Journal of Mathematical Research with Applications
Jan., 2026, Vol. 46, No. 1, pp. 119–133
DOI:10.3770/j.issn:2095-2651.2026.01.010
Http://jmre.dlut.edu.cn

Centralized Circumcentered-Reflection Method for Solving the Convex Feasibility Problem in Sparse Signal Recovery

Chunmei LI, Bangjun CHEN, Xuefeng DUAN*
College of Mathematics and Computational Science, Center for Applied Mathematics of Guangxi (GUET), Guangxi Colleges and Universities Key Laboratory of Data Analysis and Computation, Guilin University of Electronic Technology, Guangxi 541004, *P. R. China*

Abstract Convex feasibility problems are widely used in image reconstruction, sparse signal recovery, and other areas. This paper is devoted to considering a class of convex feasibility problem arising from sparse signal recovery. We first derive the projection formulas for a vector onto the feasible sets. The centralized circumcentered-reflection method is designed to solve the convex feasibility problem. Some numerical experiments demonstrate the feasibility and effectiveness of the proposed algorithm, showing superior performance compared to conventional alternating projection methods.

Keywords convex feasibility problem; centralized circumcentered-reflection method; sparse signal recovery; compressed sensing

MR(2020) Subject Classification 65K05; 65F50; 90C25; 94A12

1. Introduction

Throughout this paper, we use $\mathcal{R}^{m\times n}$ to represent the set of $m\times n$ real matrices. The symbol $\|\cdot\|$ denotes the induced norm. The symbols H^+, H^T and $\text{rank}(H)$ are used to represent the Moore-Penrose inverse, the transpose and the rank of the matrix H, respectively. The symbol $\langle\cdot,\cdot\rangle$ stands for inner product and $\|x\|_0$ stands for the number of non-zero entries in vector x.

In signal processing, if an original signal x is sparse, it can be recovered from the measurement vector y by using its sparsity and identifying the sparsest representation among all possible x that satisfy $y = Hx$, as discussed in [1–4]. The sparse recovery problem can be formulated as the following optimization problem

$$\min \|x\|_0 \quad \text{s.t.} \quad y = Hx, \tag{1.1}$$

where $H \in \mathcal{R}^{m\times n}$ is a measurement matrix satisfying the Restricted Isometry Property (RIP) (see Definition 2.2). If the original signal x is not sparse, it is necessary to find a set of bases such that the signal x can be sparsely represented (i.e., $x = \psi\alpha$). In this case, the original signal x can be reconstructed from the measurement vector $y = H\psi\alpha$. When the original signal x is

Received April 1, 2025; Accepted September 8, 2025
Supported by the Natural Science Foundation of Guangxi Province (Grant Nos. 2023GXNSFAA026067; 2024GXNSFAA010521) and the National Natural Science Foundation of China (Nos. 12361079; 12201149; 12261026).
* Corresponding author
E-mail address: guidian520@126.com (Xuefeng DUAN)

sparse, ψ is typically set to the identity matrix. In the presence of noise, the observation becomes $y = Hx + e$, where $e \in R^m$ denotes noise. Problem (1.1) is relaxed as

$$\min \|x\|_0 \quad \text{s.t.} \quad \|y - Hx\| \leq \epsilon.$$

Unfortunately, the problem (1.1) as an l_0-minimization problem, has been proven to be NP-hard [5], due to its non-convex nature. However, in order to solve this problem, some researchers have proposed convex relaxation of the problem (1.1). An effective way to solve this problem is to replace the l_0-norm in (1.1) with the l_1-norm [6], i.e.,

$$\min \|x\|_1 \quad \text{s.t.} \quad y = Hx. \tag{1.2}$$

Candes and Tao [1] also explicitly demonstrated that when the sensing matrix H satisfies the RIP with certain constant parameters, the l_1-optimization problem can completely replace the l_0-optimization problem. Let

$$\Omega_1 = \{x | y = Hx\} \text{ and } \Omega_2 = \{x | \|x\|_1 \leq s\},$$

where s is a predefined threshold that constrains the l_1-norm of the solution. Then one can relax the sparse recovery problem (1.2) into a convex feasibility problem. In other words, the problem (1.2) is equivalent to finding a vector $x \in \Omega_1 \cap \Omega_2$.

The convex feasibility problem is of great research significance because many mathematical problems and practical application problems in real life can be reduced to a convex feasibility problem. For example, optimization [7], sparse recovery, compressed sensing, approximation theory [8], sensor network location problems [9–11], and image reconstruction [12]. Consequently, there has been a steadily increasing interest in the development of both theoretical and numerical methods for finding a solution to the convex feasibility problem. However, it is very difficult to find a point in the intersection of convex sets directly. Projection algorithms, which originated from the research work of Neumann in 1950 (see [13]) (for more details see [14]), are commonly used to solve this problem but often converge slowly. To overcome this slow-convergence issue, Censor proposed the relaxed projection algorithm in 1981 (see [15]). However, this method may fail to guarantee complete convergence in incompatible problems. Subsequently, the parallel projection algorithm was proposed [16]. To enhance the flexibility of the parallel projection algorithm, Censor proposed the string averaging projection algorithm [17] and Dang proposed the block subgradient projection algorithm [18]. The extrapolated relaxation method [19] and subgradient projection method [20] were also proposed to overcome the above difficulties.

In general, the primary methods for solving convex feasibility problem are alternating projection algorithms [21,22] or simultaneous projection algorithms [23,24], with most other approaches being improvements based on these algorithms. For instance, the Douglas-Rachford (DR) algorithm, an extension of the alternating projection method, iteratively applies the reflection operator to approximate points within the intersection of convex sets. When the intersection is non-empty, the DR algorithm generates a sequence that converges to points within the intersection set, exhibiting strong convergence properties [25]. However, in each iteration, the DR algorithm requires the computation of exact projections, which results in high computational

costs, particularly when the projection operations themselves are complex or computationally expensive. To overcome this problem, Millan proposed the alternating conditional gradient method [26] and approximate DR algorithm [27]. These methods used the conditional gradient method to compute approximate projections rather than exact projections, which reduces computational cost while still being capable of handling large-scale problems. With the aid of stochastic approximation techniques, both the minibatch stochastic subgradient-based projection algorithm [28] and the stochastic block projection algorithms with extrapolation [29] have been employed to solve large-scale convex feasibility problems. While these algorithms offer computational advantages, they are sensitive to the random selection of a fixed sample size, which can impact their performance and reliability.

