

An Inverse Matrix-Free Proximal Point Algorithm for Compressive Sensing

Hongchun Sun^{a,*}, Min Sun^{b,c}, Bohan Zhang^d

^a School of Mathematics and Statistics, Linyi University, Shandong, Linyi, 276005, P. R. China

^b School of Mathematics and Statistics, Zaozhuang University, Shandong, Zaozhuang, 277160, P. R. China

^c School of Management, Qufu Normal University, Shandong, Rizhao, 276826, P. R. China

^d School of Information Science and Engineering, Jinan University, Shandong, Jinan, 250022, P. R. China

*Corresponding author, e-mail: hcsun68@126.com

Received 1 Jun 2018

Accepted 26 Aug 2018

ABSTRACT: In recent years, the compressive sensing (CS) has received considerable attention in signal processing and statistical inference. The classical proximal point algorithm (PPA) for some reformulations of CS often involves an inverse matrix at each iteration, which usually requires expensive computation if high dimensional variables are considered. Our contribution in this paper is to propose a novelly inverse matrix-free PPA to solve CS for the first time. More specifically, we first establish some equivalent reformulations of CS, which are smooth and convex. Based on these equivalent reformulations, a new proximal point algorithm is proposed to solve CS, whose inverse matrix can be removed by choosing some special parameter. Thus we get an inverse matrix-free PPA, which is implementable for large scale CS. Global convergence of the new PPA and its inverse matrix-free version is established. Comparative numerical results are presented, which substantiate the efficacy and validity of the inverse matrix-free PPA for solving some sparse signal recovery problems.

KEYWORDS: compressive sensing, inverse matrix-free PPA, global convergence, sparse signal recovery

MSC2010: 90C25, 90C30

INTRODUCTION

The problem of compressive sensing (CS) is considered to be an important issue encountered in the fields of signal processing and statistical inference, which is to recover a sparse signal $\bar{x} \in \mathcal{R}^n$ from an undetermined linear system $y = A\bar{x}$, where $A \in \mathcal{R}^{m \times n}$ ($m \ll n$) is the sensing matrix. A fundamental decoding model of CS is the following basis pursuit denoising (termed as QP_ρ):

$$\min_{x \in \mathcal{R}^n} \frac{1}{2} \|Ax - y\|_2^2 + \rho \|x\|_1, \quad (1)$$

where $\rho > 0$ is the regularization parameter and $\|x\|_1$ is the ℓ_1 -norm of the vector x defined as $\|x\|_1 = \sum_{i=1}^n |x_i|$. Throughout this paper, we assume that the solution set of (1) is nonempty.

The CS has been an important research direction in optimization since its appearance, and the theory about it has been extensively developed in the literature¹. Though the presence of nonsmoothness of ℓ_1 -norm makes QP_ρ become a nonsmooth optimization problem, a lot of numerical algorithms have been investigated in the last decades, which

can be classified into two categories. The first category aims to reformulate QP_ρ as a smooth optimization problem by some splitting technologies. For example, by splitting the decision variable x into two positive variables, Figueiredo et al² transformed QP_ρ into a bound-constrained quadratic programming formulation, and proposed some gradient projection algorithms for the reformulation of QP_ρ . These approaches perform quite well in a wide range of applications, especially the approach based on the famous Barzilai-Borwein method. Then, as a follow-up to Figueiredo et al², Xiao and Zhu³ further transformed QP_ρ into a convex constrained monotone equations, and presented a conjugate gradient method for the equivalent forms of QP_ρ . The numerical results³ substantiated the efficacy of such conjugate gradient method. Wang et al⁴ proposed a Lagrange-dual reformulation of QP_ρ , whose dimensionality is m , which is much smaller than n , the dimensionality of the original problem QP_ρ . Furthermore, it is smooth and convex. Therefore, many state of the art gradient-type algorithms can be used to solve the reformulation. The second

category reformulates QP_ρ as a separable convex programming by introducing some auxiliary variable. All the numerical methods which can solve the separable convex programming are applicable to QP_ρ , such as the alternating direction method of multipliers and its linearized version^{7-10,14}, the Peaceman-Rachford splitting method of multipliers^{11-13,15,16}, the symmetric alternating direction method of multipliers⁵, etc. Compared to the methods in the first category, the methods in the second category don't need the backtracking line search to find the steplength.

Since the work of Martinet⁶, the proximal point algorithm (PPA) has received much attention of researchers and practitioners and it is a powerful tool for algorithmic interpretation. Many famous numerical methods for optimization problems can be casted to PPA framework. The classical PPA for some reformulations of CS often involves an inverse matrix at each iteration, which usually requires expensive computation if high dimensional variables are considered. To the best of the authors' knowledge, there is no inverse matrix-free PPA for QP_ρ . On the basis of the previous work^{2,3,11}, in this paper, we are going to propose a new numerical method for QP_ρ , which is quite simple in the sense that its iterative scheme has a closed-form expression. Furthermore, the new method can also be casted to the proximal point algorithm framework. Though its iterative scheme involves an inverse matrix, by choosing some suitable parameter, the new method can reduce to an inverse matrix-free method.

The rest of this paper is organized as follows. In the next section some equivalent reformulations of QP_ρ are established. Then we propose a new PPA and an inverse matrix-free PPA for QP_ρ . The global convergence of both new methods are discussed in detail. Some numerical experiments on compressive sensing are given to illustrate the efficiency of the proposed method. Finally, some concluding remarks are drawn in the last section.

