Local Convergence of an AMP Variant to the LASSO Solution in Finite Dimensions

Yanting Ma,[1] Min Kang,[2] Jack W. Silverstein,[2] and Dror Baron[3]

***Abstract*—A common sparse linear regression formulation is the ℓ_1 regularized least squares, which is also known as least absolute shrinkage and selection operator (LASSO). Approximate message passing (AMP) has been proved to asymptotically achieve the LASSO solution when the regression matrix has independent and identically distributed (i.i.d.) Gaussian entries in the sense that the averaged per-coordinate ℓ_2 distance between the AMP iterates and the LASSO solution vanishes as the signal dimension goes to infinity before the iteration number. However, in finite dimensional settings, characterization of AMP iterates in the limit of large iteration number has not been established. In this work, we propose an AMP variant by including a parameter that depends on the largest singular value of the regression matrix. The proposed algorithm can also be considered as a primal dual hybrid gradient algorithm with adaptive stepsizes. We show that whenever the AMP variant converges, it converges to the LASSO solution for arbitrary finite dimensional regression matrices. Moreover, we show that the AMP variant is locally stable around the LASSO solution under the condition that the LASSO solution is unique and that the regression matrix is drawn from a continuous distribution. Our local stability result implies that in the special case where the regression matrix is large and has i.i.d. random entries, the original AMP, which is a special case of the proposed AMP variant, is locally stable around the LASSO solution.**

I. Introduction

Least absolute shrinkage and selection operator (LASSO) is a common formulation for sparse linear regression, which is defined as the optimization problem:

$$\mathbf{x}^* \in \arg\min_{\mathbf{x}\in\mathbb{R}^N}\{F(\mathbf{x}) := \frac{1}{2}\|\mathbf{y}-\mathbf{A}\mathbf{x}\|_2^2 + \gamma\|\mathbf{x}\|_1\}, \quad (1)$$

where $\mathbf{A} \in \mathbb{R}^{n\times N}$ is the regression matrix, $\mathbf{y} \in \mathbb{R}^n$ is the data vector, $\gamma > 0$ is the regularization parameter, and $\|\cdot\|_p$, for $p = 1, 2$, denotes the ℓ_p norm. Applications of LASSO include model selection and image processing, since natural images are usually sparse in some transform domain. While numerous standard convex optimization algorithms such as the class of proximal gradient methods [1]–[3], alternating direction method of multipliers (ADMM) [4], and primal dual hybrid gradient (PDHG) [5]–[7] can be used to solve (1), it is also of both theoretical and practical interest to study approximate message passing (AMP) for solving (1), since AMP was initially introduced by Donoho et al. [8] as a LASSO solver and it usually enjoys fast empirical convergence when it converges.

Existing theoretical convergence analysis of standard optimization algorithms and that of AMP are considered in different problem settings. Specifically, the quantity of interest to optimization algorithms is usually $\lim_{t\to\infty}\|\mathbf{x}^t-\mathbf{x}^*\|_2$, where $\mathbf{x}^t$ is the estimate at the t^{th} iteration of an iterative algorithm, for any fixed and finite n and N. It is usually assumed in the AMP framework that the data vector $\mathbf{y}$ is generated according to a linear system, $\mathbf{y} = \mathbf{A}\mathbf{x}_0 + \mathbf{w}$, with some underlying ground-truth $\mathbf{x}_0$ and some noise $\mathbf{w}$. Under these assumptions, the analysis of AMP shows that when $\mathbf{A}$ has independent and identically distributed (i.i.d.) Gaussian entries, the quantity $\lim_{N\to\infty}\frac{1}{N}\|\mathbf{x}^t-\mathbf{x}_0\|_2$ with $\frac{n}{N} \to c \in (0,\infty)$ converges to a deterministic number predicted by a scalar recursion referred to as state evolution with probability one [9]; this is later extended to a large deviation result [10]. For the LASSO problem, Bayati and Montanari [11] have proven the convergence of AMP iterates to the LASSO solution $\mathbf{x}^*$ in the sense that $\lim_{t\to\infty}\lim_{N\to\infty}\frac{1}{N}\|\mathbf{x}^t-\mathbf{x}^*\|_2^2 = 0$ with probability one, which has also been extended to a large deviation result in recent work [12]. However, this large deviation result only holds for $t = O\left(\frac{\log N}{\log\log N}\right)$.[1] Therefore, the convergence of AMP for finite N as $t \to \infty$ is still unknown. In fact, using matrices with i.i.d. Gaussian entries and following the calibration method proposed in [11] for choosing the threshold of the soft-thresholding function at each AMP iteration, we performed 2000 trials of Monte Carlo simulations with $N = 2000, n = 1000$, and AMP never converged to the LASSO solution in terms of ℓ_2 error.

The connection between AMP and standard convex optimization algorithms has enabled the design of AMP variants that have convergence guarantees for more practical settings such as the ones with non-Gaussian finite dimensional matrices. Most such results have been developed in a more general algorithmic framework known as generalized AMP (GAMP) [13]. In the context of solving optimization problems, GAMP considers objective functions of the form $\sum_{i=1}^n g([\mathbf{A}\mathbf{x}]_i) + \sum_{i=1}^N f(x_i)$. Consider now that $\mathbf{A}$ is arbitrary and finite dimensional. When both g and f are quadratic

[1] Y. Ma is with Mitsubishi Electric Research Laboratories, Cambridge, MA. Email: yma@merl.com.

[2] M. Kang and J. W. Silverstein are with the Department of Mathematics, North Carolina State University, Raleigh, NC. Emails: {mkang2, jack}@ncsu.edu.

[3] D. Baron is with the Department of Electrical and Computer Engineering, North Carolina State University, Raleigh, NC. Email: barondror@ncsu.edu.

D. Baron was supported in part by NSF EECS 1611112.

This work was completed while Y. Ma was with North Carolina State University.

