

Signal Reconstruction Algorithm Application Research under Compressed Sensing in Sparse Signal Reconstruction

Shimin Zhang

College of Artificial Intelligence and Big Data, Zibo Polytechnic University, Zibo 255000, China

E-mail: zhangsm1104@163.com

Keywords: compression perception, signal reconstruction, optimization-oriented, support set estimation, decentralization

Received: May 19, 2025

To improve the efficiency of compressed sensing sparse signal reconstruction, a reconstruction algorithm suitable for different scenarios is proposed. On the basis of greedy algorithm, a sparse reconstruction algorithm for optimization is constructed. A multi-source sparse signal reconstruction algorithm with improved support set estimation is proposed. Experimental data show that the mean square error of the optimized sparse signal reconstruction algorithm is less than 10^{-5} , which is 1-4 orders of magnitude smaller than other comparative algorithms (such as orthogonal matching pursuit). The support set estimation accuracy of the joint sparse signal reconstruction algorithm is the highest. When the signal-to-noise ratio is 10, the relative reconstruction error based on the orthogonal matching tracking algorithm is 0.57. The minimum relative reconstruction error of the proposed joint sparse signal reconstruction algorithm is 0.34. The analysis of experimental data shows that the decentralized joint sparse signal reconstruction algorithm proposed in this paper not only ensures the efficiency of signal reconstruction, but also reduces the computational complexity.

Povzetek: Prispevek se ukvarja s kompresijskim zaznavanjem (CS) in rekonstrukcija redkih signalov. Novost je kombinacija OMP-backtrackinga, glasovanja skupne podpore in enkratne porazdeljene izmenjave za skupno SSR; potrebuje teoretično utrditev, realne podatke in reproducibilnost.

1 Introduction

Compressed Sensing (CS) signal processing technology is a new type of signal processing method that can transform traditional sampling theorems into more efficient signal reconstruction methods and has become a current research hotspot. Compared with traditional sampling theorems, CS technology can greatly reduce the number of samples and improve the efficiency of signal reconstruction. In traditional communication and signal processing frameworks, based on Shannon's theorem, the sampling frequency needs to be twice higher than the highest information frequency [1]. With the development of technology, the amount of data that communication nodes and devices need to process is very large. Traditional signal sampling can generate signal redundancy, leading to a great waste of resources [2]. Therefore, the development of CS technology, with its high sampling rate beyond Nyquist limits, addresses issues such as oversampling, data compression, and data loss, while also ensuring reliable data recovery [3]. CS technology can represent signals using only a small number of non-zero coefficients within an appropriate range, potentially reducing the cost of sampling and computation. It is widely used in various fields such as electronic technology and computer science [4]. In Sparse Signal Reconstruction (SSR), CS technology can achieve SSR through iterative algorithms such as greedy algorithms. Meanwhile, in distributed network scenarios, it is necessary to consider

the processing of multi-source signals and utilize the correlation between data to complete joint SSR [5]. The current signal reconstruction approach fails to adequately leverage the inherent sparsity of the signal, leading to suboptimal reconstruction outcomes when handling sparse signals. Therefore, research has proposed joint SSR algorithms for single-sources and multi-sources, aiming to improve the reconstruction effect. The study includes four sections. The first section is a summary of relevant research, the second section is the design of SSR algorithms, the third section is the performance analysis of the proposed algorithm, and the fourth section is a summary of this study. This study proposes a joint SSR algorithm based on an improved public support set for the multi-source joint SSR problem. In SSR problems, oriented SSR algorithm (OSRA) employs global optimization techniques, encompassing support set pre-selection with backtracking and comprehensive global optimization, thereby exhibiting robust global search capabilities. Centralized joint SSR algorithm (CJSRA) focuses on improving the estimation of the public support set, by preliminarily estimating the public support set, optimizing the public support set, and using edge information-based orthogonal matching pursuit (OMP) algorithm to more accurately estimate the support set of sparse signals, thereby improving the accuracy and stability of signal reconstruction. In the problem of multi-source joint SSR, the combination of these two algorithms

can achieve efficient signal reconstruction, reduce the energy consumption of node transmission and reception of signals, and avoid the problem of repeated reception and transmission of signals.

