the threshold level.

4.1 Construction of the graphical model

As before we define a joint density distribution on the variables $s = (s_1, \dots, s_N)$

$$\mu(\mathrm{d}s) = \frac{1}{Z}\prod_{i=1}^N \exp(-\beta\lambda|s_i|)\prod_{a=1}^n \exp\Big\{-\frac{\beta}{2}(y_a - (As)_a)^2\Big\}\,\mathrm{d}s\,. \tag{32}$$

Notice that –as in the previous case– the mode of this distribution coincides with the solution of the relevant problem (31). The distribution concentrates on its mode as $\beta\to\infty$. The sum-product algorithm is

$$\nu_{i\to a}^{t+1}(s_i) \cong \exp(-\beta\lambda|s_i|)\prod_{b\neq a}\nu_{b\to i}^t(s_i), \tag{33}$$

$$\hat{\nu}_{a\to i}^t(s_i) \cong \int \exp\Big\{-\frac{\beta}{2}(y_a - (As)_a)^2\Big\}\prod_{j\neq i}\mathrm{d}\nu_{j\to a}^t(s_j)\,. \tag{34}$$

4.2 Large system limit

The normal approximation Lemma is similar in form to the one given before, with two important differences: (i) The variance of the resulting messages is larger (because the constraint $y_a = (As)_a$ is only enforced softly); (ii) We can aprproximate the density of $\hat{\nu}^t_{a\to i}$ with a gaussian density (not just the corresponding distribution function) which is in fact stronger than the previous result.

Lemma 4.1. *Let $x^t_{j\to a}$ and $(\tau^t_{j\to a}/\beta)$ be, respectively, the mean and variance of the distribution $\nu^t_{j\to a}$, for the sum-product algorithm (33), (34). Assume further $\int |s_j|^3 \mathrm{d}\nu^t_{j\to a}(s_j) \le C_t$ uniformly in N, n. Then there exists a constant C'_t such that*

$$\begin{aligned}
\sup_{s_i\in\mathbb{R}} |\hat{\nu}^t_{a\to i}(s_i) - \hat{\phi}^t_{a\to i}(s_i)| &\le \frac{C'_t}{N(\hat{\tau}^t_{a\to i})^{3/2}}\,, \\
\hat{\phi}^t_{a\to i}(s_i) &\equiv \sqrt{\frac{\beta A^2_{ai}}{2\pi(1+\hat{\tau}^t_{a\to i})}}\,\exp\left\{-\frac{\beta}{2(1+\hat{\tau}^t_{a\to i})}(A_{ai}s_i - z^t_{a\to i})^2\right\}\,,
\end{aligned} \tag{35}$$

where the distribution parameters are given by

$$z^t_{a\to i} \equiv y_a - \sum_{j\ne i} A_{aj} x^t_{j\to a}, \qquad \hat{\tau}^t_{a\to i} \equiv \sum_{j\ne i} A^2_{aj}\tau^t_{j\to a}. \tag{36}$$

Proof. We have

$$\begin{aligned}
\hat{\nu}^t_{a\to i}(s_i) &\cong \mathbb{E}\exp\big(-\frac{\beta}{2}(y_a - A_{ai}s_i - \sum_{j\ne i} A_{aj}s_j)^2\big) \\
&\cong \mathbb{E}\exp\Big(-\frac{\beta}{2}(A_{ai}s_i - Z)^2\Big) \\
&\cong \mathbb{E}h_{s_i}(Z)\,,
\end{aligned}$$

Here expectation is over $s_1, s_2, \dots, s_{i-1}, s_{i+1}, \dots, s_N$ independent and distributed according to $\nu^t_{1\to a}, \dots \nu^t_{N\to a}$. Further, we defined $Z = y_a - \sum_{j\ne i} A_{aj}s_j$ and $h_{s_i}(z) \equiv \exp(-(\beta/2)(A_{ai}s_i - z)^2)$.

It is not hard to compute the mean and the variance of Z

$$\begin{aligned}
\mathbb{E}(Z) &= y_a - \sum_{j\ne i} A_{aj}x^t_{j\to a} = z^t_{a\to i}, \\
\mathrm{Var}(Z) &= \frac{1}{\beta}\sum_{j\ne i} A^2_{aj}\tau^t_{j\to a} = \frac{1}{\beta}\hat{\tau}^t_{a\to i}\,.
\end{aligned}$$

Let W be a normal random variable with the same mean and variance as Z. By a different form of Berry-Esseen central limit theorem mentioned in appendix A,

$$\Big|\mathbb{E}h_{s_i}(Z) - \mathbb{E}h_{s_i}(W)\Big| \le ||h'||_\infty \frac{C''_t}{N^{1/2}(\hat{\tau}^t_{a\to i})^{3/2}} \le \frac{C'''_t}{N^{1/2}(\hat{\tau}^t_{a\to i})^{3/2}}\,,$$

where $||h'||_\infty = \sup_t |h'(t)|$ is the infinity norm of h' (which is bounded by $\sqrt{\beta}$). We therefore get

$$\begin{aligned}
\sup_{s_i\in\mathbb{R}}\Big|\hat{\nu}^t_{a\to i}(s_i) - \mathbb{E}h_{s_i}(W)\Big| &\le \Big|\frac{\mathbb{E}h_{s_i}(Z)}{\int\mathbb{E}h_{s_i}(Z)ds_i} - \frac{\mathbb{E}h_{s_i}(W)}{\int\mathbb{E}h_{s_i}(W)ds_i}\Big| \\
&\le \Big|\frac{\mathbb{E}h_{s_i}(Z) - \mathbb{E}h_{s_i}(W)}{\int\mathbb{E}h_{s_i}(Z)ds_i}\Big| + \Big|\frac{\int\mathbb{E}h_{s_i}(W)ds_i - \int\mathbb{E}h_{s_i}(Z)ds_i}{\int\mathbb{E}h_{s_i}(W)ds_i\int\mathbb{E}h_{s_i}(Z)ds_i}\mathbb{E}h_{s_i}(Z)\Big| \\
&\le \frac{C'_t}{N(\hat{\tau}^t_{a\to i})^{3/2}}\,.
\end{aligned}$$

The last inequality is due to the following facts,

$$\int \mathbb{E} h_{s_i}(Z) ds_i = \mathbb{E} \int h_{s_i}(Z) ds_i = \frac{\sqrt{2\pi}}{\sqrt{A_{ai}^2 \beta}},$$

$$\int \mathbb{E} h_{s_i}(W) ds_i = \frac{\sqrt{2\pi}}{\sqrt{A_{ai}^2 \beta}}.$$

The proof is completed by computing $\mathbb{E}h_{s_i}(W)$. Such computation amounts to a straightforward gaussian integral, yielding $\mathbb{E}h_{s_i}(W) \cong \hat{\phi}_{a\to i}^t(s_i)$. □

The update rule for variable-to-factor node messages, cf. Eq. (33), is identical to the one used in the case of hard constraints, cf. Eq. (12), apart from the factor λ in the exponent. Keeping track of this term we obtain the following result.