[1] The big O notation $O(\cdot)$ means that there exists an $N_0 \in \mathbb{N}$ and a positive real number B such that $t(N) \le B\frac{\log N}{\log\log N}$ for all $N \ge N_0$.

functions, the damped GAMP [14], which defines the current iterate as a convex combination of the current estimate and the iterate from the previous iteration, has global convergence guarantees. When g and f are strictly convex and twice continuously differentiable, assuming that the derivatives of the nonlinear functions used in each GAMP iteration are bounded within the open interval $(0,1)$, the damped GAMP with fixed stepsize [14] is proved to be locally stable around the equilibrium point, and the ADMM-GAMP [15], which combines an ADMM inner loop within each GAMP iteration, is guaranteed to achieve global convergence.

The interpretation of AMP as a PDHG algorithm was first mentioned by Rangan et al. [14], which has inspired the current work. The differences between our work and the local stability analysis in [14] are as follows:

1) The objective function of LASSO is non-differentiable, hence is not covered in [14].
2) Instead of damping the iterates while keeping the stepsizes fixed as in [14], we do not make changes to the updates of the iterates but design a stepsize updating schedule based on AMP.
3) The result in [14] holds only when the stepsize is fixed over all iterations, which loses the main advantage of AMP over standard optimization algorithms for fast convergence, whereas our result allows keeping the structure of the stepsize updating schedule of AMP. As we will see in Section IV, numerical results show that the number of iterations required for our algorithm to converge to the LASSO solution is orders of magnitude smaller than widely used optimization algorithms.

The remainder of the paper is organized as follows. Section II introduces our proposed AMP variant. Section III provides local stability analysis of our algorithm around its equilibrium point, which is the LASSO solution. Section IV presents simulation results that compare our algorithm with AMP and other widely used optimization algorithms. Finally, Section V concludes the paper.

II. Proposed Algorithm

Let $g:\mathbb{R}^n\to\mathbb{R}$ and $f:\mathbb{R}^N\to\mathbb{R}$ be defined as

$$g(\mathbf{u}):=\frac{1}{2}\|\mathbf{y}-\mathbf{u}\|_2^2 \quad \text{and} \quad f(\mathbf{x}):=\gamma\|\mathbf{x}\|_1, \tag{2}$$

respectively. The idea of the class of PDHG algorithms is to write the minimization problem $\inf_{\mathbf{x}\in\mathbb{R}^N}\{g(\mathbf{A}\mathbf{x})+f(\mathbf{x})\}$ as a saddle-point problem using the fact that the function g defined above is convex, closed, and proper, thus $g=(g^*)^*$ [16], where g^* is the convex-conjugate of g defined as

$$g^*(\mathbf{s}):=\sup_{\mathbf{u}\in\mathbb{R}^n}\left\{\langle\mathbf{s},\mathbf{u}\rangle-g(\mathbf{u})\right\}. \tag{3}$$

Plugging $g(\mathbf{A}\mathbf{x})=(g^*)^*(\mathbf{A}\mathbf{x})=\sup_{\mathbf{s}\in\mathbb{R}^n}\{\langle\mathbf{A}\mathbf{x},\mathbf{s}\rangle-g^*(\mathbf{s})\}$ into the minimization problem, we obtain the saddle-point problem

$$\inf_{\mathbf{x}\in\mathbb{R}^N}\sup_{\mathbf{s}\in\mathbb{R}^n}F(\mathbf{s},\mathbf{x}), \tag{4}$$

where

$$F(\mathbf{s},\mathbf{x}):=\langle\mathbf{s},\mathbf{A}\mathbf{x}\rangle-g^*(\mathbf{s})+f(\mathbf{x}). \tag{5}$$

PDHG solves (4) by alternating between the estimation of $\mathbf{s}$ and $\mathbf{x}$ as $\mathbf{s}^{t+1}=\arg\max_{\mathbf{s}\in\mathbb{R}^n}\left\{F(\mathbf{s},\mathbf{x}^t)+\frac{1}{2\tau_s^t}\|\mathbf{s}-\mathbf{s}^t\|_2^2\right\}$ and $\mathbf{x}^{t+1}=\arg\min_{\mathbf{x}\in\mathbb{R}^N}\left\{F(\mathbf{s}^{t+1},\mathbf{x})+\frac{1}{2\tau_x^t}\|\mathbf{x}-\mathbf{x}^t\|_2^2\right\}$, respectively, which is equivalent to

$$\begin{aligned}\mathbf{s}^{t+1}&=\arg\min_{\mathbf{s}\in\mathbb{R}^n}\left\{g^*(\mathbf{s})+\frac{1}{2\tau_s^t}\|\mathbf{s}-(\mathbf{s}^t+\tau_s^t\mathbf{A}\mathbf{x}^t)\|_2^2\right\},\\ \mathbf{x}^{t+1}&=\arg\min_{\mathbf{x}\in\mathbb{R}^N}\left\{f(\mathbf{x})+\frac{1}{2\tau_x^t}\|\mathbf{x}-(\mathbf{x}^t-\tau_x^t\mathbf{A}^T\mathbf{s}^{t+1})\|_2^2\right\}.\end{aligned} \tag{6}$$