In this paper, we consider the l_1-minimization problem (1.2) in the sparse signal recovery. The projections $P_{\Omega_1}(x)$ and $P_{\Omega_2}(x)$ of x onto the Ω_1 and Ω_2 are given. The centralized circumcentered-reflection method for solving the problem (1.2) is designed and its convergence analysis is also given. We use numerical experiments to demonstrate that the new algorithm is feasible. Especially, it converges faster than the commonly used alternating projection algorithms.

The organization of this paper is as follows. The centralized circumcentered-reflection method is introduced in Section 2. In Section 3, we derive some projection formulas and design the centralized circumcentered-reflection method to solve the problem (1.2). Finally, some numerical experiments are used to illustrate the feasibility and efficiency of the new method.

2. Centralized circumcentered-reflection method

In this section, we introduce the circumcentered-reflection method and its convergence theorem. Alternating projection and simultaneous projection algorithms are commonly used to find a point in the intersection of some convex sets by using a sequence of projections onto the sets. Assume A and B are closed convex sets in $\mathcal{R}^n$, with P_A and P_B being the orthogonal projections onto A and B, respectively. Then for any initial point $x^0 \in \mathcal{R}^n$, the iterative formula for the alternating projection algorithm is given by

$$x^{k+1} = P_B P_A(x^k), \tag{2.1}$$

then according to Cegielski [23, Theorem 4.3.4], if $A \cap B \neq \emptyset$, the sequence $\{x^k\}$ generated by (2.1) converges to a point $x^* \in A \cap B$. The iteration formula of the simultaneous projection algorithm is

$$x^{k+1} = \frac{1}{2}(P_A(x^k) + P_B(x^k)). \tag{2.2}$$

If $A \cap B \neq \emptyset$, by Pierro and Iusem [24, Corollary 3], the sequence $\{x^k\}$ generated by (2.2) converges to a point $x^* \in A \cap B$. See Figure 1 for more details.

Although the simultaneous projection method can be used to solve convex feasibility problems, its convergence rate is rather slow. To accelerate convergence, the parallel circumcentered-reflection method was proposed. Starting from a point $x^0 \in \mathcal{R}^n$, the next iterate x^1 is the

circumcenter of the triangle of vertexes $x^0, R_{\Omega_1}(x^0)$ and $R_{\Omega_2}(x^0)$. The iterative format of this method is presented below

$$x_{pCRM}^{k+1} = \mathcal{A}(x^k) = \text{circumcenter}\{x^k, R_{\Omega_1}(x^k), R_{\Omega_2}(x^k)\},$$

where R_{Ω_1}, R_{Ω_2} are orthogonal reflection operators on the sets Ω_1 and Ω_2. These operators are defined as $R_{\Omega_1} = 2P_{\Omega_1} - Id$, $R_{\Omega_2} = 2P_{\Omega_2} - Id$, where Id represents the identity operator, P_{Ω_1} and P_{Ω_2} are the orthogonal projections onto Ω_1 and Ω_2, respectively. However, this method has a disadvantage, namely, starting from a given initial point, the next iterative point may deviate from the solution set, as shown in Figure 2. To overcome this challenge, we attempt to preprocess the point x^k before iteration, that is, use the combination of alternating projection points and simultaneous projection points of x^k to obtain $x_c^k = \frac{1}{2}(x_{AP} + P_{\Omega_1}(x_{AP}))$, where $x_{AP} = P_{\Omega_2}P_{\Omega_1}(x^k)$, and then use the parallel circumcenter reflection method for iteration. This method is called the centralized circumcentered-reflection method (denoted by cCRM) and iterative formula is presented below

$$x_{cCRM}^{k+1} = \mathcal{A}(x_c^k) = \text{circumcenter}\{x_c^k, R_{\Omega_1}(x_c^k), R_{\Omega_2}(x_c^k)\}. \tag{2.3}$$

From Figure 2, we can intuitively see that starting from x^0 to x_{cCRM}^1 is already better than moving from x^0 to x_{pCRM}^1, i.e., x_{cCRM}^1 is closer to the solution.

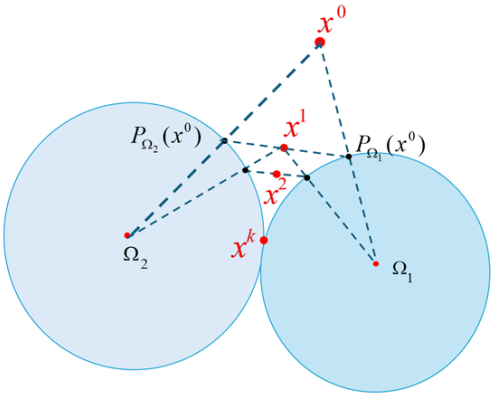


Figure 1 First few iterations of simultaneous projection algorithm

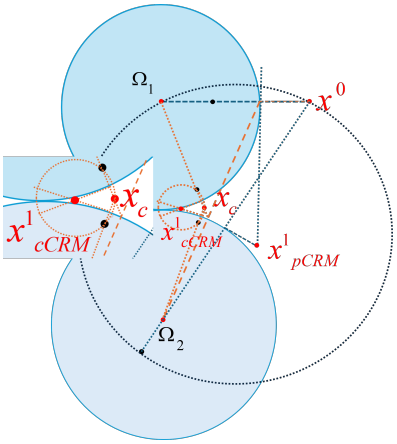


Figure 2 Characterization of pCRM and cCRM

Interestingly, when Ω_1 represents a half-space and Ω_2 corresponds to an affine set with nonempty intersection $\Omega_1 \cap \Omega_2$. Using the centralized circumcentered-reflection method to find the intersection point $x \in \Omega_1 \cap \Omega_2$ can be achieved in just one step, i.e.,

$$P_{\Omega_1 \cap \Omega_2}(x) = \text{circumcenter}\{x, R_{\Omega_1}(x), R_{\Omega_2}(x)\},$$

which also shows the efficiency of this method. This is particularly challenging for alternating projection methods, especially when the angle between the two sets is small. We also give the corresponding lemma and convergence theorem of the centralized circumcentered-reflection

method as follows.

Definition 2.1 ([30]) *Suppose $\mathbb{M}$ is a real Hilbert space, $C \subseteq \mathbb{M}$ is a closed convex nonempty set and $\{x_n\}$ is a sequence in $\mathbb{M}$, we say that $\{x_n\}$ is Fejér monotone with regard to C, if*

$$\|x_{n+1} - z\| \leq \|x_n - z\|,$$

for all $z \in C$ and every $n > 0$.