Lemma 4.2. *Suppose that at iteration t, the messages from factor nodes to the variable nodes are set to $\hat{\nu}_{a\to i}^t = \hat{\phi}_{a\to i}^t$, with $\hat{\phi}_{a\to i}^t$ defined as in Eq. (58). with parameters $z_{a\to i}^t$ and $\hat{\tau}_{a\to i}^t = \hat{\tau}^t$. Then at the next iteration we have*

$$\nu_{i\to a}^{t+1}(s_i) = \phi_{i\to a}^{t+1}(s_i) \left\{1 + O(s_i^2/n)\right\}, \qquad \phi_{i\to a}^{t+1}(s_i) \equiv \lambda\, f_\beta(\lambda s_i; \lambda \sum_{b\neq a} A_{bi} z_{b\to i}^t, \lambda^2(1+\hat{\tau}^t)) . \tag{37}$$

In particular, the mean and variances of these messages are given by

$$x_{i\to a}^{t+1} = \frac{1}{\lambda} \mathsf{F}_\beta(\lambda \sum_{b\neq a} A_{bi} z_{b\to i}^t; \lambda^2(1+\hat{\tau}^t)), \qquad \tau_{i\to a}^t = \frac{\beta}{\lambda^2} \mathsf{G}_\beta\Big(\lambda \sum_{b\neq a} A_{bi} z_{b\to i}^t; \lambda^2(1+\hat{\tau}^t)\Big),$$

where, $f_\beta, F_\beta, andG_\beta$ are defined in Eqs. 17 and 18.

The proof is very similar to the proof of Lemma 3.2 and for the sake of brevity we do not mention it here.

As a summary, we get the following simple iterative algorithm which at each iteration equivalent to the corresponding iteration of the message passing algorithm.

$$x_{i\to a}^{t+1} = \frac{1}{\lambda} \mathsf{F}_\beta\Big(\lambda \sum_{b\neq a} A_{bi} z_{b\to i}^t; \lambda^2(1+\hat{\tau}^t)\Big), \qquad z_{a\to i}^t \equiv y_a - \sum_{j\neq i} A_{aj} x_{j\to a}^t, \tag{38}$$

$$\hat{\tau}^{t+1} = \frac{\beta}{\lambda^2 n} \sum_{i=1}^N \mathsf{G}_\beta\Big(\lambda \sum_b A_{bi} z_{b\to i}^t; \lambda^2(1+\hat{\tau}^t)\Big). \tag{39}$$

As before the next step is to derive the algorithm in the limit $\beta \to \infty$ which is the most interesting regime and is equivalent to basis pursuit denoising problem.

4.3 Large β limit

Applying Lemma 3.3 to Eqs. (38), (39) they reduce –in the large β limit– to

$$x_{i\to a}^{t+1} = \eta\Big(\sum_{b\neq a} A_{bi} z_{b\to i}^t; \lambda(1+\hat{\tau}^t)\Big), \qquad z_{a\to i}^t \equiv y_a - \sum_{j\neq i} A_{aj} x_{j\to a}^t,$$

$$\hat{\tau}^{t+1} = \frac{1+\hat{\tau}^t}{N\delta} \sum_{i=1}^N \eta'\Big(\sum_b A_{bi} z_{b\to i}^t; \lambda(1+\hat{\tau}^t)\Big),$$

where we used the invariance property $\eta(a\,x; a\,b) = a\eta(x; b)$ valid for any $a > 0$. If we call $\lambda\hat{\tau}^t = \gamma^t$ the new form of the AMP algorithm is,

$$x^{t+1}_{i\to a} = \eta\Big(\sum_{b\neq a} A_{bi} z^t_{b\to i}; \lambda + \gamma^t\Big)\,, \qquad z^t_{a\to i} \equiv y_a - \sum_{j\neq i} A_{aj} x^t_{j\to a}, \tag{40}$$

$$\gamma^{t+1} = \frac{\lambda + \gamma^t}{N\delta} \sum_{i=1}^{N} \eta'\Big(\sum_b A_{bi} z^t_{b\to i}; \lambda + \gamma^t\Big)\,, \tag{41}$$

These expression should be compared with Eqs. (40), (41) for the basis pursuit algorithm. The only difference is just in the threshold value.

4.4 From message passing to AMP

Again, this algorithm can be considerably simplified using the Lemma 3.4. In matrix notation we obtain the following equations

$$\begin{aligned} x^t &= \eta(x^t + A^* z^t; \lambda + \gamma^t), & (42)\\ z^{t+1} &= y - Ax^t + \frac{1}{\delta} z^t \langle \eta'(x^{t-1} + A^* z^{t-1}),\rangle & (43) \end{aligned}$$

which generalize Eqs. (29) and (29). The threshold level is computed iteratively as follows

$$\gamma^{t+1} = \frac{\lambda + \gamma^t}{\delta} \langle \eta'(Az^t + x^t; \gamma^t + \lambda)\rangle\,. \tag{44}$$

4.5 Comments

Threshold level. The derivation presented above provides a 'parameter free' algorithm. The threshold level $\hat{\tau}^t$ or γ^t is fixed by the recursions (30), (41). In the basis pursuit problem, one could take the alternative point of view that $\hat{\tau}^t$ is a parameter that can be optimized over. This point of view was adopted in [24]. For the case of Lasso it is again possible to consider the threshold as a free parameter and then tune it such that the fixed point of iteration satisfies the KKT conditions. This approach has been adopted in [12]. The analysis and comparison of these thresholding policies are presented in section 5. We call the AMP algorithm with the thresholding policy introduced in (30) and (41) AMP.0 and AMP.A respectively. When we tune the algorithm to get the best phase transition the algorithm is called AMP.M where M stands for minimaxity. Finally, when the free parameter is tuned to satisfy the KKT conditions the algorithm is called AMP.T.

Mathematical derivation of the AMP. We showed that in a specific limit (large systems, and large β) the sum-product update rules can be considerably simplified to get the update rules (29), (42). Let us emphasize that our proofs concern just a single step of the iterative procedure. Therefore they do not prove that the (averages and variances) of the sum-product message are precisely tracked by Eqs. (29), (42). It could be that the error terms in our approximation, while negligible at each step, conjure up to become large after a finite number of iterations. We do not expect this to happen, but it is nevertheless an open mathematical problem.

5 State evolution

Recently the authors introduced the state evolution framework for analyzing the performance of the AMP algorithm [13]. This approach has been also rigorously proved recently [1]. Consider the following iterative algorithm

$$x^{t+1} = \eta_t(x^t + A^* z^t),$$
$$z^t = y - Ax^t + \frac{1}{\delta}\langle \eta'_{t-1}(A^* z^{t-1} + x^{t-1})\rangle. \tag{45}$$

where $\eta_t(.)$ is a function that may also depend on the iteration. We recall the following result from [1]. Let $\{A(N)\}$ be a sequence of sensing matrices $A \in \mathbb{R}^{n\times N}$ indexed by N, with iid entries $A_{ij} \sim N(0, 1/n)$, and assume $n/N \to \delta$. Consider further a sequence of signals $\{x_0(N)\}_{N\geq 0}$, whose empirical distributions converge to a probability measure p_{X_0} on $\mathbb{R}$ with bounded $(2k-2)^{\text{th}}$ moment, and assume $\mathbb{E}_{\hat{p}}(X_0^{2k-2}) \to \mathbb{E}_{p_{X_0}}(X_0^{2k-2})$ as $N \to \infty$ for some $k \geq 2$.