In the above, the stepsizes τ_x^t and τ_s^t can stay constant for all iterations or be updated at every iteration. One feature of PDHG algorithms is that each equilibrium point of the algorithm is a saddle-point of (4). This can be explained as follows. Let $(\widehat{\mathbf{x}},\widehat{\mathbf{s}},\widehat{\tau}_x,\widehat{\tau}_s)$ be an equilibrium point of the algorithm (6), then

$$\begin{aligned}0&\in\partial_{\mathbf{s}}\left(g^*(\mathbf{s})+\frac{1}{2\widehat{\tau}_s}\|\mathbf{s}-(\widehat{\mathbf{s}}+\widehat{\tau}_s\mathbf{A}\widehat{\mathbf{x}})\|_2^2\right)\Big|_{\mathbf{s}=\widehat{\mathbf{s}}},\\ 0&\in\partial_{\mathbf{x}}\left(f(\mathbf{x})+\frac{1}{2\widehat{\tau}_x}\|\mathbf{x}-(\widehat{\mathbf{x}}-\widehat{\tau}_x\mathbf{A}^T\widehat{\mathbf{s}})\|_2^2\right)\Big|_{\mathbf{x}=\widehat{\mathbf{x}}},\end{aligned}$$

where $\partial_{\mathbf{u}}$ denotes sub-differential with respect to $\mathbf{u}$. The above implies that $\mathbf{A}\widehat{\mathbf{x}}\in\partial g^*(\widehat{\mathbf{s}})$ and $-\mathbf{A}^T\widehat{\mathbf{s}}\in\partial f(\widehat{\mathbf{x}})$, which is the necessary and sufficient condition for $(\widehat{\mathbf{x}},\widehat{\mathbf{s}})$ to be a saddle-point of (4).

The choice of stepsizes is crucial for the convergence of an optimization algorithm. AMP can be interpreted as a special case of PDHG with an adaptive stepsize updating schedule [14]. Specifically, let

$$\tau_s^t=\frac{1}{\tau_x^t-1},\quad \tau_x^{t+1}=1+\frac{d^t}{c}\tau_x^t, \tag{7}$$

where $d^t=\|\mathbf{x}^{t+1}\|_0/N$ with $\|\mathbf{x}^{t+1}\|_0$ (the ℓ_0 quasi-norm) denoting the number of nonzero coordinates of $\mathbf{x}^{t+1}$. By (2) and (3), we have $g^*(\mathbf{s})=\langle\mathbf{y},\mathbf{s}\rangle+\frac{1}{2}\|\mathbf{s}\|_2^2$. For easy comparison, we use the same notation for the soft-thresholding function (proximal operator for the ℓ_1-norm) as in [11]. For any $\theta>0$, $\mathbf{u}\in\mathbb{R}^N$, define

$$\eta(\mathbf{u};\theta):=\arg\min_{\mathbf{x}\in\mathbb{R}^N}\|\mathbf{x}\|_1+\frac{1}{2\theta}\|\mathbf{x}-\mathbf{u}\|_2^2. \tag{8}$$

Then (6) can be written as

$$\begin{aligned}\mathbf{s}^{t+1}&=\frac{1}{\tau_x^t}\left(\mathbf{A}\mathbf{x}^t-\mathbf{y}\right)+\left(1-\frac{1}{\tau_x^t}\right)\mathbf{s}^t,\\ \mathbf{x}^{t+1}&=\eta\left(\mathbf{x}^t-\tau_x^t\mathbf{A}^T\mathbf{s}^{t+1};\gamma\tau_x^t\right).\end{aligned} \tag{9}$$

Let $\mathbf{z}^t:=-\tau_x^t\mathbf{s}^{t+1}$ for all $t\geq 1$ and notice from (7) that $(\tau_x^t-1)/\tau_x^{t-1}=d^{t-1}/c$. Then we can see that (9) matches the AMP algorithm (see [8] and [11]), but with a different choice for the threshold of the soft-thresholding function. We emphasize that the choice of the threshold in [11] does not guarantee that AMP will converge to the LASSO solution

for finite dimensional problems, whereas the choice in (9) guarantees that whenever (9) converges, it converges to the LASSO solution for arbitrary finite dimensional $\mathbf{A}$.

We notice that in many optimization algorithms, the stepsize usually depends on the largest singular value of $\mathbf{A}$, whereas the algorithm defined in (6) with τ_s^t and τ_x^t being updated according to (7), which is equivalent to AMP (9), does not depend on the largest singular value of $\mathbf{A}$. Therefore, in order to have an algorithm that is more robust than AMP with arbitrary finite dimensional $\mathbf{A}$ while retaining the fast convergence feature of AMP, we introduce a parameter e to (7) that depends on the largest singular value of $\mathbf{A}$ and still keep the general structure of the updating schedule in (7). Specifically, choosing $0 < e < \min\{1, 4/(\sigma_{\max}^2(\mathbf{A}) + 2)\}$, where $\sigma_{\max}(\mathbf{A})$ is the largest singular value of $\mathbf{A}$, we modify (7) as

$$\tau_s^t = \frac{e}{\tau_x^t - e}, \quad \tau_x^{t+1} = 1 + \frac{d^t}{c}\tau_x^t. \tag{10}$$

Note that such a choice of e ensures local stability of our proposed algorithm; details about the local stability analysis will be discussed in Section III. Our proposed AMP variant is (6) with the stepsize updating schedule defined in (10). Similar to the derivation of (9) from (6) and (7), we can write the proposed AMP variant as in Algorithm 1.

Algorithm 1 Proposed AMP Variant

Input: $\mathbf{A}$, $\mathbf{y}$, $0 < e < \min\{1, 4/(\sigma_{\max}^2(\mathbf{A}) + 2)\}$, $t_{\max}$
Initialization: $\mathbf{x}^0$, $\mathbf{s}^0$, τ_x^0
for $0 \le t \le t_{\max}$ **do**

$$\begin{aligned} \mathbf{s}^{t+1} &= \frac{e}{\tau_x^t}\left(\mathbf{A}\mathbf{x}^t - \mathbf{y}\right) + \left(1 - \frac{e}{\tau_x^t}\right)\mathbf{s}^t \\ \mathbf{x}^{t+1} &= \eta\left(\mathbf{x}^t - \tau_x^t \mathbf{A}^T \mathbf{s}^{t+1}; \gamma\tau_x^t\right) \\ \tau_x^{t+1} &= 1 + \frac{d^t}{c}\tau_x^t \quad \text{with } d^t = \|\mathbf{x}^{t+1}\|_0/N \end{aligned} \tag{11}$$

end for
Output: $\mathbf{x}^{t_{\max}}$

III. Local Stability Analysis

In this section, we study the local stability of Algorithm 1 around its equilibrium point, which is the LASSO solution. We first discuss conditions under which our analysis is valid, and then show that Algorithm 1 is locally stable under these conditions.