Definition 2.2 ([4]) *A matrix $H \in \mathcal{R}^{m\times n}$ is said to satisfy the RIP with a restricted isometry constant $\delta_s \in (0, 1)$ if for every s-sparse vector $x \in \mathcal{R}^n$ (i.e., $\|x\|_0 \leq s$), the following inequality holds*

$$(1 - \delta_s)\|x\|_2^2 \leq \|Hx\|_2^2 \leq (1 + \delta_s)\|x\|_2^2.$$

Theorem 2.3 ([31]) *Let $\Omega_1, \Omega_2 \in \mathcal{R}^n$ be two closed convex sets with nonempty intersection. Then for any initial point $x^0 \in \mathcal{R}^n$, the sequence $\{x^k\}$ generated by (2.3) is Fejér monotone with regard to $\Omega_1 \cap \Omega_2$, and converges to an element $x^* \in \Omega_1 \cap \Omega_2$.*

3. Centralized circumcentered-reflection method for solving the problem (1.2)

In this section, we use the centralized circumcentered-reflection method (2.3) to solve the convex feasibility problem (1.2) arising in sparse recovery.

The primary challenges in applying the circumcentered-reflection method to solve the problem (1.2) have two-fold. Firstly, how to compute the projections $P_{\Omega_1}(x)$ and $P_{\Omega_2}(x)$ of x onto Ω_1 and Ω_2, respectively. Secondly, how to derive an explicit formula for the computation of the circumcenter $\{x, R_{\Omega_1}(x), R_{\Omega_2}(x)\}$. These challenges are effectively addressed as follows.

Theorem 3.1 *Assume $H \in \mathcal{R}^{m\times n}$, $\mathrm{rank}(H) = m$ and $x_1 \in \mathcal{R}^n$, then the projection $P_{\Omega_1}(x_1)$ of x_1 onto Ω_1 is*

$$P_{\Omega_1}(x_1) = x_1 + H^+(y - Hx_1),$$

where $H^+ = H^T(HH^T)^{-1}$ is the pseudo-inverse.

Proof Let $x \in \Omega_1$, which implies $y = Hx$. For a given x_1, we have

$$Hx_1 - y = Hx_1 - Hx = H(x_1 - x).$$

Let $r = x_1 - x$. We have $Hx_1 - y = Hr$. Since the unique minimal Euclidean norm solution of the linear system $Hx = y$ is H^+y, we can conclude that $H^+(Hx_1 - y)$ is the solution of the following optimization problem

$$\min \|r\|_2 \quad \text{s.t.} \quad Hr = Hx_1 - y.$$

Noting that $r = x_1 - x$, then the best approximation in Ω_1 to x_1 is

$$P_{\Omega_1}(x_1) = x_1 - H^+(Hx_1 - y),$$

which can be rewritten as

$$P_{\Omega_1}(x_1) = x_1 + H^+(y - Hx_1).$$

This completes the proof. □

Theorem 3.2 *For a given vector $x \in \mathcal{R}^n$ and a positive number s, then the projection $P_{\Omega_2}(x)$ of x onto Ω_2 is as follows*

$$P_{\Omega_2}(x) = \begin{cases} x, & \text{if } \|x\|_1 \le s, \\ \frac{s}{\|x\|_1}x, & \text{if } \|x\|_1 > s. \end{cases} \tag{3.1}$$

Proof Let $f(t) = \|x - t\|_1$. From the properties of orthogonal projection on convex sets, we know that the projection $P_{\Omega_2}(x)$ is the unique solution of the minimization problem $\min_{t\in\Omega_2} f(t)$. If $\|x\|_1 \le s$, the unique solution of $\min_{t\in\Omega_2} f(t)$ is x. If $\|x\|_1 > s$, for arbitrary $t \in \Omega_2$ (i.e., $\|t\|_1 < s$) with $\|x\|_1 > s$, then

$$\|x\|_1 = \|x - t + t\|_1 \le \|x - t\|_1 + \|t\|_1,$$

which implies that

$$\|x - t\|_1 \ge \|x\|_1 - \|t\|_1 \ge \|x\|_1 - s.$$

The above inequality indicates that for arbitrary $t \in \Omega_2$ with $\|x\|_1 > s$, we have

$$\begin{aligned} f(t) = \|x - t\|_1 \ge \|x\|_1 - s &= \frac{1}{\|x\|_1}(\|x\|_1 - s)\|x\|_1 \\ &= \frac{1}{\|x\|_1}\|\|x\|_1 x - sx\|_1 = \|x - \frac{s}{\|x\|_1}x\|_1 = f(\frac{s}{\|x\|_1}x), \end{aligned}$$

which indicates that when $t = \frac{s}{\|x\|_1}x$, $f(t)$ attains its minimum. Therefore, we get the projection formula of x on the convex set Ω_2 as

$$P_{\Omega_2}(x) = \begin{cases} x, & \text{if } \|x\|_1 \le s, \\ \frac{s}{\|x\|_1}x, & \text{if } \|x\|_1 > s. \end{cases}$$

The proof is completed. □

Finally, we present the explicit computational formula for the circumcenter. Let $x_1, x_2, \ldots, x_m$ be vectors in real Hilbert space ($\mathbb{M}$). Let $C = \{x_1, \ldots, x_m\}$. Obviously,

$$\text{circumcenter}\{x_1\} = x_1$$

and

$$\text{circumcenter}\{x_1, x_2\} = \frac{x_1 + x_2}{2}.$$

Now let us assume $m \ge 3$. The explicit computational formula is given by the following lemma.