To end this section, some notations used in this paper are in order. We use \mathcal{R}_+^n to denote the nonnegative quadrant of \mathcal{R}^n , and x_+ to denote the orthogonal projection of vector $x \in \mathcal{R}^n$ onto \mathcal{R}_+^n , that is, $(x_+)_i := \max\{x_i, 0\}$, $1 \leq i \leq n$. Furthermore, $\|\cdot\|_1$, $\|\cdot\|$ and $\|\cdot\|_M$ denote the Euclidean 1-norm, 2-norm and M -norm, respectively. For any two vectors $x, y \in \mathcal{R}^n$, we use $(x; y)$ to denote the column vector $(x^\top, y^\top)^\top$, and use I_m to denote an identity matrix of order m . The transpose of a matrix M is denoted by M^\top . λ_{\min}^M and λ_{\max}^M denote the

minimum eigenvalue and the maximum eigenvalue of the matrix M , respectively.

EQUIVALENT REFORMULATIONS OF QP_ρ

In this section, we shall establish some equivalent reformulations of QP_ρ . These reformulations are the basis of our analysis to be conducted.

First, to get a smooth reformulation²⁻⁴ of QP_ρ , let us introduce two sets of auxiliary variables μ_i and ν_i ($i = 1, 2, \dots, n$), which satisfy

$$\mu_i + \nu_i = |x_i|, \quad \mu_i - \nu_i = x_i, \quad i = 1, 2, \dots, n.$$

Then, the QP_ρ can be equivalently reformulated as the following smooth optimization problem

$$\min_{\substack{(\mu; \nu) \in \mathcal{R}^{2n} \\ \text{s.t. } \mu, \nu \geq 0}} \frac{1}{2} \|(A, -A)(\mu; \nu) - y\|_2^2 + \rho(e^\top, e^\top)(\mu; \nu) \quad (2)$$

where $e \in \mathcal{R}^n$ denotes the vector composed by elements 1, i.e., $e = (1, 1, \dots, 1)^\top$.

By a simple manipulation, we can further rewrite (2) as the following compact form:

$$\min_{(\mu; \nu) \in \mathcal{R}^{2n}} \frac{1}{2} [(\mu; \nu)^\top M(\mu; \nu) - 2p^\top(\mu; \nu) + y^\top y] \quad (3)$$

where $M = (A, -A)^\top(A, -A)$ and $p = (A, -A)^\top y - \rho(e^\top, e^\top)^\top$.

It is easy to check that the Hessian matrix M of the quadratic function $f(\mu; \nu)$ is positive semi-definite, so $f(\mu; \nu)$ is a convex function. Furthermore, the feasible set is a polyhedral. Thus, problem (3) is a standard convex optimization. By the theory of the convex optimization¹⁷, we have that the stationary set of (3) coincides with its solution set, which again coincides with the solution set of the following linear variational inequality problem: find $(\mu; \nu)^* \in \mathcal{R}_+^{2n}$ such that

$$((\mu; \nu) - (\mu; \nu)^*)^\top (M(\mu; \nu)^* - p) \geq 0, \quad (4)$$

for all $(\mu; \nu) \in \mathcal{R}_+^{2n}$. Obviously, the solution set of (4) is nonempty under the nonempty assumption of the solution of (1). That is, the problem (4) is also an equivalent reformulation of the QP_ρ .

In conclusion,

$$QP_\rho \Leftrightarrow (2) \Leftrightarrow (3) \Leftrightarrow (4).$$

ALGORITHM AND GLOBAL CONVERGENCE

In this section, based on the equivalent reformulations presented above, we shall propose a new

proximal point algorithm and its inverse matrix-free version to solve QP_ρ . Furthermore, their global convergence is proved under some standard assumptions.

First, the new PPA for QP_ρ is described as follows.

Algorithm 1

Step 1. Choose the parameters $\varepsilon \geq 0, \gamma \geq 0$, and a symmetric matrix $N \in \mathcal{R}^{2n \times 2n}$ such that

$$\gamma > \max \left\{ -2\lambda_{\min}^N + \frac{1}{2}\lambda_{\max}^M, -\lambda_{\min}^N \right\}. \quad (5)$$

Select a randomly generated initial guess $(\mu; \nu)^0 \in \mathcal{R}^{2n}$, and set $k \triangleq 0$.

Step 2. Compute the new iterate $(\mu; \nu)^{k+1}$ by

$$(\mu; \nu)^{k+1} = \frac{1}{2} \left\{ \left(M + \frac{\gamma}{2}I + N \right)^{-1} \left[(M + \gamma I + 2N)(\mu; \nu)^k + p \right] \right\}_+. \quad (6)$$

Step 3. If $\|(\mu; \nu)^{k+1} - (\mu; \nu)^k\| \leq \varepsilon$, then stop; Otherwise, go to **Step 2** with $k \triangleq k + 1$.

Remark 1 In Algorithm 1, by (5), we obtain

$$\begin{aligned} \lambda_{\min}^M + \frac{\gamma}{2} + \lambda_{\min}^N &> \lambda_{\min}^M - \lambda_{\min}^N + \frac{1}{4}\lambda_{\max}^M + \lambda_{\min}^N \\ &= \frac{1}{4}\lambda_{\max}^M + \lambda_{\min}^M \geq 0. \end{aligned}$$

Thus, the matrix $M + \frac{\gamma}{2}I + N$ is positive definite, and Algorithm 1 is well-defined.

To establish the convergence of Algorithm 1, we first give the following two lemmas, which lay a basis for further discussion.