A. Assumptions

When Null($\mathbf{A}$), the null space of $\mathbf{A}$, contains nonzero components, the objective function is not strictly convex in $\mathbf{x}$, hence there may be multiple solutions. Conditions for the uniqueness of the LASSO solution have been studied by Tibshirani [17], which states that a sufficient [17, Lemma 2] and necessary [17, Lemma 6] condition for (1) to admit a unique solution is that Null($\mathsf{C}_K(\mathbf{A})$) $= \{\mathbf{0}\}$, where $K := \{i \in \{1, \dots, N\} | x_i^* \neq 0\}$ with x_i^* the i^{th} coordinate of $\mathbf{x}^*$, and $\mathsf{C}_K(\mathbf{A})$ is the submatrix of $\mathbf{A}$ formed by deleting the i^{th} column of $\mathbf{A}$ for all $i \notin K$. Notice that the necessary condition implies that when the LASSO solution is unique, we have that $|K| \le \min\{n, N\}$; this is a condition that we need to prove our result. A more explicit sufficient condition for uniqueness in the almost sure sense is also provided in [17, Lemma 4]: if entries of $\mathbf{A}$ are drawn from a continuous probability distribution on $\mathbb{R}^{n\times N}$, then the LASSO solution is unique with probability one regardless of the dimension of $\mathbf{A}$.

Note that with $\eta(\cdot)$ being the soft-thresholding function (8), the definition of d^t can be written as

$$d^t = \left|\{i \in \{1, \dots, N\} : |[\mathbf{x}^t - \tau_x^t \mathbf{A}^T \mathbf{s}^{t+1}]_i| > \gamma\tau_x^t\}\right|,$$

where we can further replace $\mathbf{s}^{t+1}$ by $\frac{e}{\tau_x^t}\left(\mathbf{A}\mathbf{x}^t - \mathbf{y}\right) + \left(1 - \frac{e}{\tau_x^t}\right)\mathbf{s}^t$, so that d^t only depends on the iterates at the t^{th} iteration. Similarly, the update of $\mathbf{x}^{t+1}$ in (11) can also be written as a function of iterates at the t^{th} iteration only. Therefore, letting $\mathbf{v}^t \in \mathbb{R}^{n+N+1}$ be defined as $\mathbf{v}^t := [\mathbf{s}^t; \mathbf{x}^t; \tau_x^t]$ for all $t \ge 0$, we have that (11) defines a nonlinear operator $G : \mathbb{R}^{n+N+1} \to \mathbb{R}^{n+N+1}$ such that $\mathbf{v}^{t+1} = G(\mathbf{v}^t)$. Note that G is differentiable at $[\mathbf{s}^t; \mathbf{x}^t; \tau_x^t]$ except when there exists an $i \in \{1, \dots, N\}$, such that

$$[\mathbf{x}^t - \mathbf{A}^T\left(e(\mathbf{A}\mathbf{x}^t - \mathbf{y}) + (\tau_x^t - e)\mathbf{s}^t\right)]_i = \pm\gamma\tau_x^t,$$

which has probability zero if $\mathbf{A}$ obeys a continuous distribution.

To summarize, we make the following two assumptions on components in (1):

1) The matrix $\mathbf{A}$ is drawn from a continuous probability distribution on $\mathbb{R}^{n\times N}$.
2) The regularization parameter $\gamma > 0$.

B. Stability around the Equilibrium Point

Having clarified our assumptions, we now prove our main result, which is the local stability guarantee of Algorithm 1 as stated in the following proposition.

Proposition 1. *Consider the LASSO problem defined in (1) and suppose that the conditions in Section III-A are satisfied. Then Algorithm 1 is stable around its equilibrium point with probability one.*

Proof. Suppose that G is differentiable around the equilibrium point $\widehat{\mathbf{v}}$. Then the local stability of G around $\widehat{\mathbf{v}}$ is determined by the largest eigenvalue (in modulus) of the Jacobian matrix $\mathbf{J}$ of G evaluated at $\widehat{\mathbf{v}}$. The expression for $\mathbf{J}$ is

$$\begin{bmatrix} (1 - e/\widehat{\tau}_x)\,\mathbf{I}_n & (e/\widehat{\tau}_x)\mathbf{A} & \mathbf{0}_{n\times 1} \\ (e - \widehat{\tau}_x)\,\widehat{\mathbf{D}}\mathbf{A}^T & \widehat{\mathbf{D}}\left(\mathbf{I}_N - e\mathbf{A}^T\mathbf{A}\right) & -\widehat{\mathbf{D}}\mathbf{A}^T\widehat{\mathbf{s}} \\ \mathbf{0}_{1\times n} & \mathbf{0}_{1\times N} & \widehat{d}/c \end{bmatrix}, \tag{12}$$

where $\widehat{\mathbf{D}}$ is a diagonal matrix defined as $\widehat{D}_{ii} = \mathbb{I}\{[\widehat{\mathbf{x}}]_i \neq 0\}$ with $\mathbb{I}$ being the indicator function, $\mathbf{I}_n$ is the $n \times n$ identity matrix, and $\mathbf{0}_{n\times m}$ is an $n \times m$ matrix with all zero-valued coordinates. In what follows, we show that with an appropriate choice of e, the eigenvalue of $\mathbf{J}$ with the largest modulus is within the unit circle of the complex plane.