Theorem 5.1. *For any pseudo-Lipschitz function $\psi : \mathbb{R}^2 \to \mathbb{R}$ we have,*

$$\lim_{N\to\infty} \frac{1}{N}\sum_{i=1}^{N} \psi(x_i^{t+1}, x_{0,i}) = \mathbb{E}[\psi(\eta_t(X_0 + \tau_t Z), X_0)],$$

with $X_0 \sim p_{X_0}$ and $Z \sim N(0,1)$ independent.

According to the above theorem we can consider the parameter τ_t as the state of the algorithm and track the behavior of this state variable across iterations. This is called state evolution [13]. If we consider the measurements to be of the form $y = Ax + w$ with $w \sim N(0, \sigma^2 I_n)$, the state evolution equation is given by,

$$\tau_{t+1}^2 = \sigma^2 + \frac{1}{\delta}\mathbb{E}[\eta_t(X_0 + \tau_t Z) - X_0]^2, \tag{46}$$

where again $X_0 \sim p_{X_0}$ and $Z \sim N(0,1)$ independent.
Although the state evolution equation has been proved for the case of Gaussian measurement matrix, its validity has been carefully verified through extensive simulations for other random matrices [13]. According to the state evolution we can predict the performance of the algorithms derived in this paper theoretically. For the AMP.A algorithm the state evolution can be written as,

$$\tau_{t+1}^2 = \sigma^2 + \frac{1}{\delta}\mathbb{E}[\eta(X_0 + \tau_t Z; \lambda + \gamma^t) - X_0]^2,$$
$$\gamma^{t+1} = \frac{\gamma^t + \lambda}{\delta}\mathbb{E}[\eta'(X_0 + \tau_t Z; \lambda + \gamma^t)]. \tag{47}$$

Figure (1) shows the match between the predictions of the state evolution and the Monte Carlo simulation results.

5.1 Exactly Sparse Solution

Suppose there is no measurement noise in the system, i.e. $y = As_o$. Also the the elements of s_o are drawn from $(1-\epsilon)\delta_0(s_{oi}) + \epsilon G(s_{oi})$ where G is a density function on $\mathbb{R}^+$ [2] without a point mass at 0 and define $\rho = \epsilon/\delta$. We are interested in the solution of the basis pursuit problem and therefore we consider the AMP.0 algorithm. It is easy to see that the state evolution equation for this case is given by,

$$\tau_{t+1}^2 = \frac{1}{\delta}\mathbb{E}[\eta(X_0 + \tau_t Z; \gamma^t) - X_0]^2,$$
$$\gamma^{t+1} = \frac{\gamma^t}{\delta}\mathbb{E}[\eta'(X_0 + \tau_t Z; \gamma^t)]. \tag{48}$$

[2] This is just for the simplicity of exposition. The results can be easily extended to other cases.

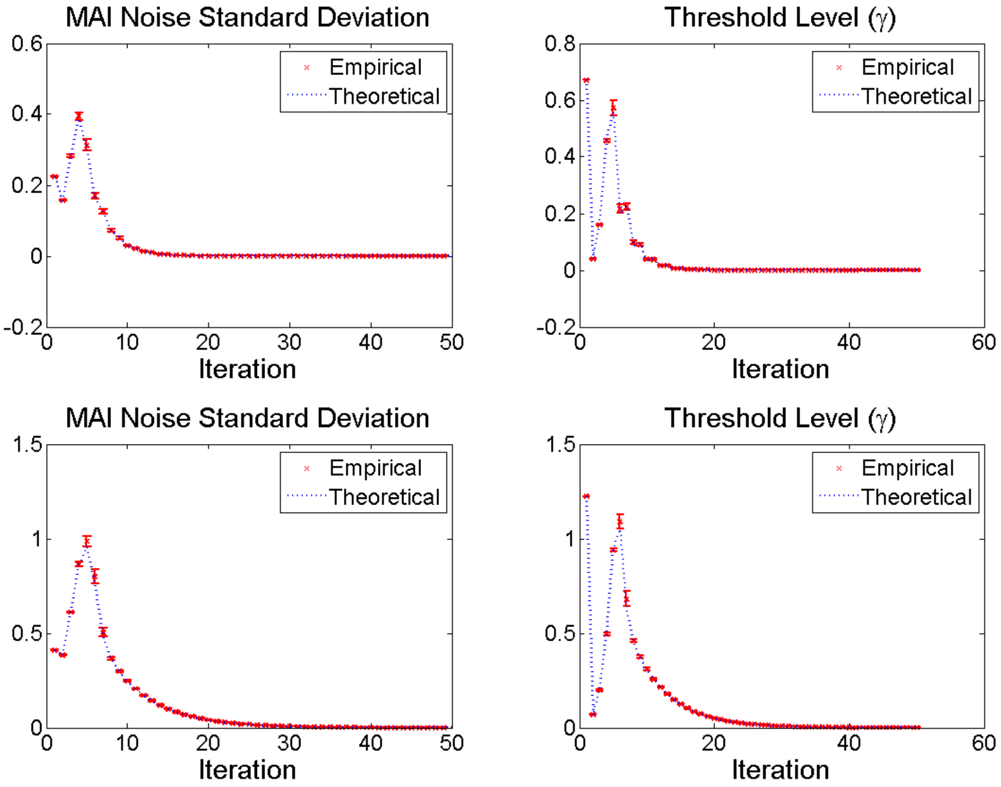

Figure 1: Comparison of state evolution predictions against observations. Top row: $N = 4000$, $\delta = .2$ and $\rho = 0.05$. Bottom row: $N = 4000$, $\delta = .3$ and $\rho = 0.17$. Each red point is the average of 50 Monte Carlo samples and the bars show the 95 percent confidence interval.

Lemma 5.2. *Consider the state evolution equation 48. Suppose that the sparsity level ϵ is small enough such that the algorithm converges to the correct answer, i.e. $(\tau_t, \gamma_t) \to (0,0)$. Then*

$$\lim_{t\to\infty} \frac{\tau_t}{\gamma_t} = c, \tag{49}$$

where c is a finite non-zero constant.

Proof. The proof is by contradiction. The goal is to rule out the two cases $c = 0$ and $c = \infty$. Although the proof is a simple application of dominated convergence theorem, but for the sake of clarity we mention the proof here.
case I: $c = \infty$. We know that,

$$\frac{\gamma^{t+1}}{\gamma^t} = \frac{1}{\delta}\mathbb{E}\eta'(X + \tau^t Z; \gamma^t) = \frac{1-\epsilon}{\delta}\mathbb{E}\eta'(\tau^t Z; \gamma^t) + \frac{\epsilon}{\delta}\mathbb{E}_{X\sim G}\eta'(X + \tau^t Z; \gamma^t) =$$
$$\frac{1-\epsilon}{\delta}\mathbb{E}\eta'(Z; \gamma^t/\tau^t) + \frac{\epsilon}{\delta}\mathbb{E}_{X\sim G}\eta'(X + \tau^t Z; \gamma^t).$$

By taking the limit and using the dominated convergence theorem we get,

$$\lim_{t\to\infty} \frac{\gamma^{t+1}}{\gamma^t} = \frac{1-\epsilon}{\delta} + \frac{\epsilon}{\delta} = \frac{1}{\delta} > 1,$$

and this means that γ^t is not going to zero which is a contradiction.
Case II: $c = 0$.