Lemma 1 *The iterate $(\mu; \nu)^{k+1}$ generated by iterative scheme (6) is a solution of the following strictly quadratic programming*

$$\min_{(\mu; \nu) \in \mathcal{R}_+^{2n}} \tilde{f}(u, \nu) \quad (7)$$

where

$$\begin{aligned} \tilde{f}(u, \nu) &= [(\mu; \nu) - (\mu; \nu)^k]^\top (M(\mu; \nu) - p) \\ &\quad + \frac{\gamma}{2} \|(\mu; \nu) - (\mu; \nu)^k\|^2 \\ &\quad + [(\mu; \nu) - (\mu; \nu)^k]^\top N [(\mu; \nu) - (\mu; \nu)^k] \end{aligned}$$

where N defined in Algorithm 1.

Proof: To make the work more concise, we introduce the notation $\xi = (\mu; \nu), \xi^k = (\mu; \nu)^k, \xi^{k+1} = (\mu; \nu)^{k+1}$. By a manipulation, we have

$$\begin{aligned} \xi^{k+1} &= \operatorname{argmin}_{\xi \in \mathcal{R}_+^{2n}} \left\{ \|\xi - q\|_{(M + \frac{\gamma}{2}I + N)}^2 \right\} \\ &= \operatorname{argmin}_{\xi \in \mathcal{R}_+^{2n}} \left\{ \|\xi - q\|_{(M + \frac{\gamma}{2}I + N)}^2 - q^\top \left(M + \frac{\gamma}{2}I + N \right) q \right\} \\ &= \operatorname{argmin}_{\xi \in \mathcal{R}_+^{2n}} \left\{ \|\xi - q\|_{(M + \frac{\gamma}{2}I + N)}^2 - q^\top \left(M + \frac{\gamma}{2}I + N \right) q \right. \\ &\quad \left. + p^\top \xi^k + \frac{1}{2} \gamma (\xi^k)^\top \xi^k + (\xi^k)^\top N \xi^k \right\} \\ &= \operatorname{argmin}_{\xi \in \mathcal{R}_+^{2n}} \left\{ (\xi - \xi^k)^\top (M \xi - p) + \frac{1}{2} \gamma \|\xi - \xi^k\|^2 \right. \\ &\quad \left. + (\xi - \xi^k)^\top N (\xi - \xi^k) \right\}, \quad (8) \end{aligned}$$

where $q = \frac{1}{2}(M + \frac{\gamma}{2}I + N)^{-1}((M + \gamma I + 2N)\xi^k + p)$. Then, the desired result follows. \square

Lemma 2 *If the vector $(\mu; \nu)^*$ is a solution of (4) and $\lambda_{\max}^M \neq 0$, then we have*

$$\begin{aligned} &\langle M(\mu; \nu) - p, (\mu; \nu) - (\mu; \nu)^* \rangle \\ &\geq \frac{1}{\lambda_{\max}^M} \|M((\mu; \nu) - (\mu; \nu)^*)\|^2, \end{aligned}$$

for all $(\mu; \nu) \in \mathcal{R}_+^{2n}$.

Proof: For the ease of description, we introduce the notation $\xi = (\mu; \nu), \xi^* = (\mu; \nu)^*$. Since the matrix M is positive semi-definite, there exists an orthogonal matrix P such that

$$PMP^\top = \operatorname{diag}(\lambda_1, \dots, \lambda_s, \vec{0}),$$

with $\lambda_i > 0$ ($i = 1, 2, \dots, s$). Setting $\zeta = P(\xi - \xi^*)$ and by a direct manipulation, we have

$$\begin{aligned} &\langle (M \xi - p) - (M \xi^* - p), \xi - \xi^* \rangle \\ &= (\xi - \xi^*)^\top M (\xi - \xi^*) \\ &= (\xi - \xi^*)^\top P^\top \operatorname{diag}(\lambda_1, \dots, \lambda_s, \vec{0}) P (\xi - \xi^*) \\ &\geq \frac{1}{\lambda_{\max}^M} (\lambda_1^2 \zeta_1^2 + \dots + \lambda_s^2 \zeta_s^2) \\ &= \frac{1}{\lambda_{\max}^M} (\xi - \xi^*)^\top P^\top \operatorname{diag}(\lambda_1^2, \dots, \lambda_s^2, \vec{0}) P (\xi - \xi^*) \\ &= \frac{1}{\lambda_{\max}^M} \|M(\xi - \xi^*)\|^2 \quad (9) \end{aligned}$$

Since ξ^* is a solution of (4), one has

$$\langle M \xi^* - p, \xi - \xi^* \rangle \geq 0 \quad \forall \xi \in \mathcal{R}_+^{2n}.$$

This and (9) imply the conclusion of this lemma immediately. \square

Remark 2 Algorithm 1 is a descent method in the sense that the generated sequence $\{(\mu; \nu)^k\}$ satisfies $f((\mu; \nu)^{k+1}) \leq f((\mu; \nu)^k)$, where $f(\cdot)$ is the objective function of (3). In fact, from (5), the matrix $M + \frac{\gamma}{2}I + N$ is positive definite. Then $(\mu; \nu)^{k+1}$ is the unique solution of (7). In addition, since the quadratic programming (7) is strictly convex, it can be equivalently reformulated as the following linear variational inequality problem:

$$\left\langle 2\left(M + \frac{1}{2}\gamma I + N\right)\left((\mu; \nu)^{k+1} - (\mu; \nu)^k\right) + M(\mu; \nu)^k - p, (\mu; \nu) - (\mu; \nu)^{k+1} \right\rangle \geq 0, \quad (10)$$