Lemma 3.3 ([32]) *Let $x_1, x_2, \ldots, x_m$ be affinely independent. Then the* circumcenter$(C) \in \mathbb{M}$ *is the unique point satisfying the following two conditions*

(i) circumcenter$(C) \in$ affine(C), *and*

(ii) $\{\|\text{circumcenter}(C) - x_i\| | x_i \in C, i = 1, 2, \ldots, m\}$ *is a singleton.*

Moreover,

$$\text{circumcenter}(C) = x_1 + \frac{1}{2}(x_2 - x_1, \ldots, x_m - x_1)G(x_2 - x_1, \ldots, x_m - x_1)^{-1} \begin{pmatrix} ||x_2 - x_1||^2 \\ \vdots \\ ||x_m - x_1||^2 \end{pmatrix},$$

where $G(x_2 - x_1, \ldots, x_{m-1} - x_1, x_m - x_1)$ *is the Gram matrix*

$$G(x_2 - x_1, \ldots, x_{m-1} - x_1, x_m - x_1) = \begin{pmatrix} ||x_2 - x_1||^2 & \langle x_2 - x_1, x_3 - x_1\rangle & \ldots & \langle x_2 - x_1, x_m - x_1\rangle \\ \vdots & \vdots & & \vdots \\ \langle x_{m-1} - x_1, x_2 - x_1\rangle & \langle x_{m-1} - x_1, x_3 - x_1\rangle & \ldots & \langle x_{m-1} - x_1, x_m - x_1\rangle \\ \langle x_m - x_1, x_2 - x_1\rangle & \langle x_m - x_1, x_3 - x_1\rangle & \ldots & ||x_m - x_1||^2 \end{pmatrix}.$$

Algorithm 1 cCRM algorithm for solving the problem (1.2)

Input: Initial value $x^0 \in \mathbb{R}^n$, sensing matrix $H \in \mathbb{R}^{m\times n}$, measurement vector $y \in \mathbb{R}^m$, positive number s and the error tolerance ε

Output: the solution x^* of the problem (1.2)

1: $x^k \leftarrow x^0$
2: **while** $e_k > \varepsilon$ **do**
3: Calculate centralized point x_c^k
3.1: $P_{\Omega_1} = x^k + H^+(y - Hx^k)$
3.2: Calculate the projection $x_{\text{AP}} = P_{\Omega_2}P_{\Omega_1}$ by (3.1)
3.3: $P_{\Omega_1}(x_{\text{AP}}) = x_{\text{AP}} + H^+(y - Hx_{\text{AP}})$
3.4: $x_c^k = \frac{1}{2}(x_{\text{AP}} + P_{\Omega_1}(x_{\text{AP}}))$
4: Calculate the reflection $R_{\Omega_1}(x_c^k) = 2P_{\Omega_1}(x_{\text{AP}}) - x_c^k$ on the set Ω_1
5: Calculate the projection $P_{\Omega_2}(x_c^k)$ by (3.1)
6: Calculate the reflection $R_{\Omega_2}(x_c^k) = 2P_{\Omega_2}(x_c^k) - x_c^k$ on the set Ω_2
7: Calculate $x^{k+1} = \text{circumcenter}\{x^k, R_{\Omega_1}(x_c^k), R_{\Omega_2}(x_c^k)\}$ according to (3.2)
8: $e_k = \|x^{k+1} - x^k\|$
9: **end while**

Making use of Lemma 3.3, we can also derive an explicit computational formula for the circumcenter$\{x, R_{\Omega_1}(x), R_{\Omega_2}(x)\}$ as follows

$$\begin{aligned} &\text{circumcenter}\{x, R_{\Omega_1}(x), R_{\Omega_2}(x)\} \\ &= x + \frac{1}{2}(R_{\Omega_1} - x, R_{\Omega_2} - x)G(R_{\Omega_1} - x, R_{\Omega_2} - x)^{-1} \begin{pmatrix} \|R_{\Omega_1} - x\|^2 \\ \|R_{\Omega_2} - x\|^2 \end{pmatrix}, \end{aligned} \tag{3.2}$$

where

$$G(R_{\Omega_1}-x,R_{\Omega_2}-x)=\begin{pmatrix} ||R_{\Omega_1}-x||^2 & \langle R_{\Omega_1}-x,R_{\Omega_2}-x\rangle \\ \\ \langle R_{\Omega_2}-x,R_{\Omega_1}-x\rangle & ||R_{\Omega_2}-x||^2 \end{pmatrix}.$$

The centralized circumcentered-reflection method (2.3) to solve the problem (1.2) can be summarized as in Algorithm 1. In terms of the calculation formula for the circumcenter of the cCRM, computing the centralized point x_c requires three orthogonal projections. Additionally, calculating circumcenter$\{x^k, R_{\Omega_1}(x_c^k), R_{\Omega_2}(x_c^k)\}$ necessitates two orthogonal projections. Therefore, completing one iteration of the cCRM entails a total of five orthogonal projections. However, in Algorithm 1, only four orthogonal projections are provided. This is because $x_c=\frac{1}{2}(x_{\rm AP}+P_{\Omega_1}(x_{\rm AP}))$ indicates that x_c lies between $x_{\rm AP}$ and $P_{\Omega_1}(x_{\rm AP})$, which implies that $P_{\Omega_1}(x_{\rm AP})=P_{\Omega_1}(x_c)$. Now we will give the numerical analysis for Algorithm 1.

(1) Computational complexity

Algorithm 1 illustrates that Steps 3.1 to 3.4 are designed to obtain the centralized point. Steps 4–6 are conducted to obtain the other two vertices of the triangle. Step 7 is aimed at calculating the circumcenter of the triangle. The computational cost of each step is evaluated in Tables 1 and 2.

Step	Step 3.1	Step 3.2	Step 3.3	Step 3.4
Complexity	$m^2n+mn+mn+n$	n	$m^2n+mn+mn+n$	n

Table 1 Computational cost of Steps 3.1–3.4

Step	Step 4	Step 5	Step 6	Step 7
Complexity	n	n	n	n

Table 2 Computational cost of Steps 4–7

The computational complexity of Algorithm 1 primarily arises from the calculation of H^+, with a complexity of $O(m^2n)+O(m^3)+O(nm^2)$. Since $m<n$, the complexity of calculating H^+ is $O(m^2n)$. Therefore, the total computational cost of Algorithm 1 is $O(k(2nm^2+4mn+8n))$, where k is the total number of iteration step. The computational cost of Algorithm 1 is determined by $O(k(nm^2))$.

(2) Initial value and stopping criteria

We choose $x_0=H^T*y$ as the initial point and choose the $\|x^{k+1}-x^k\|<\varepsilon$ as the stopping criteria, where ε is a small enough parameter.

Finally, we perform the convergence analysis of Algorithm 1 .