$$\frac{\tau_{t+1}^2}{\gamma_t^2} = \frac{1}{\delta}\mathbb{E}[\eta(X_0/\gamma_t + Z\tau_t/\gamma_t; 1) - X_0/\gamma_t]^2 =$$
$$\frac{1-\epsilon}{\delta}\mathbb{E}[\eta(Z\tau_t/\gamma_t; 1)]^2 + \frac{\epsilon}{\delta}\mathbb{E}_{X\sim G}[\eta(X_0/\gamma_t + Z\tau_t/\gamma_t; 1) - X_0/\gamma_t]^2. \tag{50}$$

Clearly we can use the dominated convergence theorem again to get the limit and therefore,

$$\lim_{t\to\infty} \frac{\tau_{t+1}^2}{\gamma_t^2} = \frac{\epsilon}{\delta} > 0, \tag{51}$$

which is again a contradiction and the proof is complete. □

Inspired with the above lemma we can introduce a simpler AMP algorithm.

$$x^{t+1} = \eta(x^t + A^* z^t; \theta\sigma^t),$$
$$z^t = y - Ax^t + \frac{1}{\delta} z^{t-1}\langle \eta'(x^{t-1} + A^* z^{t-1}; \theta\sigma^{t-1})\rangle, \tag{52}$$

where σ^t is the standard deviation of the MAI noise at iteration t and θ is a constant number. This is the algorithm that was proposed by the authors in [13]. Here θ is a parameter that has to be tuned before applying the algorithm. The state evolution for this algorithm is simpler since there is just one state variable.

$$\tau_{t+1}^2 \mapsto \Psi(\tau_t^2) = \frac{1}{\delta}\mathbb{E}[\eta(X_0 + \tau_t Z; \theta\tau_t) - X_0]^2. \tag{53}$$

The performance of these two algorithms is very similar as $\tau_t \to 0$. This is formally stated in the following lemma.

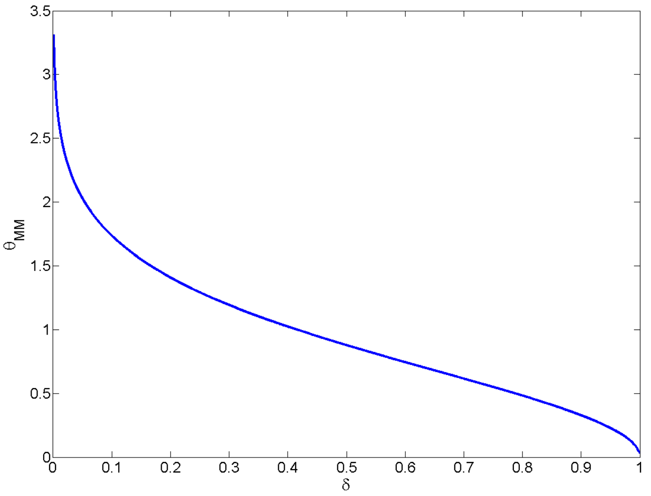

Figure 2: The maxmin optimal values of θ as proposed in [13].

Lemma 5.3. *Suppose that* $\lim_{\tau \to 0} \frac{\gamma}{\tau} \to c$ *we then have,*

$$\lim_{\tau \to 0} \frac{\mathbb{E}[\eta(X_0 + \tau Z; \gamma) - X_0]^2/\tau^2}{\mathbb{E}[\eta(X_0 + \tau Z; c\tau) - X_0]^2/\tau^2} = 1.$$

The proof of this lemma is very simple and is omitted. The following result taken from [13] helps us analyze the performance of the new thresholding policy.

Lemma 5.4. *The* Ψ *function is concave and its derivative at* $\tau_t = 0$ *is independent of* G *which is the distribution of the non-zero coefficients .*

Let $\rho = \epsilon/\delta$ and define $\rho^*_{LS}(\delta) \equiv \sup_\theta \sup\{\rho : \frac{d\Psi}{d\tau^2}|_0 < 1\}$. The optimal value of θ is represented by θ_{MM} which stands for the maximin. The value of θ_{MM} as a function of δ is shown in figure 2. For more information about θ_{MM} refer to the appendix. Using the above two lemmas it is easy to prove the following theorem.

Theorem 5.5. *If* $\rho > \rho^*_{LS}(\delta)$, *AMP.0 does not converge to the correct answer, i.e.* $(\tau_t, \gamma_t) \nrightarrow (0, 0)$. *On the other hand for* $\rho < \rho^*_{LS}(\delta)$ *the AMP.M algorithm converges to the correct answer.*

According to the above theorem from the sparsity measurement point of view AMP.M is at least as good as AMP.0. The only advantage of AMP.0 is that it does not need any tuning. We will show in the next section that for most of the values of δ the phase transitions of AMP.0 and AMP.A happen at the same place. But for small values of δ, the recursion of AMP.A suffer from oscillatory phenomena.

5.1.1 Comparison of AMP.A and AMP.M

In the previous section we showed that the phase transition of AMP.0 algorithm can not surpass the phase transition of AMP.M. However the main question is if the state evolution of AMP.0 always converges to the correct answer for $\rho < \rho^*_{LS}$. In other words what is the actual phase transition region of the AMP.A algorithm? In order to answer this question precisely, we again use the state evolution equation. We consider 200 equisapced points on the $[0,1]$ for the values of δ. For each value of δ we also consider 200 equi-spaced values of ρ. For each pair (δ,ρ) we run the state evolution for 500 iterations and measure the ℓ_2 norm of the estimate after a) 50, (b) 100 (c) 200 and (d) 500 iterations. If $\frac{\|\hat{x}^t - s_o\|_2}{\|s_o\|_2} < .001$, we declare success. With this method we calculate the phase transition of the AMP.A algorithm. In this simulation we have chosen the input ensemble from a constant amplitude ensemble which is known to be the least favorable distribution for approximate message passing algorithms [24]. Figure 3 compares the phase transition of the AMP.0 algorithm derived by this method with the phase transition of AMP.M or basis pursuit algorithm. As it is seen in this figure above $\delta > 0.2$ the phase transitions are indistinguishable. However below $\delta = 0.2$ the extra state variable γ causes some instability in the recursive equations that does not exist in AMP.M.

6 Extensions

So far, we have considered the general compressed sensing problem. However in some applications more information is known about the original signal. In this section we consider two of these scenarios and derive the corresponding approximate message passing algorithms.

6.1 Positivity Constraint

Suppose that the signal is known to lie in the positive orthant, i.e. $s_{o,i} \geq 0 \quad \forall i$. It has been proved that this extra information may be used properly to improve the phase transition region of the ℓ_1 minimization [15]. This information can be easily incorporated into the message passing algorithm. In this section we just consider the BPDN problem with the above constraint. The BP problem is a very simple modification of this approach and is therefore skipped in the paper.

6.1.1 Large system limit

Define a joint probability density on the variables $s = (s_1, \dots, s_N)$

$$\mu(\mathrm{d}s) = \frac{1}{Z}\prod_{i=1}^{N} \exp(-\beta\lambda s_i)\mathbb{I}_{\{s_i>0\}} \prod_{a=1}^{n} \exp\Big\{-\frac{\beta}{2}(y_a - (As)_a)^2\Big\}\,\mathrm{d}s\,. \tag{54}$$

As before the messages in the sum-product message passing algorithm are

$$\nu^{t+1}_{i\to a}(s_i) \;\cong\; \exp(-\beta\lambda s_i)\mathbb{I}\{s_i > 0\}\prod_{b\neq a}\nu^t_{b\to i}(s_i), \tag{55}$$

$$\hat{\nu}^t_{a\to i}(s_i) \;\cong\; \int \exp\Big\{-\frac{\beta}{2}(y_a - (As)_a)^2\Big\}\prod_{j\neq i}\mathrm{d}\nu^t_{j\to a}(s_j)\,. \tag{56}$$

Clearly the messages from the functional nodes to the variable nodes have exactly the same form and therefore the following lemma is the immediate result of theorem 4.1.