The novelty of the proposed method in the public support set estimation of joint SSR lies in that CJSRA improves the support set estimation process of the traditional joint sparse algorithm through the voting strategy and error screening mechanism. Most of the existing methods rely on the direct intersection of single-source estimation and fail to effectively filter out random errors. However, CJSRA improves the reliability of the initial estimation through multi-source voting and further optimizes the support set through edge information OMP, solving the dependence problem of traditional methods on high observation rates. DJSRA proposes a distributed single information interaction mechanism, breaking through the limitation that existing distributed algorithms require multiple communications. The public support set is estimated through the collaborative estimation of neighborhood nodes without the coordination of a central node, which ADAPTS to the dynamic network topology. Moreover, the support set estimation process is compatible with the correlation of partially overlapping signals instead of forcing the assumption of complete sharing. Compared with the existing methods, the method proposed in this paper demonstrates better robustness in low observation rates, noise interference and distributed scenarios through hybrid support set modeling (public support set + innovative support set) and adaptive collaborative strategies, rather than merely continuing the concept of public support sets. The contribution of this study is mainly reflected in the following points. Firstly, for the problem of single-source SSR, the key difference between OSRA and traditional greedy algorithms (such as OMP and generalized orthogonal matching pursuit (gOMP)) lies in the introduction of a support set pre-selection with backtracking mechanism and a global optimization objective function. Traditional methods only rely on local search to iteratively update the support set, while OSRA verifies the reliability of the initial estimation through backtracking and exits the local optimum through the global objective function, significantly improving the accuracy and robustness of SSR. Secondly, in the aspect of multi-source joint SSR, the study proposes CJSRA and DJSRA based on improved public support set estimation. CJSRA can improve the accuracy of support set estimation, while DJSRA can reduce the computational complexity while ensuring the signal reconstruction effect, and reduce the time consumption of nodes repeatedly sending and receiving information. These algorithms provide more efficient solutions for SSR in different scenarios and enrich the knowledge system in this field. The drawback of OSRA and CJSRA algorithms themselves is that as sparsity increases, the computational complexity of the algorithms also increases accordingly.

2 Related works

CS, also known as sparse sampling, is a technique used in the field of electronic engineering signal processing to

find sparse solutions for under-determined linear systems. It can be used to obtain and reconstruct sparse or compressible signals. Wang et al. proposed an energy-efficient collection method using CS to construct a data collection model for underwater acoustic sensor networks. The results indicated that this method could reduce the sensor energy consumption by 15% [6]. Wang et al. proposed a triple image encryption and hiding algorithm that utilized a two-dimensional discrete wavelet transform to sparsely represent three grayscale images, and processed and compressed the measurement matrix [7]. The results showed that this method could improve encryption efficiency by 23%. Bai et al. proposed a non-convex CS method for detecting fan noise patterns in CS, utilizing the L-1/2 minimization algorithm to optimize the sensor array. Results showed that this method could improve the dynamic range by 10dB [8]. Kato et al. introduced compressive sensing into the human-order analysis to address the limitation that traditional methods are only applicable to steady-state vibrations, enabling the measurement of operational vibrations with rotational speed fluctuations (such as in automotive engines). The aluminum plate vibration experiment and engine application showed that this method could accurately reconstruct the vibration (modal guarantee criterion 0.98, mean square error less than 1%, etc.), was suitable for the full-field measurement of high-speed vibration of rotating machinery, and was expected to assist in signal recovery and data compression [9]. Kadhim et al. aimed at the image encryption requirements in remote working and learning. In their work, a multi-chaotic encryption algorithm based on block CS, Swin Transformer and Mustang optimization was proposed. Encryption was achieved through processes such as wavelet transform, block compression, chaotic diffusion and optimized scrambling. Experiments showed that the average information entropy of the algorithm was 7.9749 and the NPCR reached 99.5453%, etc., indicating that it was efficient and robust and could resist various attacks [10].

CS technology includes three key steps: signal sparse representation, measurement matrix construction, and reconstruction algorithm. Among them, the reconstruction algorithm is the core of compressive sensing technology, as it has a direct impact on the complexity and quality of signal reconstruction. Zeinali et al. proposed a new algorithm for reconstructing wireless sensor network (WSNs) data with spatio-temporal correlation using CS. A time-varying sliding window mechanism was adopted to dynamically adjust the window size and the number of measurements, so as to effectively utilize spatio-temporal correlation, balance the sampling rate and reduce the transmission cost. Simulation showed that this algorithm maximized the utilization of prior signal information through the distributed data window framework. Compared with other CS reconstruction methods, it achieved higher reconstruction accuracy with less transmission volume and was suitable for various WSN scenarios [11]. Zhao et al. proposed a smooth inertial neural dynamics method to reconstruct sparse signals through the processing of norm minimization problems. The research results indicated that under certain