Theorem 3.4 *If the convex feasibility problem (1.2) has a solution, then the sequence $\{x^k\}$ generated by Algorithm 1 converges to an element $x^*\in\Omega_1\cap\Omega_2$, which is the solution of the problem (1.2).*

Proof It is obvious that the set Ω_1 involved in Algorithm 1 is an affine set and the set Ω_2 is

a closed convex set. If the problem (1.2) has a solution, that is to say, the intersection of sets Ω_1 and Ω_2 is nonempty. Therefore, we can obtain from Theorem 2.3 that the sequence $\{x^k\}$ generated by Algorithm 1 converges to an element $x^* \in \Omega_1 \cap \Omega_2$, which is the solution of the problem (1.2). □

4. Numerical experiments

In this section, we first present a simple numerical experiment to demonstrate the feasibility of Algorithm 1 to solve the problem (1.2). Subsequently, we compare Algorithm 1 with other commonly used algorithms for addressing the problem (1.2). Finally, we recover two-dimensional natural images by using Algorithm 1. All codes are written by version of Matlab 2019a and performed on a Windows 11 PC with Intel(R) Core(TM) i7-8550U CPU @ 1.80GHz 1.99 GHz and 8GB of memory.

The performance of reconstruction algorithm is closely related to some properties of the measurement matrix H. To accurately recover the original signal x from the observation vector y, the observation matrix H must satisfy the RIP. In our experiments, we use the Gaussian matrix as the measurement matrix.

Example 4.1 In signal processing, compressed sensing is a powerful technique for recovering sparse signals from a limited number of measurements. Specifically, an n-sample original signal x with k nonzero components can be accurately recovered from $k \ll m < n$ measurements. Recovering x from m measurements requires solving an underdeterminated system of linear equations

$$y = Hx + e,$$

where $y \in R^m$ is the vector of noisy observations or measurements, $x \in R^n$ with k non-zero elements is that needs to be restored, and e denotes the noise, with each entry drawn from the Gaussian distribution $N(0, \sigma^2)$. Since the sampling may be subject to errors in practical situations, we add Gaussian white noise here to mimic this effect. Hence, the recovery of x can be achieved by solving the problem (1.2).

In our experiment, we consider the problem (1.2) with the original signal x being generated as $x = \text{sprandn}(n, 1, k)$ and measurement matrix H is constructed as $H = \text{randn}(m, n)$, and the observed value $y = Hx + e$. We assess the quality of reconstruction by means of squared error (MSE) to the original signal x, i.e.,

$$MSE = \frac{1}{n}\|x - x^*\|,$$

where x^* denotes the recovery signal. The stopping criterion for Example 4.1 is $\|x^{k+1} - x^k\| < 10^{-6}$. For the experimental setup, we choose $n = 2560$, $m = \frac{n}{4}$, $k = 0.05n$ (k denotes sparsity level). The experimental results are presented in Figure 3.

In Figure 3, the recovered signal is represented by red asterisks "$*$" and the original signal is denoted by blue circles "$\circ$". Despite the presence of noise, we can see that these two signals

can match very well, with an MSE of only 0.0052897‰. It is shown that Algorithm 1 is really feasible and can accurately recover the original signal.