for all $(\mu; \nu) \in \mathcal{R}_+^{2n}$. From (3), we have

$$\begin{aligned} & f((\mu; \nu)^{k+1}) - f((\mu; \nu)^k) \\ &= \frac{1}{2} \left[((\mu; \nu)^{k+1})^\top M (\mu; \nu)^{k+1} - 2p^\top (\mu; \nu)^{k+1} + y^\top y \right] \\ &\quad - \frac{1}{2} \left[((\mu; \nu)^k)^\top M (\mu; \nu)^k - 2p^\top (\mu; \nu)^k + y^\top y \right] \\ &= \frac{1}{2} \left[((\mu; \nu)^{k+1} - (\mu; \nu)^k)^\top M ((\mu; \nu)^{k+1} - (\mu; \nu)^k) \right. \\ &\quad \left. + \langle M(\mu; \nu)^k - p, (\mu; \nu)^{k+1} - (\mu; \nu)^k \rangle \right] \\ &\leq \frac{1}{2} \left[((\mu; \nu)^{k+1} - (\mu; \nu)^k)^\top M ((\mu; \nu)^{k+1} - (\mu; \nu)^k) \right. \\ &\quad \left. - \left\langle 2\left(M + \frac{1}{2}\gamma I + N\right)\left((\mu; \nu)^{k+1} - (\mu; \nu)^k\right), \right. \right. \\ &\quad \quad \left. \left. (\mu; \nu)^{k+1} - (\mu; \nu)^k \right\rangle \right] \\ &= ((\mu; \nu)^{k+1} - (\mu; \nu)^k)^\top \left[\frac{1}{2}M - 2\left(M + \frac{1}{2}\gamma I + N\right) \right] \\ &\quad \times ((\mu; \nu)^{k+1} - (\mu; \nu)^k) \\ &= -((\mu; \nu)^{k+1} - (\mu; \nu)^k)^\top \left[\frac{3}{2}M + \gamma I + N \right] \\ &\quad \times ((\mu; \nu)^{k+1} - (\mu; \nu)^k) \\ &\leq 0, \end{aligned}$$

where the first inequality follows from (10) with $(\mu; \nu) = (\mu; \nu)^k$, and the second inequality holds since the matrix $\frac{3}{2}M + \gamma I + N$ is positive definite (see (5)).

Remark 3 The linear variational inequality problem (10) can be written as the following compact form:

$$\left\langle M(\mu; \nu)^{k+1} - p + (M + \gamma I + 2N) \times ((\mu; \nu)^{k+1} - (\mu; \nu)^k), (\mu; \nu) - (\mu; \nu)^{k+1} \right\rangle \geq 0,$$

for all $(\mu; \nu) \in \mathcal{R}_+^{2n}$. Therefore, Algorithm 1 can be viewed as a customized proximal point algorithm for (1), and the matrix associated with the proximal regularization term is $M + \gamma I + 2N$, which is positive definite. Furthermore, the above inequality and (4) indicate that the stopping criterion of Algorithm 1 is reasonable. That is, the vector $(\mu; \nu)^k$ is a solution of (4) if $\|(\mu; \nu)^{k+1} - (\mu; \nu)^k\| = 0$, so the ensuing vector x^k with $x^k = \mu^k - \nu^k$ is a solution of QP_ρ , where $(\mu^k, \nu^k) = (\mu, \nu)^k$.

In the following, the tolerance ε in Algorithm 1 is set to zero. Based on Lemmas 1 and 2, we get the global convergence of Algorithm 1 as follows.

Theorem 1 Suppose that the assumption (5) holds. Then, the sequence $\{(\mu; \nu)^k\}$ generated by Algorithm 1 terminates at a solution of (2) in a finite number of steps or converges globally to a solution of (2).

Proof: The proof is divided into two cases. For the ease of description, let $\xi = (\mu; \nu)$, $\xi^k = (\mu; \nu)^k$, $\xi^{k+1} = (\mu; \nu)^{k+1}$, $\xi^* = (\mu; \nu)^*$, $\hat{\xi} = (\hat{\mu}; \hat{\nu})$.

Case 1 $\{\xi^k\}$ is a finite sequence. Then, for sufficiently large k , we have $\xi^{k+1} = \xi^k$. Combining this with (10), we have

$$\langle M\xi^k - p, \xi - \xi^k \rangle \geq 0 \quad \text{for all } \xi \in \mathcal{R}_+^{2n}. \quad (11)$$

Thus, ξ^k is a solution of (4), i.e., the sequence $\{\xi^k\}$ terminates at a solution of (2) in a finite number of steps.

Case 2 $\{\xi^k\}$ is an infinite sequence, i.e. $\xi^{k+1} \neq \xi^k$ for all k . Set $\mathcal{M}_\gamma = M + \frac{1}{2}\gamma I + N$ and

$$\Psi(\xi) = (\xi - \xi^*)^\top \mathcal{M}_\gamma (\xi - \xi^*) + \langle M\xi^* - p, \xi - \xi^* \rangle, \quad (12)$$

where ξ^* is a solution of (2). Combining this with (4) and using (5), it is easy to deduce

$$\begin{aligned} \Psi(\xi) &\geq (\xi - \xi^*)^\top \mathcal{M}_\gamma (\xi - \xi^*) \\ &\geq \left(\frac{1}{2}\gamma + \lambda_{\min}^N \right) \|\xi - \xi^*\|^2 \geq 0. \end{aligned} \quad (13)$$