Lemma 6.1. *Let $x^t_{j\to a}$ and $(\tau^t_{j\to a}/\beta)$ be, respectively, the mean and variance of the distribution $\nu^t_{j\to a}$, for the sum-product algorithm (33), (34). Assume further $\int |s_j|^3 \mathrm{d}\nu^t_{j\to a}(s_j) \leq C_t$ uniformly in N, n. Then*

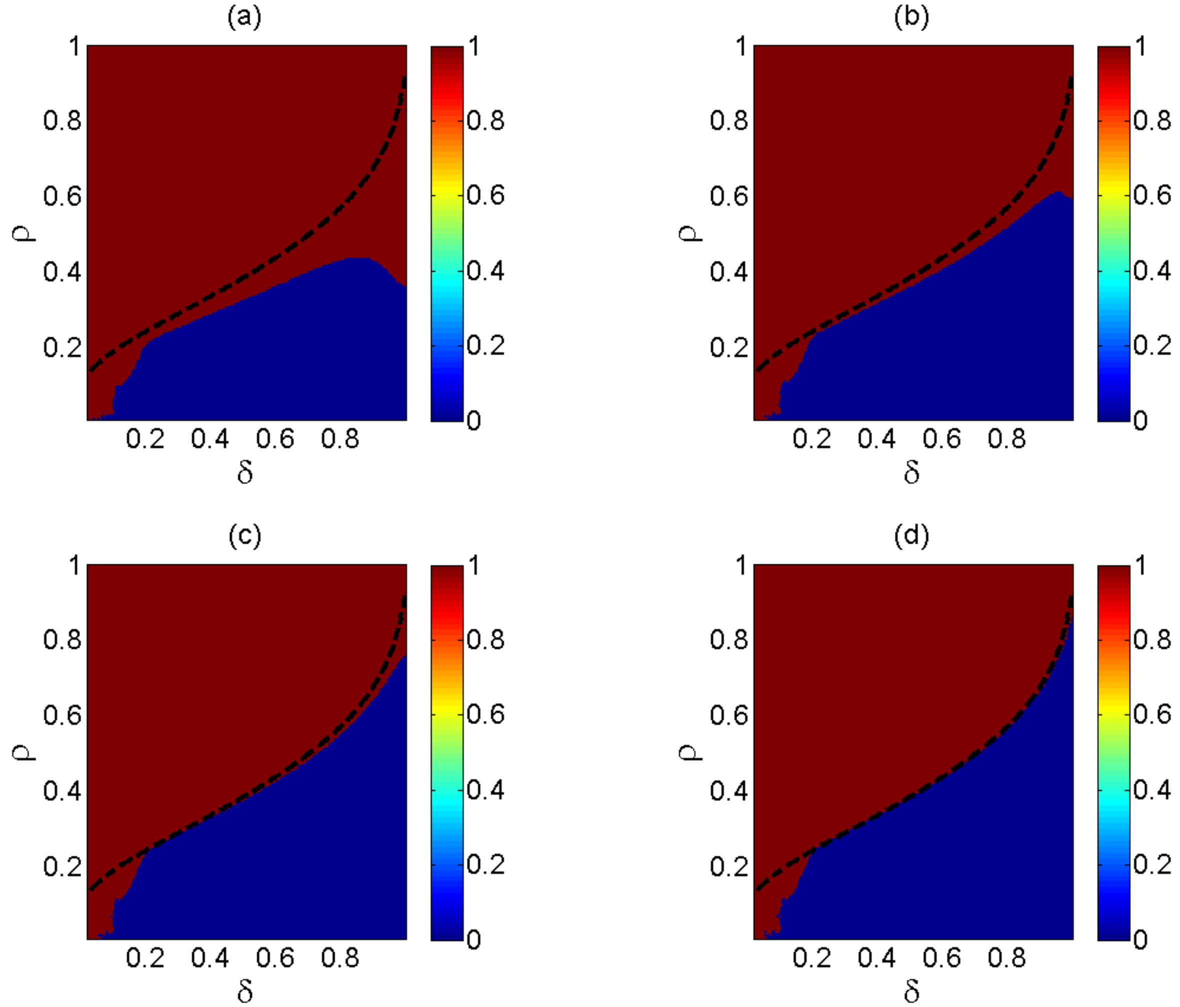

Figure 3: Theoretical phase transition of AMP.A after (a)50 (b) 100 (c) 200 and (d) 500 iterations. Dotted line is the phase transition curve of the basis pursuit problem derived in [14] and AMP.M [13].

there exists a constant C'_t such that

$$\begin{aligned}
\sup_{s_i\in\mathbb{R}} |\hat{\nu}^t_{a\to i}(s_i) - \hat{\phi}^t_{a\to i}(s_i)| \;&\leq\; \frac{C'_t}{N(\hat{\tau}^t_{a\to i})^3}\,, &(57)\\
\hat{\phi}^t_{a\to i}(s_i) \;&\equiv\; \sqrt{\frac{\beta A_{ai}^2}{2\pi(1+\hat{\tau}^t_{a\to i})}}\;\exp\left\{-\frac{\beta}{2(1+\hat{\tau}^t_{a\to i})}(A_{ai}s_i - z^t_{a\to i})^2\right\}\;, &(58)
\end{aligned}$$

where the distribution parameters are given by

$$z^t_{a\to i} \equiv y_a - \sum_{j\neq i} A_{aj}x^t_{j\to a}, \qquad \hat{\tau}^t_{a\to i} \equiv \sum_{j\neq i} A_{aj}^2\tau^t_{j\to a}. \tag{59}$$

Define

$$f^+_\beta(s;x,b) \equiv \frac{1}{z_\beta(x,b)}\exp\{-\beta s - \frac{\beta}{2b}(s-x)^2\}, \tag{60}$$

and

$$F^+_\beta(x;b) \equiv \mathbb{E}_{f_\beta(\cdot;x,b)}(Z), \qquad G^+_\beta(x;b) \equiv \mathrm{Var}_{f_\beta(\cdot;x,b)}(Z). \tag{61}$$

It is easy to prove that,

Lemma 6.2. *Suppose that at iteration t, the messages from factor nodes to the variable nodes are set to $\hat{\nu}^t_{a\to i} = \hat{\phi}^t_{a\to i}$, with $\hat{\phi}^t_{a\to i}$ defined as in Eq. (58). with parameters $z^t_{a\to i}$ and $\hat{\tau}^t_{a\to i} = \hat{\tau}^t$. Then at the next iteration we have*

$$\nu^{t+1}_{i\to a}(s_i) = \phi^{t+1}_{i\to a}(s_i)\,\{1+O(s_i^2/n)\}\,, \qquad \phi^{t+1}_{i\to a}(s_i) \equiv \lambda\, f^+_\beta(\lambda s_i; \lambda\sum_{b\neq a} A_{bi}z^t_{b\to i}, \lambda^2(1+\hat{\tau}^t))\,. \tag{62}$$

In particular, the mean and variances of these messages are given by

$$x^{t+1}_{i\to a} = \frac{1}{\lambda}\,\mathsf{F}^+_\beta(\lambda\sum_{b\neq a} A_{bi}z^t_{b\to i}; \lambda^2(1+\hat{\tau}^t)), \qquad \tau^t_{i\to a} = \frac{\beta}{\lambda^2}\,\mathsf{G}^+_\beta\Big(\lambda\sum_{b\neq a} A_{bi}z^t_{b\to i}; \lambda^2(1+\hat{\tau}^t)\Big),$$

where, f^+_β, F^+_β and G^+_β are defined in Eqs. 60 and 61.