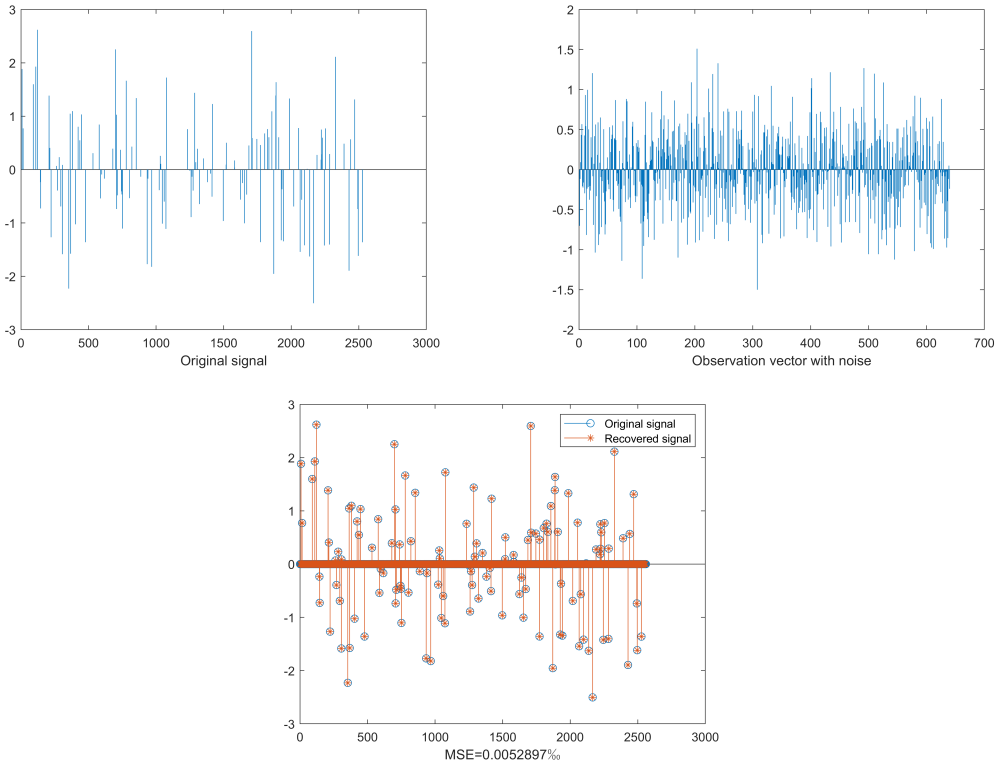


Figure 3 The recovery results for Example 4.1

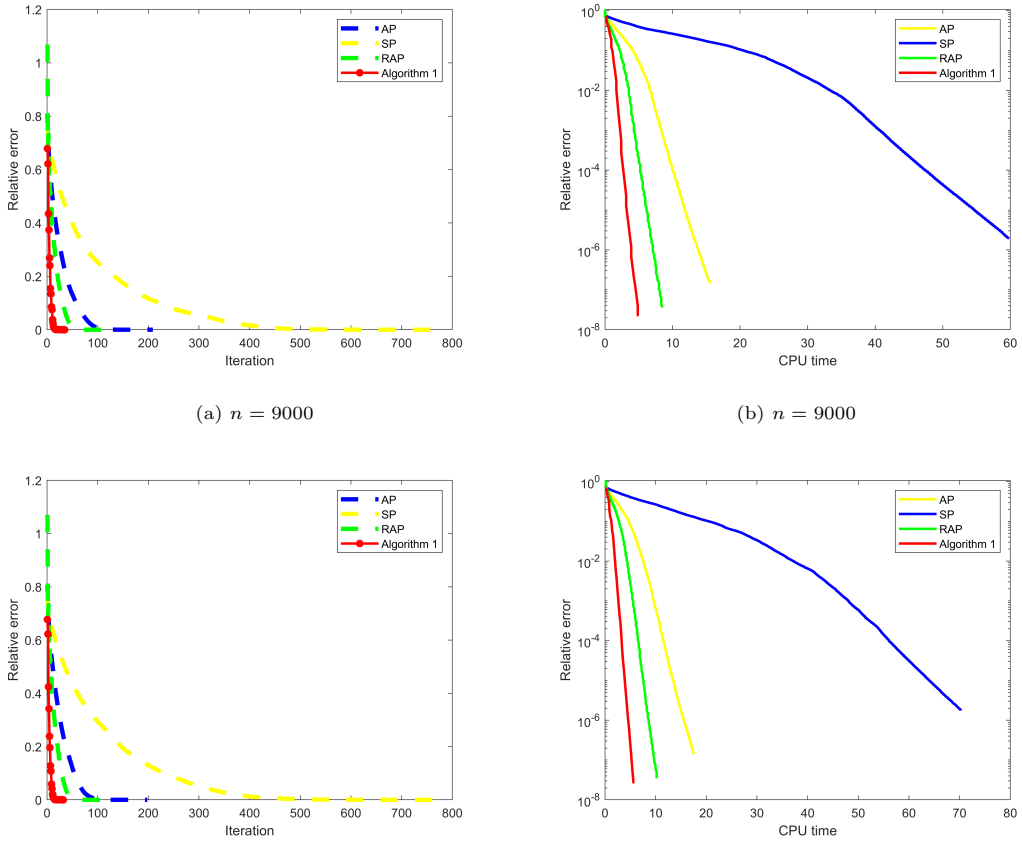
Example 4.2 We consider the problem (1.2) with measurement matrix $H = \text{randn}(m, n)$. The sampling rate is set to 25%, i.e., $m = \frac{n}{4}$. The number of non-zero elements in the vector x is $k = 0.05n$. The original signal x and the observation vector y are generated in the same manner as in the previous example. We list MSE, CPU time and iterations among the alternating projection algorithm (AP) [21], the simultaneous projection algorithm (SP) [24], the relaxed alternating projection algorithm (RAP) [33] and Algorithm 1 in different dimensions n. This comparison is motivated by the fact that AP, SP and RAP are widely used for solving convex feasibility problem. The experiment is conducted in two scenarios, without noise and with the addition of Gaussian noise. The stopping criterion for Example 4.2 are defined as follows

$$\begin{cases} \|x^{k+1} - x^k\| < 10^{-2}, & \text{noise ,} \\ \|x^{k+1} - x^k\| < 10^{-6}, & \text{no-noise.} \end{cases}$$

The experimental results are shown in Table 3 and Figure 4.

From Table 3, it is demonstrated that across various dimensions, Algorithm 1 and the other three methods can effectively solve the problem (1.2). However, Algorithm 1 consistently exhibits superior performance. Regardless of the presence or absence of added noise, Algorithm 1

outperforms the other three algorithms in terms of the computing time, the number of iterations, and the discrepancy between the recovered solution and the exact solution.



(a) $n = 9000$ (b) $n = 9000$

(c) $n = 10000$ (d) $n = 10000$

Figure 4 Comparative results for Example 4.2

To provide a more direct visualization of the superior convergence rate of our algorithm, we present the relationship between the relative error of Algorithm 1 and AP, SP, RAP of the same iteration type over time and the number of iterations. The dimensions of the tests are 9000 and 10000 respectively. The results are shown in Figure 4.

Figure 4 illustrates that Algorithm 1 converges to the solution more rapidly than the other three algorithms.

From Table 3 and Figure 4, we can see that in the case of low dimensions, Algorithm 1 has a good performance. As the dimensionality of the recovered signal increases, the recovery speed advantage of Algorithm 1 becomes more obvious. The most significant aspect is that as the dimensionality of the problem increases, the computational time of Algorithm 1 grows at a relatively slow rate. This characteristic makes Algorithm 1 particularly suitable for addressing some larger scale problems in signal recovery.

		No-noise			Noise		
Dimension	Tested method	Iter	CPU time	MSE	Iter	CPU time	MSE
1000	AP	197	0.125	3.2063e-09	104	0.0625	2.5479e-05
	SP	801	0.70313	4.4347e-08	174	0.20313	0.0010279
	RAP	116	0.0625	9.1181e-10	68	0.14063	6.8156e-06
	Algorithm 1	**34**	**0.0625**	**4.0708e-10**	**18**	**0.0625**	**3.6204e-06**
2000	AP	183	0.92188	1.4654e-09	108	0.64063	1.3799e-05
	SP	728	4.2344	2.1405e-08	293	1.8281	0.00043136
	RAP	101	0.85938	3.9106e-10	66	0.39063	3.5751e-06
	Algorithm 1	**31**	**0.54688**	**2.5762e-10**	**18**	**0.21875**	**1.8114e-06**
3000	AP	186	1.9063	9.6667e-10	94	1.0938	9.4811e-06
	SP	794	8.2344	1.3097e-08	234	3.0469	0.00020858
	RAP	101	1.375	2.9209e-10	57	0.85938	2.383e-06
	Algorithm 1	**29**	**0.71875**	**1.6165e-10**	**17**	**0.4375**	**1.4262e-06**
4000	AP	185	4.3594	7.3698e-10	99	1.9844	5.7699e-06
	SP	754	15.4063	9.4057e-09	283	5.7969	0.00020624
	RAP	106	2.2188	1.8573e-10	59	1.3594	1.7161e-06
	Algorithm 1	**31**	**1.2969**	**1.4138e-10**	**18**	**0.9375**	**1.0842e-06**
5000	AP	199	5.7188	6.0396e-10	105	2.9375	5.2399e-06
	SP	720	21.0625	8.1544e-09	343	9.2656	0.00010738
	RAP	111	3.2813	1.5663e-10	62	1.7500	1.415e-06
	Algorithm 1	**31**	**1.8594**	**1.0721e-10**	**18**	**1.0000**	**1.0063e-06**
6000	AP	207	8.5781	5.2029e-10	114	4.625	4.6062e-06
	SP	759	31.4219	6.9555e-09	302	11.6875	8.0981e-05
	RAP	119	4.8906	1.4942e-10	68	2.6719	1.0889e-06
	Algorithm 1	**34**	**2.7813**	**8.2543e-11**	**18**	**1.375**	**9.0137e-07**
7000	AP	188	10.4063	4.1429e-10	105	5.6406	3.4932e-06
	SP	761	41.4375	5.8038e-09	303	15.5469	5.8159e-05
	RAP	105	5.6719	1.0308e-10	63	3.1563	9.1602e-07
	Algorithm 1	**31**	**3.3906**	**5.8568e-11**	**19**	**2.0469**	**7.0292e-07**