Let $\alpha = 1 - e/\widehat{\tau}_x$ and let $|\mathbf{A}|$ denote the determinant of a matrix $\mathbf{A}$. Then

$$\begin{aligned}|\mathbf{J}-\lambda\mathbf{I}| &\overset{(a)}{=} \left(\widehat{d}/c-\lambda\right)\begin{vmatrix}(\alpha-\lambda)\mathbf{I}_n & (e/\widehat{\tau}_x)\mathbf{A}\\ (e-\widehat{\tau}_x)\widehat{\mathbf{D}}\mathbf{A}^T & \widehat{\mathbf{D}}-\lambda\mathbf{I}_N - e\widehat{\mathbf{D}}\mathbf{A}^T\mathbf{A}\end{vmatrix}\\ &\overset{(b)}{=} \left(\widehat{d}/c-\lambda\right)\begin{vmatrix}(\alpha-\lambda)\mathbf{I}_n & (e/\widehat{\tau}_x)\mathbf{A}\\ \mathbf{0}_{N\times n} & \widehat{\mathbf{D}}-\lambda\mathbf{I}_N + \lambda e/(\alpha-\lambda)\widehat{\mathbf{D}}\mathbf{A}^T\mathbf{A}\end{vmatrix}\\ &= \left(\widehat{d}/c-\lambda\right)(\alpha-\lambda)^{n-N}\\ &\quad\cdot\left|(\alpha-\lambda)\widehat{\mathbf{D}}-(\alpha-\lambda)\lambda\mathbf{I}_N+\lambda e\widehat{\mathbf{D}}\mathbf{A}^T\mathbf{A}\right|, \end{aligned} \tag{13}$$

where step (a) follows by expanding the last row of $\mathbf{J}-\lambda\mathbf{I}$, and step (b) follows by subtracting $\frac{e-\widehat{\tau}_x}{\alpha-\lambda}\widehat{\mathbf{D}}\mathbf{A}^T$ times the first row from the second row and noticing that

$$\begin{aligned}e+\frac{e}{\widehat{\tau}_x}\frac{e-\widehat{\tau}_x}{\alpha-\lambda} &= e\left(1+\frac{1}{\alpha-\lambda}\left(\frac{e}{\widehat{\tau}_x}-1\right)\right)\\ &= e\left(1-\frac{\alpha}{\alpha-\lambda}\right) = \frac{-\lambda e}{\alpha-\lambda}.\end{aligned}$$

To calculate $\left|(\alpha-\lambda)\widehat{\mathbf{D}}-(\alpha-\lambda)\lambda\mathbf{I}_N+\lambda e\widehat{\mathbf{D}}\mathbf{A}^T\mathbf{A}\right|$, we first introduce some notation. For a matrix $\mathbf{B}\in\mathbb{R}^{N\times N}$ and index set $K\subset\{1,2,...,N\}$, let $[\mathbf{B}]_K\in\mathbb{R}^{|K|\times|K|}$ denote the submatrix of $\mathbf{B}$ obtained by eliminating the i^{th} row and i^{th} column of $\mathbf{B}$ for all $i\notin K$. Moreover, let $\mathsf{R}_K(\mathbf{B})$ (resp. $\mathsf{C}_K(\mathbf{B})$) denote the submatrix of $\mathbf{B}$ formed by deleting the i^{th} row (resp. column) of $\mathbf{B}$ for all $i\notin K$.

Now letting $\mathbf{B}=(\alpha-\lambda)\widehat{\mathbf{D}}-(\alpha-\lambda)\lambda\mathbf{I}_N+\lambda e\widehat{\mathbf{D}}\mathbf{A}^T\mathbf{A}$, we have

$$\mathsf{R}_{\{i\}}(\mathbf{B}) = \begin{cases}-\lambda(\alpha-\lambda)\mathbf{e}_i^T, & \text{if } i\in K^c\\ (1-\lambda)(\alpha-\lambda)\mathbf{e}_i^T + e\lambda\mathsf{R}_{\{i\}}(\mathbf{A}^T\mathbf{A}), & \text{if } i\in K\end{cases},$$

where all but the i^{th} coordinates of $\mathbf{e}_i\in\mathbb{R}^N$ are zero and the i^{th} coordinate is 1, and $K=\{i\in\{1,...,N\}|\widehat{D}_{ii}=1\}$. By expanding the i^{th} row of $\mathbf{B}$ for all $i\notin K$, we have

$$|\mathbf{B}| = (-\lambda(\alpha-\lambda))^{N(1-\widehat{d})}\left|(\alpha-\lambda)(1-\lambda)\mathbf{I}_{N\widehat{d}}+\lambda e[\mathbf{A}^T\mathbf{A}]_K\right|.$$

Plugging the above into (13), we have

$$\begin{aligned}|\mathbf{J}-\lambda\mathbf{I}| &= \left(\widehat{d}/c-\lambda\right)(\alpha-\lambda)^{n-N}(-\lambda(\alpha-\lambda))^{N(1-\widehat{d})}\\ &\quad\cdot\left|(\alpha-\lambda)(1-\lambda)\mathbf{I}_{N\widehat{d}}+\lambda e[\mathbf{A}^T\mathbf{A}]_K\right|\\ &= (-1)^{N(1-\widehat{d})}\left(\widehat{d}/c-\lambda\right)\lambda^{N(1-\widehat{d})}(\alpha-\lambda)^{n-N\widehat{d}}\\ &\quad\cdot\left|(\alpha-\lambda)(1-\lambda)\mathbf{I}_{N\widehat{d}}+\lambda e[\mathbf{A}^T\mathbf{A}]_K\right|. \end{aligned} \tag{14}$$

Let $\mathbf{H}=[\mathbf{A}^T\mathbf{A}]_K$, we now need to solve for λ in the following equation:

$$\left|(\alpha-\lambda)(1-\lambda)\mathbf{I}_{N\widehat{d}}+\lambda e\mathbf{H}\right| = 0. \tag{15}$$