6.1.2 Large β limit

Consider the following new form of the soft thresholding function.

$$\eta^+(x;b) = \begin{cases} x-b & \text{if } b<x,\\ 0 & \text{if } -b\leq x\leq b. \end{cases} \tag{63}$$

b in this equation is assumed to be larger than 0. As before when $\beta\to\infty$, $F^+_\beta(x;\beta)$ and $G^+_\beta(x;\beta)$ can be simplified even more.

Lemma 6.3. *For bounded x,b, we have*

$$\begin{aligned}
&\lim_{\beta\to\infty} F^+_\beta(x;\beta) = \eta^+(x;b),\\
&\lim_{\beta\to\infty} \beta G^+_\beta(x;\beta) = b\eta'^+(x;b).
\end{aligned}$$

we are therefore led to the following message passing algorithm,

$$x_{i\to a}^{t+1} = \eta^+\Big(\sum_{b\neq a} A_{bi} z_{b\to i}^t; \hat{\tau}^t\Big)\,, \qquad z_{a\to i}^t \equiv y_a - \sum_{j\neq i} A_{aj} x_{j\to a}^t, \tag{64}$$

$$\hat{\tau}^{t+1} = \frac{\hat{\tau}^t}{N\delta} \sum_{i=1}^N \eta'^+\Big(\sum_b A_{bi} z_{b\to i}^t; \hat{\tau}^t\Big)\,. \tag{65}$$

Finally by similar arguments we can reach to the following approximate message passing algorithm.

$$\begin{aligned}
x^{t+1} &= \eta^+(A^* z^t + x^t; \hat{\tau^t}),\\
z^t &= y - Ax^t + \frac{1}{\delta}\langle \eta'(A^* z^{t-1} + x^{t-1}; \hat{\tau}^{t-1})\rangle,\\
\hat{\tau}^t &= \frac{\hat{\tau}^t}{\delta}\langle \eta'(A^* z^{t-1} + x^{t-1}; \hat{\tau}^{t-1})\rangle.
\end{aligned}$$

6.2 AMP for reconstruction with prior information

In many compressed sensing applications it is not realistic to assume that the signal s is random with a known distribution. Nevertheless, it might be possible in specific scenarios to estimate the input distribution. Further, the case of known signal distribution provides a benchmark for other approaches.

6.2.1 Construction of the graphical model

Let $\rho = \rho_1 \times \rho_2 \cdots \times \rho_N$ be a joint probability distribution on the variables $s_1, s_2, \dots, s_N$. It is then natural to consider the joint distribution

$$\mu(\mathrm{d}s) = \frac{1}{Z}\prod_{a=1}^n \exp\Big\{-\frac{\beta}{2}(y_a - (As)_a)^2\Big\}\prod_{i=1}^N \rho_i(\mathrm{d}s_i)\,, \tag{66}$$

since μ is the *a posteriori* distribution of s, when $y = As + z$ is observed, with z a noise vector with i.i.d. normal entries and independent of s. The sum-product update rules are

$$\begin{aligned}
\nu_{i\to a}^{t+1}(\mathrm{d}s_i) &\cong \prod_{b\neq a} \hat{\nu}_{b\to i}^t(s_i)\,\rho_i(\mathrm{d}s_i)\,,\\
\nu_{a\to i}^t(s_i) &\cong \int \exp\Big\{-\frac{\beta}{2}(y_a - (As)_a)^2\Big\}\prod_{j\neq i}\nu_{j\to a}^t(\mathrm{d}s_j)\,.
\end{aligned}$$

Notice that the above update rules are well defined. At each iteration t, the message $\nu_{i\to a}^{t+1}(\mathrm{d}s_i)$ is a probability measure on $\mathbb{R}$, and Eq. (67) gives its density with respect to ρ_i. The message $\nu_{a\to i}^t(s_i)$ is instead a non-negative measurable function (equivalently, a density) given by Eq. (68).

It is easy to see that the case studied in the previous section corresponds to choosing the ρ_i's to be identical exponential distributions.

6.2.2 Large system limit

In view the parallel between the update equations Eq. (67), (68) and (33), (34) it is easy to realize that Lemma 4.1 applies verbatimly to the algorithm described above.

In order to formulate the analogous of Lemma 6.2, we introduce the following family of measures over $\mathbb{R}$:

$$f_i(\mathrm{d}s;x,b) \equiv \frac{1}{z_\beta(x,b)}\ \exp\Big\{-\frac{\beta}{2b}(s-x)^2\Big\}\,\rho_i(\mathrm{d}s)\,, \tag{67}$$

indexed by $i\in[N]$, $x\in\mathbb{R}$, $b\in\mathbb{R}_+$ (we think β as fixed). We use this notation for its mean and variance (here $Z\sim f_i(\,\cdot\,;x,b)$)

$$\mathsf{F}_i(x;b)\equiv \mathbb{E}_{f_i(\,\cdot\,;x,b)}(Z)\,,\qquad \mathsf{G}_i(x;b)\equiv \mathrm{Var}_{f_i(\,\cdot\,;x,b)}(Z)\,. \tag{68}$$

These functions have a natural estimation theoretic interpretation. Let X_i be a random variable with distribution ρ_i, and assume that $\widetilde{Y}_i = X_i + W_i$ is observed with W_i gaussian noise with variance b/β. The above functions are –respectively– the conditional expectation and conditional variance of X_i, given that $\widetilde{Y}_i = x$:

$$\mathsf{F}_i(x;b) = \mathbb{E}(X_i|\widetilde{Y}_i = x)\,,\qquad \mathsf{G}_i(x;b) = \mathrm{Var}(X_i|\widetilde{Y} = x)\,. \tag{69}$$

With these definitions, it is immediate to prove the following analogous of Lemma 6.2.