Table 3 Compare iteration, CPU time and MSE among AP, SP, RAP and Algorithm 1 in different dimensions n

Example 4.3 We formulate the sparse image restoration problem as the problem (1.2), using a measurement matrix $H = \text{randn}(m, n)$, where the parameters m, n are determined by the dimensions of the image to be restored. The two-dimensional natural images are chosen as the original signal. We also compare Algorithm 1 with three other widely used image restoration algorithms, including the orthogonal matching pursuit (OMP) [34], the alternating projection algorithm (AP) [35], and the basis pursuit (BP) [36]. The size of image is set to 512×512 and Gaussian random matrix is utilized as the measurement matrix. To represent the sparse image,

we apply the discrete wavelet transform (DWT). The parameters are set as $n = 512$, $m = \rho \cdot n$ and $\sigma = 0$, where ρ denotes the sampling rate. The algorithms are executed with a stopping criterion of $\|y - Hx^k\| \leq 10^{-3}$.

Tested method	Algorithm 1	OMP	AP	BP
CPU time	10.5781s	58.8281s	22.125s	219.1563s
SNR	48.2151dB	48.2128dB	48.2129dB	48.2151dB

Table 4 Compare CPU time and SNR among Algorithm 1, OMP, AP and BP in image recovery

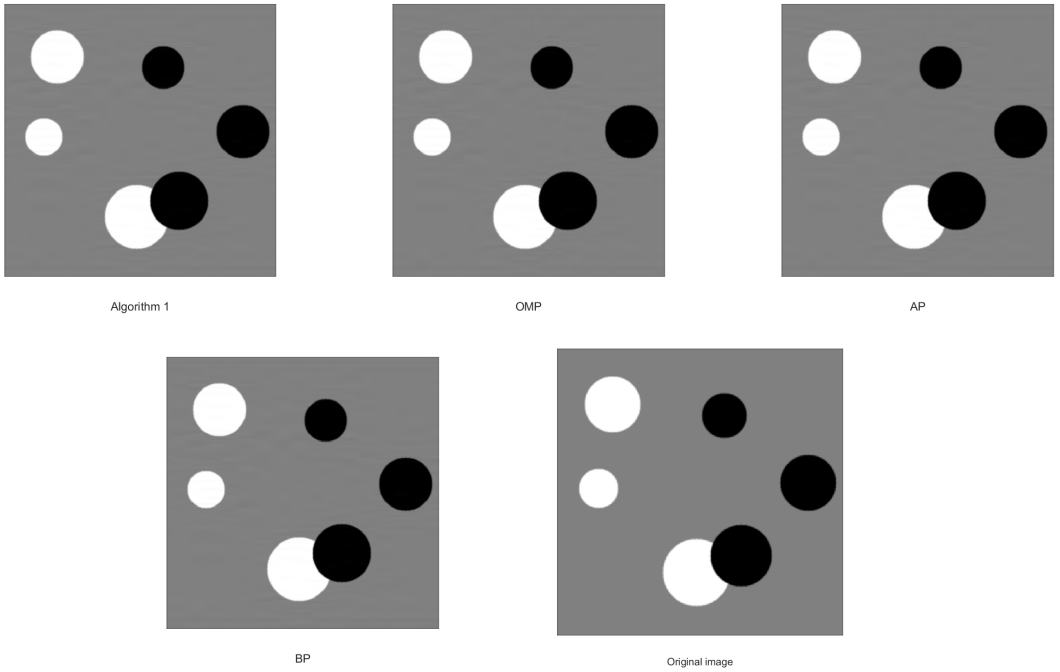


Figure 5 Original signal, recovered signal by different algorithms under the Gaussian sampling of 58.5%

To evaluate the reconstruction quality of the algorithms, we introduce the Signal-to-Noise Ratio (denoted by SNR), calculated using the following formula

$$SNR = 10 \log_{10} \frac{\|x\|_2^2}{\|x - x^*\|_2^2},$$

where x and x^* denote the true signal and the recovered signal, respectively.

In the following experiment, we select the "circlesBrightDark" image (512×512) as the test image and conduct a comparison of Algorithm 1 with OMP, AP and BP in terms of computational time and the relative error. The experimental results, presented in Figure 5 and Table 4, reveal that Algorithm 1 and the other algorithms can accurately restore images, but Algorithm 1 is slightly better. Furthermore, compared to the other three algorithms, Algorithm 1 takes the least computing time. Therefore, Algorithm 1 converges faster than the other three algorithms.

5. Conclusion

In this paper, we consider the convex feasibility problem (1.2), which arises in sparse signal recovery and compressed sensing. However, the original compressed sensing or sparse recovery problem does not conform to the standard convex feasibility framework. To bridge this gap, we approximate the l_0-norm with the l_1-norm, thereby transforming the sparse recovery problem into a convex feasibility problem. The centralized circumcentered-reflection method is designed to solve this convex feasible problem. In comparison with other algorithms, Algorithm 1 achieves faster convergence. Furthermore, in the context of sparse image restoration, Algorithm 1 exhibits significant advantages in terms of computational efficiency.

Acknowledgements We thank the editor and the referees for providing useful suggestions to improve this paper.

Conflict of Interest The authors declare no conflict of interest.

Availability of Data and Material The datasets used and analysed during the current study are synthetic and their generation procedures are described within the article.

References

[1] E. J. CANDES, T. TAO. *Decoding by linear programming*. IEEE Trans. Inform. Theory, 2005, **51**(12): 4203–4215.

[2] M. KHOSRAVY, N. GUPTA, N. PATEL, et al. *Recovery in Compressive Sensing: a Review*. Compressive Sensing in Healthcare, 2020.

[3] I. AHMED, A. KHALIL, I. AHMED, et al. *Sparse signal representation, sampling, and recovery in compressive sensing frameworks*. IEEE Access, 2022, **10**: 85002–85018.