First, we check that $\lambda=0$ is not a solution to (15). Plugging $\lambda=0$ into (15), we have

$$\left|(\alpha-\lambda)(1-\lambda)\mathbf{I}_{N\widehat{d}}+\lambda e\mathbf{H}\right| = |\alpha\mathbf{I}_{N\widehat{d}}| = (1-e/\widehat{\tau}_x)^{N\widehat{d}} > 0,$$

where the last inequality holds because $e\in(0,1]$ and $\widehat{\tau}_x>1$, hence $1-e/\widehat{\tau}_x>0$. Now that $\lambda\neq 0$, we divide both sides of (15) by $(e\lambda)^{N\widehat{d}}$:

$$\left|\frac{(\alpha-\lambda)(1-\lambda)}{\lambda e}\mathbf{I}_{N\widehat{d}}+\mathbf{H}\right| = 0. \tag{16}$$

Therefore, λ is a solution to (15) if and only if $-\frac{(\alpha-\lambda)(1-\lambda)}{\lambda e}$ is an eigenvalue of $\mathbf{H}$. Let $\mathrm{sp}(\mathbf{H})$ denote the spectrum (i.e., set of eigenvalues) of $\mathbf{H}$, and define

$$m := \min_{\lambda\in\mathrm{sp}(\mathbf{H})}\lambda, \qquad \text{and} \qquad M := \max_{\lambda\in\mathrm{sp}(\mathbf{H})}\lambda. \tag{17}$$

Note that $\mathbf{H}=[\mathbf{A}^T\mathbf{A}]_K=(\mathsf{C}_K(\mathbf{A}))^T\mathsf{C}_K(\mathbf{A})$. By condition of the uniqueness of LASSO solution, we have that $\mathsf{C}_K(\mathbf{A})$ is non-singular, hence $m>0$. Let $u=-\frac{(\alpha-\lambda)(1-\lambda)}{\lambda e}$, then

$$\lambda^2+(eu-\alpha-1)\lambda+\alpha = 0. \tag{18}$$

Let $b:=eu-\alpha-1$, and we solve (18) for λ. When $|b|<2\sqrt{\alpha}$, we have complex roots $\lambda=\frac{1}{2}(-b\pm i\sqrt{4\alpha-b^2})$, hence $|\lambda|=\frac{1}{2}(\sqrt{b^2+4\alpha-b^2})=\sqrt{\alpha}<1$. When $|b|\geq 2\sqrt{\alpha}$, we have real roots. Define

$$\begin{aligned}h_1(b) &:= \frac{1}{2}(-b+\sqrt{b^2-4\alpha}),\\ h_2(b) &:= \frac{1}{2}(-b-\sqrt{b^2-4\alpha}).\end{aligned} \tag{19}$$

Notice that

$$\begin{aligned}h_1'(b) &= \frac{1}{2}\left(-1+\frac{b}{\sqrt{b^2-4\alpha}}\right)\begin{cases}>0, & \text{if } b>2\sqrt{\alpha}\\ <0, & \text{if } b\leq -2\sqrt{\alpha}\end{cases},\\ h_2'(b) &= \frac{1}{2}\left(-1-\frac{b}{\sqrt{b^2-4\alpha}}\right)\begin{cases}<0, & \text{if } b\geq 2\sqrt{\alpha}\\ >0, & \text{if } b<-2\sqrt{\alpha}\end{cases}.\end{aligned} \tag{20}$$

Also notice that $h_1(b)\uparrow 0$ as $b\uparrow\infty$, and $h_2(b)\downarrow 0$ as $b\downarrow-\infty$. The graph of h_1 and h_2 as a function of b, respectively, is shown in Fig. 1. It can be seen that

$$\max(|h_1(b)|,|h_2(b)|) = \begin{cases}h_1(b), & \text{if } b\leq -2\sqrt{\alpha}\\ -h_2(b), & \text{if } b\geq 2\sqrt{\alpha}\end{cases}.$$

First consider $b\leq -2\sqrt{\alpha}$. Recall that $b=eu-\alpha-1>em-\alpha-1>-\alpha-1$, since $e>0$ and $m>0$. Notice from (19) and (20) that $h_1(-\alpha-1)=1$, $h_1(-2\sqrt{\alpha})=\sqrt{\alpha}<1$, and that $h_1(b)$ is monotone decreasing when $b\leq -2\sqrt{\alpha}$. Therefore, we have that $|h_1(b)|<1$, when $b\leq -2\sqrt{\alpha}$.

Next consider $b\geq 2\sqrt{\alpha}$. Notice that $h_2(1+\alpha)=-1$ and that $h_2(b)$ is monotone decreasing when $b\geq 2\sqrt{\alpha}$. Therefore, in order to have $|h_2(b)|<1$, we need $b=eu-\alpha-1<1+\alpha, \forall u\in\mathrm{sp}(\mathbf{H})$. This condition is satisfied if we let $e<\frac{2(1+\alpha)}{M}$. Recall that $\alpha=1-e/\widehat{\tau}_x$. By (11), we have $1/\widehat{\tau}_x=1-\widehat{d}/c$. Combining the analysis on h_1 and h_2 given above, it follows that the condition on the parameter e for all eigenvalues of $\mathbf{J}$ to be within the unit circle of the complex plane is

$$0<e<\min\left\{1,\frac{4}{M+2(1-\widehat{d}/c)}\right\}. \tag{21}$$

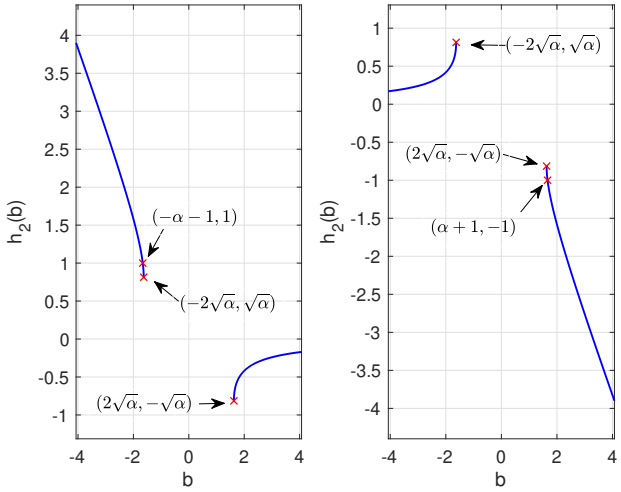


Fig. 1: Graph of h_1 and h_2 as a function of b, where a specific e is chosen for demonstration purposes.