Lemma 6.4. *Suppose that at iteration t, the messages from factor nodes to the variable nodes are set to $\hat{\nu}^t_{a\to i} = \hat{\phi}^t_{a\to i}$, with $\hat{\phi}^t_{a\to i}$ defined as in Eq. (58). with parameters $z^t_{a\to i}$ and $\hat{\tau}^t_{a\to i} = \hat{\tau}^t$. Then at the next iteration we have*

$$\nu^{t+1}_{i\to a}(s_i) = \phi^{t+1}_{i\to a}(s_i)\,\{1+O(s_i^2/n)\}\,,\qquad \phi^{t+1}_{i\to a}(s_i)\equiv f_i(s_i;\sum_{b\neq a} A_{bi} z^t_{b\to i},(1+\hat{\tau}^t))\,. \tag{70}$$

In particular, the mean and variances of these messages are given by

$$x^{t+1}_{i\to a} = \mathsf{F}_i(\sum_{b\neq a} A_{bi} z^t_{b\to i};(1+\hat{\tau}^t)),\qquad \tau^t_{i\to a} = \beta\mathsf{G}_i\Big(\sum_{b\neq a} A_{bi} z^t_{b\to i};(1+\hat{\tau}^t)\Big).$$

If we let $\hat{\tau}^{t+1}_{i\to a} = \hat{\tau}^{t+1}$ for all edges (i,a) we get the message passing algorithm

$$x^{t+1}_{i\to a} = \mathsf{F}_i\Big(\sum_{b\neq a} A_{bi} z^t_{b\to i};(1+\hat{\tau}^t)\Big)\,,\qquad z^t_{a\to i}\equiv y_a - \sum_{j\neq i} A_{aj} x^t_{j\to a}, \tag{71}$$

$$\hat{\tau}^{t+1} = \frac{\beta}{n}\sum_{i=1}^N \mathsf{G}_i\Big(\lambda\sum_b A_{bi} z^t_{b\to i};(1+\hat{\tau}^t)\Big). \tag{72}$$

Remarkably, knowledge of the prior distribution is asymptotically equivalent to knowledge of the functions F_i and G_i.

6.2.3 From message passing to AMP

By applying Lemma 3.4 we obtain the following algorithm (in matrix notation)

$$\begin{aligned} x^t &= \mathsf{F}(x^t + A^* z^t;\lambda+\gamma^t), && (73)\\ z^{t+1} &= y - Ax^t + \frac{1}{\delta} z^t \langle \mathsf{F}'(x^{t-1} + A^* z^{t-1})\rangle\,. && (74)\end{aligned}$$

Here, if $x\in\mathbb{R}^N$, $\mathsf{F}(x;b)\in\mathbb{R}^N$ is the vector $\mathsf{F}(x;b) = (\mathsf{F}_1(x_i;b),\mathsf{F}_2(x_2;b),\dots,\mathsf{F}_N(x_N;b))$. Analogously $\mathsf{F}'(x) = (\mathsf{F}'_1(x_i;b),\mathsf{F}'_2(x_2;b),\dots,\mathsf{F}'_N(x_N;b))$ (derivative being taken with respect to the first argument). Finally, the threshold level is computed iteratively as follows

$$\gamma^{t+1} = \frac{1}{\delta}\langle \mathsf{G}(Az^t + x^t;\gamma^t+\lambda)\rangle\,. \tag{75}$$

7 Comments and discussion

We presented a step-by-step approach for constructing message passing algorithms. This approach has several advantages:

1. The approach provided here is very general and can be applied to many other settings. The AMP algorithms in general may be slightly more complicated than the AMP.A and AMP.0 was demonstrated in section 6, but they are much simpler than the actual message passing algorithms on the complete graph.

2. The final approximate message passing algorithm does not have any free parameter. This may come at the cost of more complicated algorithm. The complications may show themselves specially in the analysis as was demonstrated in 5.

3. The state evolution framework provides a very simple approach to predict the asymptotic performance of the resulting algorithms.

There are a few open questions that are yet to be answered.

1. The state evolution has been proved to accurately predict the performance of AMP algorithm when the measurement matrix is i.i.d Gaussian. However simulation results show the correctness of state evolution on a wider range of matrix ensembles.

2. Our main concern in the derivation has been the single step performance of the message passing algorithm and not the whole algorithm. Therefore it is conceivable that the errors accumulate and the algorithm does not perform as well as the actual message passing. The simulation results again confirm that this phenomena does not happen but this has not been addressed theoretically.

A Berry-Esseen Central Limit Theorem

In this section we present a proof of the Berry-Esseen central limit theorem which is used in section 4. This proof is mainly due to Charles Stein [35]. Although more widely known approaches such as Lindeberg swapping trick [3] can also be used for proving similar results, we use Stein's method that gives stronger bound.

Theorem A.1. *Let $s_1, s_2, \ldots, s_n$ be independent zero mean random variables. Suppose $\mathbb{E}(s_i^2) = 1$ and $\mathbb{E}(|s_i|^3) \leq C$ where C is independent of i. For any differentiable bounded function $\phi(x)$ with bounded first derivative we have,*

$$\mathbb{E}(\phi(\frac{s_1 + s_2 + \ldots + s_n}{\sqrt{n}})) = \mathbb{E}(\phi(G)) + O(\frac{C}{\sqrt{n}}(1 + \sup |\phi'(x)|)),$$

where $G \sim \mathcal{N}(0, 1)$.

Proof. Let $Z_n = \frac{s_1+s_2+\ldots+s_n}{\sqrt{n}}$. Following Stein's method, for a given function $\phi(x)$ we define its Stein transform as,

$$T_\phi(x) = \mathrm{e}^{\frac{\mathrm{x}^2}{2}} \int_{-\infty}^{\mathrm{x}} \mathrm{e}^{-\frac{\mathrm{y}^2}{2}} (\phi(\mathrm{y}) - \mathbb{E}\phi(\mathrm{G}))\mathrm{dy}.$$

This function is bounded and has bounded derivative and second order derivative and $\sup_x |T''_\phi(x)| \leq \sup_x |\phi'(x)|$. It is also not difficult to see that that $\mathbb{E}(T'_\phi(Z_n) - Z_n T_\phi(Z_n)) = E(\phi(Z_n) - \phi(G))$. Define $T_i = Z_n - \frac{s_i}{\sqrt{n}}$.

δ	0.001	0.05	0.10	0.15	0.20	0.25	0.30	0.35	0.40	0.45	0.50	0.55	0.60
θ_{MM}	3.312	2.02	1.73	1.55	1.41	1.29	1.19	1.10	1.02	0.95	0.88	0.81	0.74

Table 1: Maxmin optimal value of $\theta_M M$ for several values of δ.

$$\mathbb{E} Z_n T_\phi(Z_n) = \sum_i s_i T_\phi(Z_n) = \mathbb{E}\frac{1}{\sqrt{n}}\sum_i s_i T_\phi(T_i) + \frac{s_i^2}{\sqrt{n}} T'_\phi(T_i + t\frac{s_i}{\sqrt{n}}) = \frac{1}{n}\mathbb{E} s_i^2 T'_\phi(T_i + t\frac{s_i}{\sqrt{n}}).$$

We now try to bound $E(f'(Z_n) - Z_n f(Z_n))$.

$$\begin{aligned}
\mathbb{E} \quad & (T'_\phi(Z_n) - Z_n T_\phi(Z_n)) = \mathbb{E} T'_\phi(Z_n) + \frac{1}{n}\sum_i T'_\phi(T_i + \frac{s_i}{\sqrt{n}}) - s_i^2 T'_\phi(T_i + t\frac{s_i}{\sqrt{n}}) \\
= \quad & \mathbb{E}\frac{1}{n}\sum_i T'_\phi(T_i + \frac{s_i}{\sqrt{n}}) - T'_\phi(T_i) + f'(T_i) - s_i^2 T'_\phi(T_i + t\frac{s_i}{\sqrt{n}}) \\
= \quad & \mathbb{E}\frac{1}{n}\sum_i T'_\phi(T_i + \frac{s_i}{\sqrt{n}}) - T'_\phi(T_i) + s_i^2 T'_\phi(T_i) - s_i^2 T'_\phi(T_i + t\frac{s_i}{\sqrt{n}}) \\
\leq \quad & \frac{1}{\sqrt{n}} E(|s_i| \sup_x |T''_\phi(x)|) + E(|s_i|^3) \sup_x |T''_\phi(x)|) \\
\leq \quad & \frac{4}{\sqrt{n}} \sup |\phi'(x)|.
\end{aligned}$$

□

B Table for $\theta_{MM}(\delta)$

In section 5 we considered the following algorithm,

$$\begin{aligned}
x^{t+1} &= \eta(x^t + A^* z^t; \theta\sigma^t), \\
z^t &= y - Ax^t + \frac{1}{\delta} z^{t-1}\langle \eta'(x^{t-1} + A^* z^{t-1}; \theta\sigma^{t-1})\rangle,
\end{aligned} \tag{76}$$

and mentioned that θ is a free parameter. In [13] the optimal value of θ that achieves the maximum phase transition was proved to be equal to

$$\theta_{MM}(\delta) = \arg\max \frac{1 - 2/\delta[(1+z^2)\Phi(-z) - z\phi(z)]}{1 + z^2 - 2[(1+z^2)\Phi(-z) - z\phi(z)]}.$$

Table 1 shows the optimal value of θ for several values of δ.