[4] Xueping CHEN, Jianzhong LIU, Jiandong CHEN. *A new result on recovery sparse signals using orthogonal matching pursuit*. Stat. Theory Relat. Fields, 2022, **6**(3): 220–226.

[5] B. K. NATARAJAN. *Sparse approximate solutions to linear systems*. SIAM J. Comput., 1995, **24**(2): 227–234.

[6] Zhe WANG, Shixi WEN, Guojie LI, et al. *Improved sparse signal reconstruction algorithm based on smoothed ℓ_0 norm*. In: 2023 3rd Int. Symp. Comput. Technol. Inf. Sci. (ISCTIS), 2023, 963–966.

[7] J. W. CHINNECK. *The constraint consensus method for finding approximately feasible points in nonlinear programs*. INFORMS J. Comput., 2004, **16**(3): 255–265.

[8] A. J. ZASLAVSKI. *Subgradient projection algorithms and approximate solutions of convex feasibility problems*. J. Optim. Theory Appl., 2013, **157**(3): 803–819.

[9] T. L. N. NGUYEN, Y. SHIN. *Indoor wireless localization via convex feasibility problem*. In: 2014 Int. Conf. Indoor Position. Indoor Navig. (IPIN), 2014, 152–156.

[10] M. R. GHOLAMI, H. WYMEERSCH, E. G. STROM, et al. *Wireless network positioning as a convex feasibility problem*. EURASIP J. Wirel. Commun. Netw., 2011, 1–15.

[11] Biao SONG, Jian SUN, Hailong WANG, et al. *Convex feasibility problem based geometric approach for device-free localization*. In: 2017 20th Int. Conf. Inf. Fusion (Fusion), 2017, 1–7.

[12] Xiaopeng ZHAO, M. A. KOBIS. *On the convergence of general projection methods for solving convex feasibility problems with applications to the inverse problem of image recovery*. Optimization, 2018, **67**(9): 1409–1427.

[13] J. VON NEUMANN. *Functional Operators. II. The Geometry of Orthogonal Spaces*. Princeton University Press, Princeton, NJ, 1950.

[14] H. H. BAUSCHKE, J. M. BORWEIN. *On projection algorithms for solving convex feasibility problems*. SIAM Rev., 1996, **38**(3): 367–426.

[15] Y. CENSOR. *Row-action methods for huge and sparse systems and their applications*. SIAM Rev., 1981, **23**(4): 444–466.

[16] G. CROMBEZ. *Solving convex feasibility problems by a parallel projection method with geometrically-defined parameters*. Appl. Anal., 1997, **64**(3-4): 277–290.

[17] Y. CENSOR, A. SEGAL. *On the string averaging method for sparse common fixed-point problems*. Int. Trans. Oper. Res., 2009, **16**(4): 481–494.

[18] Yazheng DANG, Yan GAO, Liping ZHI. *Block-iterative subgradient projection algorithms for the convex feasibility problem*. OR Trans, 2011, **15**(1): 59–70.

[19] A. CEGIELSKI, N. NIMANA. *Extrapolated cyclic subgradient projection methods for the convex feasibility problems and their numerical behaviour*. Optimization, 2019, **68**(1): 145–161.

[20] Yazheng DANG, Yan GAO. *Non-monotonous accelerated parallel subgradient projection algorithm for convex feasibility problem*. Optimization, 2014, **63**(4): 571–584.

[21] R. ESCALANTE, M. RAYDAN. *Alternating Projection Methods*. SIAM, Philadelphia, 2011.

[22] N. BUONG, P. V. SON. *An explicit iteration method for convex feasibility problems in Hilbert spaces*. Appl. Math. Sci., 2008, **2**(15): 725–734.

[23] A. CEGIELSKI. *Projection Methods*. In: *Iterative Methods for Fixed Point Problems in Hilbert Spaces*, Springer, Berlin, Heidelberg, 2012, 203–274.

[24] A. R. DE PIERRO, A. N. IUSEM. *A simultaneous projections method for linear inequalities*. Linear Algebra Appl., 1985, **64**: 243–253.

[25] F. J. ARAGON ARTACHO, R. CAMPOY, M. K. TAM. *The Douglas-Rachford algorithm for convex and nonconvex feasibility problems*. Math. Methods Oper. Res., 2020, **91**: 201–240.

[26] R. D. MILLAN, O. P. FERREIRA, L. F. PRUDENTE. *Alternating conditional gradient method for convex feasibility problems*. Comput. Optim. Appl., 2021, **80**(1): 245–269.

[27] R. D. MILLAN, O. P. FERREIRA, J. UGON. *Approximate Douglas-Rachford algorithm for two-sets convex feasibility problems*. J. Global Optim., 2023, **86**(3): 621–636.

[28] I. NECOARA, A. NEDIC. *Minibatch stochastic subgradient-based projection algorithms for feasibility problems with convex inequalities*. Comput. Optim. Appl., 2021, **80**(1): 121–152.

[29] I. NECOARA. *Stochastic block projection algorithms with extrapolation for convex feasibility problems*. Optim. Methods Softw., 2022, **37**(5): 1845–1875.

[30] H. H. BAUSCHKE, M. K. LAL, Xianfu WANG. *Directional asymptotics of Fejér monotone sequences*. Optim. Lett., 2023, **17**(3): 531–544.

[31] R. BEHLING, Y. BELLO-CRUZ, A. N. IUSEM, et al. *On the centralization of the circumcentered-reflection method*. Math. Program., 2024, **205**(1-2): Ser. A, 337–371.

[32] H. H. BAUSCHKE, Hui OUYANG, Xianfu WANG. *On circumcenters of finite sets in Hilbert spaces*. Linear Nonlinear Anal., 2018, **4**(2): 271–295.

[33] A. CEGIELSKI, R. DYLEWSKI. *Variable target value relaxed alternating projection method*. Comput. Optim. Appl., 2010, **47**(3): 455–476.

[34] Quan SUN, Feiyun WU, Kunde YANG, et al. *Sparse signal recovery from noisy measurements via searching forward OMP*. Electron. Lett., 2022, **58**(3): 124–126.

[35] Tao SUN, Peibing DU, Lizhi CHENG, et al. *Alternating projection for sparse recovery*. IET Signal Process., 2017, **11**(2): 135–144.

[36] T. RAHIM, R. MAGDALENA, L. NOVAMIZANTI, et al. *Basis pursuit with sparsity averaging for compressive sampling of iris images*. IEEE Access, 2022, **10**: 13728–13737.