The upper-bound for e in (21) is tight. However, it depends on the equilibrium point. It is desirable to have a condition on e that does not depend on knowledge about the equilibrium point, so that an appropriate value for e can be chosen before running the algorithm (11). Notice that $\mathbf{H} = [\mathbf{A}^T\mathbf{A}]_K$ is a principal submatrix of the symmetric matrix $\mathbf{A}^T\mathbf{A}$. The interlacing property of eigenvalues implies that $M \leq L := \max_{\lambda \in \text{sp}(\mathbf{A}^T\mathbf{A})} \lambda$. Moreover, the condition for the uniqueness of the LASSO solution implies that $\widehat{d}/c \in [0,1]$. Therefore, local stability is guaranteed if

$$0 < e \leq \min\{1, 4/(L+2)\}. \tag{22}$$

□

C. Random Matrices with i.i.d. Entries

In the special case where the matrix $\mathbf{A} \in \mathbb{R}^{n\times N}$ is the upper left corner of a doubly infinite array[2] of i.i.d. zero-mean random variables with finite fourth moment and is normalized such that the variance is $1/n$, the asymptotic largest singular value of $\mathsf{C}_K(\mathbf{A}) \in \mathbb{R}^{n\times N\widehat{d}}$, where $N\widehat{d} < n$, is $1+\sqrt{N\widehat{d}/n} = 1+\sqrt{\widehat{d}/c}$ with probability one [18]. It follows that the denominator of the upper-bound in (21) is

$$M+2(1-\widehat{d}/c) = 1+\widehat{d}/c+2\sqrt{\widehat{d}/c}+2-2\widehat{d}/c = 3+2\sqrt{\widehat{d}/c}-\widehat{d}/c.$$

Let $x = \widehat{d}/c$, hence $x \in [0,1]$. Define $f(x) = 2\sqrt{x} - x$ and notice that f is monotone increasing on $[0,1]$. Therefore, $f(x) < f(1) = 1$, which implies that $M + 2(1 - \widehat{d}/c) \leq 4$. That is, local stability for large zero-mean random matrices with variance $1/n$ is guaranteed by setting $e = 1$, which, as mentioned before, makes Algorithm 1 coincide with the original AMP (9), as seen in [8] and [11].

[2]That is, we have an array $\{X_{ij}\}, i = 1,2,\ldots; j = 1,2,\ldots$ and $\mathbf{A} = (X_{ij}), i = 1,2,\ldots,n; j = 1,2,\ldots,N$.

IV. Numerical Demonstration

To demonstrate the efficiency of the proposed AMP variant, we compare it with the original AMP that uses the calibration method proposed in [11], a PDHG algorithm with a fixed stepsize that guarantees convergence (see [7]), and a popular convex optimization algorithm, fast iterative shrinkage and thresholding algorithm (FISTA) [3]. Because our proposed algorithm is inspired by AMP and depends on a parameter e, we will refer to it as "eAMP" and we choose e according to (22). In addition, we also include results for eAMP with $e = 1$, which we recall is of the same form as the standard AMP but the threshold for the thresholding function at each iteration is different from that in [11].

In all the simulations, the problem dimension is set to be $N = 2000$, $n = 1000$. The data vector $\mathbf{y}$ is obtained by $\mathbf{y} = \mathbf{A}\mathbf{x}_0 + \mathbf{w}$, where entries of $\mathbf{w}$ are independent realizations of a Gaussian distribution with mean zero and variance σ_w^2 and entries of $\mathbf{x}_0$ are independent realizations of a Bernoulli(0.1)-Uniform[-1,1] distribution (i.e., $X_0 = BU$ with $B \sim$ Bernoulli(0.1) and $U \sim$ Uniform$[-1,1]$). The value of σ_w^2 is chosen such that $10\log_{10}\left(\frac{\|\mathbf{A}\mathbf{x}_0\|_2^2}{n\sigma_w^2}\right) = 25$ (i.e., the signal-to-noise ratio is 25dB). All tested algorithms are initialized with an all-zero vector. Since FISTA and PDHG have theoretical convergence guarantees, we present their results only for comparison of empirical convergence speed, hence we sometimes stop these two algorithms early when the convergence speed comparison is clear.

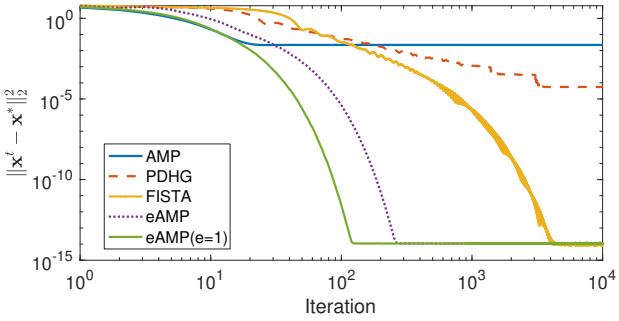
For the first set of simulations, we use matrices $\mathbf{A}$ whose entries are independent realizations of a zero-mean Gaussian distribution, which is the case studied in [11] in the limit as $N, n \to \infty$. The simulation results are shown in Fig. 2a. We notice that while AMP seems to have converged, it does not converge to the LASSO solution $\mathbf{x}^*$, whereas eAMP with both choices of e has converged to the LASSO solution. Moreover, the empirical convergence speed (in terms of number of iterations) of eAMP is much faster than that of FISTA or PDHG.