References

[1] M. Bayati and Andrea Montanri. The dynamics of message passing on dense graphs, with applications to compressed sensing. *IEEE Transactions on Information theory*, 2010. submitted.

[2] A. Beck and M. Teboulle. A fast iterative shrinkage thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences*, 2(1):183–202, 2009.

[3] P. Billingsley. *Probability and measure.* John Willey and sons, 3 edition, 1995.

[4] J. Bioucas-Dias and M. Figueiredo. A new twist: Two-step iterative shrinkage/thresholding algorithms for image restoration. *IEEE transactions on image processing*, 16:2992–3004, 2007.

[5] T. Blumensath and M. E. Davies. Iterative thresholding for sparse approximations. *Journal of Fourier Analysis and Applications, special issue on sparsity*, 14(5):629–654, 2008.

[6] T. Blumensath and M. E. Davies. How to use the iterative hard thresholding algorithm. *Proc. SPARS*, 2009.

[7] S.S. Chen, D.L. Donoho, and M.A. Saunders. Atomic decomposition by basis pursuit. *SIAM Journal on Scientific Computing*, 20:33–61, 1998.

[8] P. L. Combettes and V. R. Wajs. Signal recovery by proximal forward-backward splitting. *Multiscale Modeling and Simulation*, 4(4):1168–1200, 2005.

[9] D. Guo D. Baron and S. Shamai. A single-letter characterization of optimal noisy compressed sensing. *Proc. of the 47th Annual Allerton Conference*, 2009.

[10] I. Daubechies, M. Defrise, and C. De Mol. An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Communications on Pure and Applied Mathematics*, 75:1412–1457, 2004.

[11] D. L. Donoho. High-dimensional data analysis: The curses and blessings of dimensionality. *Amer. Math. Soc. Lecture: "Math challenges of the 21st century"*, 2000.

[12] D. L. Donoho, A. Maleki, and A. Montanari. Noise sensitivity phase transition. *IEEE Transactions on Information Theory*, 2010. submitted.

[13] D. L. Donoho, Arian Maleki, and Andrea Montanari. Message passing algorithms for compressed sensing. *Proceedings of National Academy of Sciences*, 106(45):18914–18919, 2009.

[14] D. L. Donoho and J. Tanner. Neighborliness of randomly-projected simplices in high dimensions. *Proceedings of the National Academy of Sciences*, 102(27):9452–9457, 2005.

[15] D. L. Donoho and J. Tanner. Counting faces of randomly projected polytopes when the projection radically lowers dimension. *Journal of American Mathematical Society*, 22:1–53, 2009.

[16] M. Elad, B. Matalon, J. Shtok, and M. Zibulevsky. A wide-angle view at iterated shrinkage algorithms. *Proc. SPIE (Wavelet XII)*, August 2007.

[17] M. Elad, B. Matalon, and M. Zibulevsky. Image denosing with shrinkage and redundant representations. *Proceedings of the IEEE computer Society Conference on Computer Vision and Pattern Recognition*, 2006.

[18] M. Figueiredo, J. Bioucas-Dias, and R. Nowak. Majorization-minimization algorithms for wavelet-based image restoration. *IEEE Transactions on Image Processing*, 16(12):2980–2991, 2007.

[19] M. Figueiredo and R. Nowak. An em algorithm for wavelet-based image restoration. *IEEE Transactions on Image Processing*, 12(8):906–916, 2003.

[20] M. Figueiredo, R. Nowak, and S. Wright. Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems. *IEEE Journal of Selected Topics of Signal Processing*, 1(4):586–598, 2007.

[21] E. Hale, W. Yin, and Y. Zhang. Fixed point continuation method for ℓ_1 minimization with application to compressed sensing. *Rice University Technial Report TR07-07*, 2007.

[22] Y. Kabashima, T. Wadayama, and T. Tanaka. A typical reconstruction limit for compressed sensing based on lp-norm minimization. *Journal of Statistical Mechanics*, 2009.

[23] A. Maleki and D. L. Donoho. Optimally tuned iterative thresholding algorithm for compressed sensing. *IEEE journal on selected areas in signal processing*, April 2010.

[24] A. Maleki and A. Montanari. Analysis of approximate message passing algorithm. *44th Annual Conforence on Information Sciences and Systems*, 2010.

[25] Arian Maleki. Approximate message passing algorithm for compressed sensing. *Stanford University PhD Thesis*, 2010.

[26] M. Mézard and A. Montanari. *Information, physics, computation: Probabilistic approaches*. ambridge University Press, 2008.

[27] Yurii Nestrov. Gradient methods for minimizing composite objective function. *CORE Report*, 2007.

[28] S. Rankan, A. K. Fletcher, and V. K. Goyal. Asymptotic analysis of map estimation via the replica method and applications to compressed sensing. *submitted to IEEE Transactions on Information Theory*, 2010.

[29] T.J. Richardson and R. Urbanke. *Modern Coding Theory*. Cambridge University Press.

[30] J. Bobin S. Becker and E. Candès. Nesta: a fast and accurate first-order method for sparse recovery. 2010. submitted for publication.

[31] M. Figueiredo S. Wright, R. Nowak. Sparse reconstruction by separable approximation. *IEEE Transactions on Signal Processing*, 57(7):2479–2493, 2009.

[32] S. Sardy, A. G. Bruce, and P. Tseng. Block coordinate relaxation methods for nonparametric wavelet denoising. *Journal of Computational and Graphical Statistics*, 9:361–379, 2000.

[33] S. Sarvotham, D. Baron, and R. Baraniuk. Compressed sensing reconstruction via belief propagation. Preprint, 2006.

[34] J. L. Starck, M. Elad, and D. L. Donoho. Redundant multiscale transforms and their application for morphological component analysis. *Journal of Advances in Imaging and Electron Physics*, 132:287–348, 2004.

[35] C. Stein. *Approximate Computation of Expectations*, volume 7. Institute of Mathematical Statistics, 1986.

[36] E. van den Berg and M. P. Friedlander. Probing the pareto frontier for basis pursuit solutions. *SIAM Journal on Scientific Computing*, 31(2):890–912, 2008.

[37] W. Yin, S. Osher, D. Goldfarb, and J. Darbon. Bregman iterative algorithms for ℓ_1-minimization with applications to compressed sensing. *SIAM Journal on Imaging Sciences*, 1(1):143–168, 2008.