In this following, we are going to study the monotonicity property of the sequence $\{\Psi(\xi^k)\}$. By the definition of $\Psi(\xi)$, we have

$$\begin{aligned} & \Psi(\xi^k) - \Psi(\xi^{k+1}) \\ &= (\xi^k - \xi^*)^\top \mathcal{M}_\gamma (\xi^k - \xi^*) + \langle M\xi^* - p, \xi^k - \xi^* \rangle \\ &\quad - (\xi^{k+1} - \xi^*)^\top \mathcal{M}_\gamma (\xi^{k+1} - \xi^*) \end{aligned}$$

$$\begin{aligned}
 & -\langle M\xi^* - p, \xi^{k+1} - \xi^* \rangle \\
 = & (\xi^k)^\top \mathcal{M}_\gamma \xi^k - (\xi^*)^\top \mathcal{M}_\gamma \xi^* - 2\langle \mathcal{M}_\gamma \xi^*, \xi^k - \xi^* \rangle \\
 & - (\xi^{k+1})^\top \mathcal{M}_\gamma \xi^{k+1} + (\xi^*)^\top \mathcal{M}_\gamma \xi^* \\
 & + 2\langle \mathcal{M}_\gamma \xi^*, \xi^{k+1} - \xi^* \rangle + \langle M\xi^* - p, \xi^k - \xi^{k+1} \rangle \\
 = & (\xi^k)^\top \mathcal{M}_\gamma \xi^k - (\xi^{k+1})^\top \mathcal{M}_\gamma \xi^{k+1} \\
 & + 2\langle \mathcal{M}_\gamma \xi^*, \xi^{k+1} - \xi^k \rangle + \langle M\xi^* - p, \xi^k - \xi^{k+1} \rangle \\
 = & (\xi^k)^\top \mathcal{M}_\gamma \xi^k - (\xi^{k+1})^\top \mathcal{M}_\gamma \xi^{k+1} \\
 & - 2\langle \mathcal{M}_\gamma \xi^{k+1}, \xi^k - \xi^{k+1} \rangle + \langle M\xi^* - p, \xi^k - \xi^{k+1} \rangle \\
 & + 2\langle \mathcal{M}_\gamma (\xi^{k+1} - \xi^*), \xi^k - \xi^{k+1} \rangle \\
 = & (\xi^k - \xi^{k+1})^\top \mathcal{M}_\gamma (\xi^k - \xi^{k+1}) \\
 & + 2\langle \mathcal{M}_\gamma (\xi^{k+1} - \xi^k), \xi^* - \xi^{k+1} \rangle \\
 & + \langle M\xi^* - p, \xi^k - \xi^{k+1} \rangle. \tag{14}
 \end{aligned}$$

Using (10) with $\xi = \xi^*$, one has

$$\begin{aligned}
 & \langle 2\mathcal{M}_\gamma (\xi^{k+1} - \xi^k), \xi^* - \xi^{k+1} \rangle \\
 & \geq -\langle M\xi^* - p, \xi^* - \xi^{k+1} \rangle.
 \end{aligned}$$

Combining this with (14), we obtain

$$\begin{aligned}
 & \Psi(\xi^k) - \Psi(\xi^{k+1}) \\
 \geq & (\xi^k - \xi^{k+1})^\top \mathcal{M}_\gamma (\xi^k - \xi^{k+1}) \\
 & - \langle M\xi^k - p, \xi^* - \xi^{k+1} \rangle + \langle M\xi^* - p, \xi^k - \xi^{k+1} \rangle \\
 = & (\xi^k - \xi^{k+1})^\top \mathcal{M}_\gamma (\xi^k - \xi^{k+1}) \\
 & + \langle M\xi^k - p, \xi^k - \xi^* \rangle - \langle M\xi^k - p, \xi^k - \xi^{k+1} \rangle \\
 & + \langle M\xi^* - p, \xi^k - \xi^{k+1} \rangle. \tag{15}
 \end{aligned}$$

From Lemma 2 with $\xi = \xi^k$, we have

$$\langle M\xi^k - p, \xi^k - \xi^* \rangle \geq \frac{1}{\lambda_{\max}^M} \|M(\xi^k - \xi^*)\|^2.$$

Combining this with (15), one has

$$\begin{aligned}
 & \Psi(\xi^k) - \Psi(\xi^{k+1}) \\
 \geq & (\xi^k - \xi^{k+1})^\top \mathcal{M}_\gamma (\xi^k - \xi^{k+1}) \\
 & + \frac{1}{\lambda_{\max}^M} \|M(\xi^k - \xi^*)\|^2 - \langle M(\xi^k - \xi^*), \xi^k - \xi^{k+1} \rangle. \tag{16}
 \end{aligned}$$

By (16), combining (13) with the Cauchy-Schwartz inequality, one has

$$\begin{aligned}
 & \Psi(\xi^k) - \Psi(\xi^{k+1}) \\
 \geq & \left(\frac{1}{2}\gamma + \lambda_{\min}^N\right) \|\xi^k - \xi^{k+1}\|^2 + \frac{1}{\lambda_{\max}^M} \|M(\xi^k - \xi^*)\|^2 \\
 & - \|M(\xi^k - \xi^*)\| \|\xi^k - \xi^{k+1}\|
 \end{aligned}$$

$$\begin{aligned}
 & \geq \left(\frac{1}{2}\gamma + \lambda_{\min}^N\right) \|\xi^k - \xi^{k+1}\|^2 + \frac{1}{\lambda_{\max}^M} \|M(\xi^k - \xi^*)\|^2 \\
 & \quad - \frac{1}{\lambda_{\max}^M} \|M(\xi^k - \xi^*)\|^2 - \frac{1}{4}\lambda_{\max}^M \|\xi^k - \xi^{k+1}\|^2 \\
 = & \left(\frac{1}{2}\gamma + \lambda_{\min}^N\right) \|\xi^k - \xi^{k+1}\|^2 - \frac{1}{4}\lambda_{\max}^M \|\xi^k - \xi^{k+1}\|^2 \\
 = & \frac{1}{2} \left(\gamma + 2\lambda_{\min}^N - \frac{1}{2}\lambda_{\max}^M\right) \|\xi^k - \xi^{k+1}\|^2. \tag{17}
 \end{aligned}$$