For the second set of simulations, we use matrices $\mathbf{A}$ whose rows are independent realizations of a zero-mean multivariate Gaussian distribution, where diagonal entries of the covariance matrix have value $1/n$ and off-diagonal entries have value $0.01/n$. The simulation results are shown in Fig. 2b. In this case, AMP and eAMP with the inappropriate choice of $e = 1$ have diverged, whereas eAMP with e chosen according to (22) has converged to the LASSO solution and requires far fewer iterations than FISTA and PDHG.

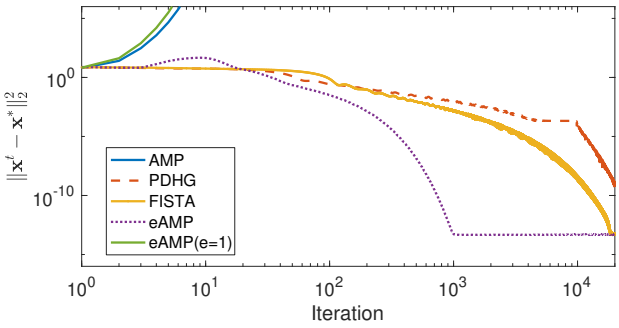
While our analysis in Section III only guarantees local stability for eAMP, the encouraging simulation results suggest that a global convergence result might be possible; we leave the global convergence analysis for future work.

V. Conclusion

In this paper, we proposed an AMP variant (Algorithm 1) for solving the LASSO problem (1). Unlike the work in [11] that analyzes the limiting behavior of AMP iterates as



(a) Entries of the matrix $\mathbf{A}$ are drawn independently from a zero-mean Gaussian distribution.



(b) Rows of the matrix $\mathbf{A}$ are drawn independently from a zero-mean multivariate Gaussian distribution.

Fig. 2: Comparison of different algorithms for solving the LASSO problem (1).

the iteration number goes to infinity for *infinite* dimensional problems, we focused on more practical *finite* dimensional problems. Specifically, for any finite dimensional matrix $\mathbf{A}$, whenever our algorithm converges, it converges to the LASSO solution. We emphasize that this is not the case for AMP with finite dimensional $\mathbf{A}$ even when $\mathbf{A}$ has i.i.d. Gaussian entries, as we have seen in Fig. 2a. The proposed algorithm contains a parameter e that depends on the largest singular value of the matrix $\mathbf{A}$. In Proposition 1, we provided conditions on e under which the algorithm is locally stable around the LASSO solution with probability one when entries of $\mathbf{A}$ are drawn from a continuous distribution. Finally, simulation results showed that the number of iterations required for our algorithm to converge is orders of magnitude smaller than that of widely used optimization algorithms such as FISTA [3] and PDHG [7].

REFERENCES

[1] Y. Nesterov, "Gradient methods for minimizing composite objective function," *CORE Report*, 2007.

[2] 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: A Journal Issued by the Courant Institute of Mathematical Sciences*, vol. 57, no. 11, pp. 1413–1457, 2004.

[3] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM J. Imaging Sci.*, vol. 2, no. 1, pp. 183–202, 2009.

[4] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein *et al.*, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends® in Machine learning*, vol. 3, no. 1, pp. 1–122, 2011.

[5] E. Esser, X. Zhang, and T. F. Chan, "A general framework for a class of first order primal-dual algorithms for convex optimization in imaging science," *SIAM J. Imaging Sci.*, vol. 3, no. 4, pp. 1015–1046, 2010.

[6] A. Chambolle and T. Pock, "A first-order primal-dual algorithm for convex problems with applications to imaging," *J. Math. Imaging Vis.*, vol. 40, no. 1, pp. 120–145, 2011.

[7] B. He and X. Yuan, "Convergence analysis of primal-dual algorithms for a saddle-point problem: from contraction perspective," *SIAM J. Imaging Sci.*, vol. 5, no. 1, pp. 119–149, 2012.

[8] D. L. Donoho, A. Maleki, and A. Montanari, "Message-passing algorithms for compressed sensing," *Proc. Nat. Academy Sci. (PNAS)*, vol. 106, no. 45, pp. 18 914–18 919, 2009.

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

[10] C. Rush and R. Venkataramanan, "Finite sample analysis of approximate message passing algorithms," *IEEE Trans. Inf. Theory*, 2018.

[11] M. Bayati and A. Montanari, "The LASSO risk for Gaussian matrices," *IEEE Trans. Inf. Theory*, vol. 58, no. 4, pp. 1997–2017, 2012.

[12] C. Rush, "An asymptotic rate for the LASSO loss," in *The 23rd International Conference on Artificial Intelligence and Statistics (AISTATS)*, ser. Proceedings of Machine Learning Research, vol. 108. PMLR, Aug. 2020, pp. 3664–3673.

[13] S. Rangan, "Generalized approximate message passing for estimation with random linear mixing," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, 2011, pp. 2168–2172.

[14] S. Rangan, P. Schniter, A. K. Fletcher, and S. Sarkar, "On the convergence of approximate message passing with arbitrary matrices," *IEEE Trans. Inf. Theory*, vol. 65, no. 9, pp. 5339–5351, 2019.

[15] S. Rangan, A. K. Fletcher, P. Schniter, and U. S. Kamilov, "Inference for generalized linear models via alternating directions and Bethe free energy minimization," *IEEE Trans. Inf. Theory*, vol. 63, no. 1, pp. 676–697, 2017.

[16] R. T. Rockafellar, *Convex Analysis*. Princeton University Press, 1970.

[17] R. J. Tibshirani, "The lasso problem and uniqueness," *Electronic Journal of Statistics*, vol. 7, pp. 1456–1490, 2013.

[18] Y.-Q. Yin, Z.-D. Bai, and P. R. Krishnaiah, "On the limit of the largest eigenvalue of the large dimensional sample covariance matrix," *Probability Theory and Related Fields*, vol. 78, no. 4, pp. 509–521, 1988.