From (5), one has $\gamma + 2\lambda_{\min}^N - \frac{1}{2}\lambda_{\max}^M > 0$. Combining this with (17), we have $\Psi(\xi^k) - \Psi(\xi^{k+1}) > 0$, which together with (13) implies that the nonnegative sequence $\{\Psi(\xi^k)\}$ is strictly decreasing. Thus it is convergent. Then, one has $\Psi(\xi^k) - \Psi(\xi^{k+1}) \rightarrow 0$ as $k \rightarrow \infty$. This and (17) imply

$$\lim_{k \rightarrow \infty} \|\xi^k - \xi^{k+1}\| = 0. \tag{18}$$

In addition, the sequence $\{\Psi(\xi^k)\}$ is bounded since it is convergent. Combining this with (13), we have that the sequence $\{\xi^k\}$ is also bounded. Let $\{\xi^{k_i}\}$ be a convergent subsequence of $\{\xi^k\}$, and set

$$\lim_{k_i \rightarrow \infty} \xi^{k_i} = \hat{\xi}. \tag{19}$$

Combining this with (18), we obtain

$$\begin{aligned}
 & \lim_{k_i \rightarrow \infty} \|\xi^{k_i+1} - \hat{\xi}\| \\
 \leq & \lim_{k_i \rightarrow \infty} \|\xi^{k_i+1} - \xi^{k_i}\| + \lim_{k_i \rightarrow \infty} \|\xi^{k_i} - \hat{\xi}\| = 0.
 \end{aligned}$$

This and (10) imply

$$\langle M\hat{\xi} - p, \xi - \hat{\xi} \rangle \geq 0 \quad \text{for all } \xi \in \mathcal{R}_+^{2n}.$$

Thus, $\hat{\xi}$ is a solution of (2). Replacing ξ^* in $\Psi(\xi)$ (see (12)) by $\hat{\xi}$, and the resultant expression is denoted by $\hat{\Psi}(\xi)$. That is,

$$\hat{\Psi}(\xi) = (\xi - \hat{\xi})^\top \mathcal{M}_\gamma (\xi - \hat{\xi}) + \langle M\hat{\xi} - p, \xi - \hat{\xi} \rangle. \tag{20}$$

From (20), we have

$$\begin{aligned}
 & \left(\frac{\gamma}{2} + \lambda_{\min}^N\right) \|\xi - \hat{\xi}\|^2 \leq \hat{\Psi}(\xi) \\
 & \leq \left(\lambda_{\max}^M + \frac{1}{2}\gamma + \lambda_{\max}^N\right) \|\xi - \hat{\xi}\|^2 \\
 & \quad + \|M\hat{\xi} + p\| \|\xi - \hat{\xi}\|. \tag{21}
 \end{aligned}$$

By the same procedure to prove the convergence of the sequence $\{\Psi(\xi^k)\}$, we can prove that the sequence $\{\hat{\Psi}(\xi^{k_i})\}$ is also convergent. Setting $\xi = \xi^{k_i}$ in (21), we get $\hat{\Psi}(\xi^{k_i}) \rightarrow 0$ ($k_i \rightarrow \infty$) by (19). Thus, we have $\hat{\Psi}(\xi^k) \rightarrow 0$ ($k \rightarrow \infty$). By (21) again, we have that the sequence $\{\xi^k\}$ converges globally to $\hat{\xi}$, and the assertion of this theorem is proved. \square

Theorem 2 The sequence $\{x^k\}$ generated by Algorithm 1 terminates in a finite number of steps or converges globally to a solution of (1), where $x^k = \mu^k - \nu^k$, $(\mu^k; \nu^k) = (\mu; \nu)^k$.

Proof: If the sequence $\{(\mu; \nu)^k\}$ terminates in a finite number of steps, the last iterate $(\mu; \nu)^k$ is a solution of (2). Thus the vector x^k with $x^k = \mu^k - \nu^k$ is a solution of (1).

In the following analysis, we assume that the sequence $\{(\mu; \nu)^k\}$ is an infinite sequence. From Theorem 1, we have

$$\lim_{k \rightarrow \infty} (\mu; \nu)^k = (\hat{\mu}; \hat{\nu}). \tag{22}$$

Set $\hat{x} = \hat{\mu} - \hat{\nu}$, and a direct computation yields that

$$\begin{aligned} \|x^k - \hat{x}\| &= \|(\mu^k - \nu^k) - (\hat{\mu} - \hat{\nu})\| \\ &\leq \|(\mu^k - \hat{\mu})\| + \|(\nu^k - \hat{\nu})\| \\ &\leq \|(\mu^k - \hat{\mu})\|_1 + \|(\nu^k - \hat{\nu})\|_1 \\ &= \|(\mu^k - \hat{\mu}; \nu^k - \hat{\nu})\|_1 \\ &\leq \sqrt{2n} \|(\mu^k - \hat{\mu}; \nu^k - \hat{\nu})\| \rightarrow 0 \end{aligned} \tag{23}$$

as $k \rightarrow \infty$, where the second and third inequalities follow from the fact that

$$\|x\| \leq \|x\|_1 \leq \sqrt{n} \|x\| \quad \text{for all } x \in \mathcal{R}^n.$$

Thus, the sequence $\{x^k\}$ converges globally to a solution of (1). The proof is completed. \square

From (6), the computation of $\mathcal{M}_\gamma^{-1} \triangleq (M + \frac{\gamma}{2}I + N)^{-1}$ is time consuming if n is large. In order to remove it, we can set $N = 2\tau I_{2n} - M$ in (6), and some simple manipulations yield that

$$\begin{aligned} (\mu; \nu)^{k+1} &= \frac{1}{2} \left\{ \mathcal{M}_\gamma^{-1} [(M + \gamma I + 2N)(\mu; \nu)^k + p] \right\}_+ \\ &= \frac{1}{2} \left\{ \left(M + \frac{\gamma}{2}I + [2\tau I - M] \right)^{-1} [(M + \gamma I \right. \\ &\quad \left. + 2[2\tau I - M])(\mu; \nu)^k + p] \right\}_+ \\ &= \frac{1}{2} \left\{ \left(\frac{\gamma}{2} + 2\tau \right)^{-1} [(\gamma + 4\tau)I - M](\mu; \nu)^k + p \right\}_+ \\ &= \left\{ \left(I - \frac{1}{2\sigma}M \right) (\mu; \nu)^k + \frac{1}{2\sigma}p \right\}_+, \end{aligned} \tag{24}$$

where $\sigma = \frac{1}{2}\gamma + 2\tau$.

Combining (24) with Algorithm 1, we get the following inverse matrix-free PPA.

Algorithm 2

Step 1. Choose the parameters $\varepsilon > 0, \tau > 0$ and $\gamma > 0$ satisfying

$$\gamma > \max \left\{ -4\tau + \frac{5}{2}\lambda_{\max}^M, -2\tau + \lambda_{\max}^M \right\}. \tag{25}$$

Select a randomly generated initial point $(\mu; \nu)^0 \in \mathcal{R}^{2n}$, and set $k = 0$.

Step 2. Compute the new iterate $(\mu; \nu)^{k+1}$ by

$$(\mu; \nu)^{k+1} = \left\{ \left(I - \frac{1}{2\sigma}M \right) (\mu; \nu)^k + \frac{1}{2\sigma}p \right\}_+, \tag{26}$$

where σ defined in (24);

Step 3. If $\|(\mu; \nu)^{k+1} - (\mu; \nu)^k\| \leq \varepsilon$, stop; Otherwise, go to **Step 2** with $k \triangleq k + 1$.

Theorem 3 Suppose that the assumption (25) holds. Then, the sequence $\{(\mu; \nu)^k\}$ generated by Algorithm 2 and the corresponding sequence $\{x^k\}$ terminate in a finite number of steps or converges globally to a solution of (2) and (1), respectively, where $x^k = \mu^k - \nu^k$ and $(\mu^k; \nu^k) = (\mu; \nu)^k$.

Proof: Obviously, Algorithm 1 reduces to Algorithm 2 by setting $N = 2\tau I_{2n} - M$. Therefore, in order to prove this theorem, we only need to ensure that the condition (5) holds. In fact, by $N = 2\tau I_{2n} - M$, we have $\lambda_{\min}^N = 2\tau - \lambda_{\max}^M$. Combining this with (25), we get the condition (5). Hence, the assertion of this theorem holds by Theorem 1 and Theorem 2. This completes the proof. \square

NUMERICAL EXPERIMENTS

In this section, we provide some numerical tests about compressive sensing to show the efficiency of our proposed method. For comparison, we also give the numerical results of a conjugate gradient descent (denoted by CGD) method proposed recently by Xiao and Zhu³. All codes are written by MATLAB 7.10.0 and performed on a Windows 7 PC with an Pentium(R) Dual-Core CPU T4400, 2.20GHz CPU and 4GB of memory. In the experiment, we set $\rho = 0.01, n = 2^{11}, m = \lfloor n/a \rfloor, k = \lfloor m/b \rfloor$, and the measurement matrix A is generated by MATLAB scripts:

$$[Q, R] = \text{qr}(A', 0); A = Q'.$$

The original signal \bar{x} is generated by

$$\begin{aligned} p &= \text{randperm}(n); \\ x(p(1:k)) &= \text{randn}(k, 1). \end{aligned}$$

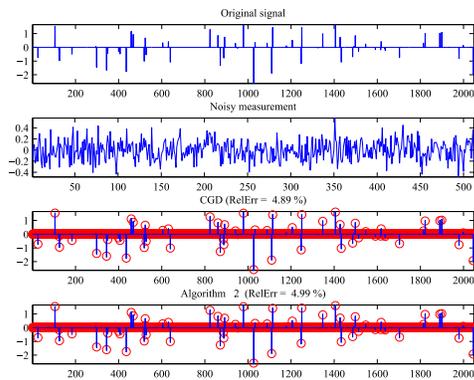


Fig. 1 The original signal, noisy measurement and recovery signal.

The observed signal is $y = A\bar{x} + \bar{n}$, where \bar{n} is generated by a standard Gaussian distribution $N(0, 1)$ and then it is normalized to the norm $\sigma = 0.01$ or 0.001 . For the two methods, the stopping criterion is

$$\frac{\|f_k - f_{k-1}\|}{\|f_{k-1}\|} < 10^{-5},$$

where f_k denotes the objective value of (1) at iteration x_k . For Algorithm 2, we set $\tau = 0.2, \gamma = 0.01$. In addition, the initial points $\mu_0 = \max\{0, A^T y\}$, $\nu_0 = \max\{0, -A^T y\}$. For CGD, we set $\xi = 10, \sigma = 10^{-4}$ and $\rho = 0.5$ in the line search (2.9) of CGD, and the initial points μ_0 and ν_0 are set the same as those in Algorithm 2. In each test, we calculate the relative error

$$\text{Rel. Err} = \frac{\|\tilde{x} - \bar{x}\|}{\|\bar{x}\|},$$

where \tilde{x} denotes the recovery signal.

For $a = 4, b = 8$, the original signal, the measurement and the recovery signal (marked by red point) by Algorithm 2 and CGD are given in Fig. 1.

Obviously, from the last two subplots in Fig. 1, all elements in the original signal are circled by the red points, which indicates that the two methods can recover the original signal quite well. More numerical results are given in Tables 1 and 2, in which, we report the number of iterations, the CPU time in seconds, the relative error of the Algorithm 2 and CGD for various scenarios of (a, b) . Note that the codes of the two methods are repeatedly run 5 times and the listed results are the averages of the numerical results.

Table 1 and Table 2 indicate that Algorithm 2 is always faster than CGD method for all combinations

Table 1 Comparison of Algorithm 2 with CGD for $\sigma = 0.001$.

a	b	Algorithm 2			CGD		
		Time	Iter	Rel. Err	Time	Iter	Rel. Err
4	8	2.509	115.60	0.0466	14.702	210.00	0.0549
3	9	1.616	72.60	0.0361	11.154	159.50	0.0324
2	10	1.136	44.80	0.0219	8.475	123.75	0.0220

Table 2 Comparison of Algorithm 2 with CGD for $\sigma = 0.01$.

a	b	Algorithm 2			CGD		
		Time	Iter	Rel. Err	Time	Iter	Rel. Err
4	8	2.253	109.80	0.0498	15.884	215.20	0.0500
3	9	1.632	75.20	0.0348	10.005	140.33	0.0350
2	10	1.048	44.60	0.0225	7.675	109.67	0.0222

of σ and (a, b) . The reason may be that Algorithm 2 does not to execute any line search at each iteration, while CGD method has to perform a line search to find a suitable steplength. In addition, Algorithm 2 always takes smaller number of iteration than CGD method to achieve the same relative error. Therefore, Algorithm 2 is an efficient method for compressive sensing.

CONCLUSIONS

In this paper, we have developed a new proximal point algorithm for compressive sensing. By choosing some special parameter, the inverse matrix in the new PPA can be removed, and we get an inverse matrix-free PPA. Under some standard assumptions, we have established the global convergence of the new PPA in detail. Finally, some numerical results illustrate that the inverse matrix-free PPA is efficient for the given tests.

According to its limitations, the method proposed in this work has several possible extensions. First, it could be numerically beneficial to tune the parameter γ , and thus it is meaningful to investigate the global convergence of Algorithm 1 with adaptively adjusted parameter γ . Second, Algorithm 2 is tailored for the linear variational inequalities (4). Therefore, how to extend it to nonlinear variational inequalities is worthy of further research.

Acknowledgements: This work was supported by the Natural Science Foundation of China (No.11671228), and the Applied Mathematics Enhancement Program of Linyi University.

REFERENCES

1. Candès EJ, Wakin MB, Boyd SP (2008) Enhancing sparsity by reweighted ℓ_1 -minimization. *J Fourier Anal Appl* **14**(5-6), 877–905.
2. Figueiredo MAT, Nowak RD, Wright SJ (2007) Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems. *IEEE J-STSP* **1**(4), 586–97.
3. Xiao YH, Zhu H (2013) A conjugate gradient method to solve convex constrained monotone equations with applications in compressive sensing. *J Math Anal Appl* **405**(1), 310–9.
4. Wang YJ, Zhou GL, Caccetta L, Liu WQ (2011) An alternative Lagrange-dual based algorithm for sparse signal reconstruction. *IEEE T Signal Proces* **59**(4), 1895–901.
5. He BS, Ma F, Yuan XM (2016) Convergence study on the symmetric version of ADMM with larger step sizes. *SIAM J Imaging Sci* **9**(3), 1467–501.
6. Martinet B (1970) Régularization d' inéquations variationnelles par approximations successives. *Rev Fr Inform Rech Opér* **4**, 154–9.
7. Yang JF, Zhang Y (2011) Alternating direction algorithms for ℓ_1 -problems in compressive sensing. *SIAM J Sci Comput* **33**(1), 250–78.
8. Yang JF, Yuan XM (2013) Linearized augmented Lagrangian and alternating direction methods for nuclear norm minimization. *Math Comput* **82** (281), 301–29.
9. Xiao YH, Song HN (2012) An inexact alternating directions algorithm for constrained total variation regularized compressive sensing problems. *J Math Imaging Vis* **44**(2), 114–27.
10. Cao SH, Xiao YH, Zhu H (2014) Linearized alternating directions method for ℓ_1 -norm inequality constrained ℓ_1 -norm minimization. *Appl Numer Math* **85**, 142–53.
11. Sun M, Liu J (2016) An inexact generalized PRSM with LQP regularization for structured variational inequalities and its applications to traffic equilibrium problems. *J Inequal Appl* **150**, 1–17.
12. Sun M, Wang YJ, Liu J (2017) Generalized Peaceman-Rachford splitting method for multiple-block separable convex programming with applications to robust PCA. *Calcolo* **54**(1), 77–94.
13. He BS, Liu H, Wang ZR, Yuan XM (2014) A strictly contractive Peaceman-Rachford splitting method for convex programming. *SIAM J Optim* **24**(3), 1011–40.
14. Han DR, Yuan XM, Zhang WX, Cai XJ (2013) An ADM-based splitting method for separable convex programming. *Comput Optim Appl* **54**(2), 343–69.
15. Sun HC, Sun M, Zhou HC (2016) A proximal splitting method for separable convex programming and its application to compressive sensing. *J Nonlinear Sci Appl* **9**, 392–403.
16. He BS, Yuan XM (2012) On the $\mathcal{O}(1/n)$ convergence rate of Douglas-Rachford alternating direction method. *SIAM J Numer Anal* **50**, 700–9.
17. Bertsekas DP (1999) *Nonlinear Programming* 2nd ed. Boston, MA: Athena.