conditions, this method could average sub-linear convergence speed and improve the efficiency of SSR and image restoration by 11.1% [12]. Lin et al. optimized the regularization parameters and iteration times under different sparsity ratios to address the issue of reduced quality in sparse imaging reconstruction. Experimental data showed that the quality of SSR using the best parameters was improved by 16.7% [13]. Sun et al. proposed an improved OMP method that combined a forward search strategy to find residual errors. Results showed that this method could improve the Signal-to-Noise Ratio (SNR) of signal reconstruction and reduce the computational complexity of measurement [14]. Li et al. proposed a variable step-matching tracking CS algorithm based on oblique projection, which estimated initial sparsity using the constrained isometry property and created a support set for the target signal. The results showed that this method was superior to traditional matching tracking algorithms and could reduce computational complexity by 13% [15]. Bayesian Compressive Sensing (BCS) is a signal acquisition and processing technique based on Bayesian theory. It is a new technology developed in the field of signal processing in recent years, with the basic idea of compressing signals before sampling and digitization. Liu et al. proposed a new method based on Bayesian CS algorithm for spectrum estimation of multiple frequency hopping signals with randomly omitted observations. By designing a specific bilinear time-frequency representation framework with a time-frequency kernel, the artifacts caused by cross terms and missed observations are effectively suppressed, while preserving the self-terms of the frequency hopping signal. By utilizing the redesigned structure aware Bayesian CS algorithm to process the kernels in the time-frequency domain, high-resolution frequency hopping signal spectrum estimation was achieved even in the absence of most data observations. The simulation results showed that the method was effective [16].

To sum up, many researchers have conducted extensive studies and designs on compressive sensing and SSR. However, traditional greedy reconstruction algorithms (such as OMP and segmented orthogonal matching pursuit (StOMP)) only update the support set through local iteration, which has the defects of overfitting and short-sightedness, and does not introduce a global optimization mechanism to escape the local optimum. The OSRA proposed in this paper compensates for the limitations of traditional algorithms in support set estimation through the support set preselection backtracking and global search strategies. The existing joint SSR algorithms (such as inner-outer support set pursuit algorithm (IOSSP) and edge information greedy pursuit algorithm (SIPP)) do not estimate the public support set of multi-source signals accurately enough and fail to make full use of the correlation between signals for support set optimization. In this paper, CJSRA improves the preliminary estimation and screening process of the public support set through the voting strategy and the OMP algorithm based on edge information, and enhances the accuracy of multi-source signal support set estimation. Distributed joint SSR algorithms (such as distributed

parallel pursuit algorithm (DIPP)) rely on multiple round-trip communications between nodes to complete signal reconstruction, resulting in high energy consumption and delay. In this paper, DJSRA eliminates the reliance on the central node through a single information interaction and an improved public support set estimation method, and reduces the communication cost and computational complexity between nodes. Furthermore, Model-Based CS improves the reconstruction performance by explicitly utilizing the structured sparse priors of the signal (such as group sparsity and tree sparsity), but its dependence on the preset structure limits its applicability in multiple scenarios. BCS models sparsity through hierarchical priors and performs well at low observation rates, but it has high computational complexity and requires preset signal distributions. Although structural sparsity methods (such as the Joint Sparse Model (JSM)) can capture the correlations of multi-source signals, most of them assume that the support sets are completely shared and are difficult to adapt to actual scenarios with partial overlap or dynamic changes. The hybrid support set model and the improved support set estimation method proposed in this paper do not require strict prior assumptions. They can model the correlation of multi-source signals more flexibly and reduce the computational and communication overhead through a distributed collaborative mechanism at the same time.

3 Design of SSR Algorithm under CS

This chapter mainly proposes the design of an optimization-oriented SSR algorithm and a CJSRA and decentralized joint SSR algorithm (DJSRA). The OSRA improves the performance of the signal reconstruction algorithm by improving the greedy algorithm support set estimation. The CJSRA improves the estimation of the public support set to solve the problem of correlation between multi-sources. The DJSRA eliminates the presence of central nodes through information transmission and reconstruction between nodes, thereby improving energy consumption and latency issues.

3.1 Design of SSR algorithm using optimization orientation

The core issue of CS is SSR, which includes single-source and multi-source joint SSR. The signal reconstruction problem is described as equation (1).

$$\min \|x\|_0 \text{ subject to } \|y - \Phi x\| \leq \delta \quad (1)$$

In equation (1), the original signal is denoted as x , the measurement matrix as Φ , and the signal observation vector as y . The reconstruction threshold is set at δ , with the stipulation that the noisy energy value must be less than this reconstruction threshold. The noise observation model estimates signals based on observation data, where the observation data is affected by noise interference. This model attempts to restore certain features of the original signal by considering the influence of noise, in order to improve the accuracy and reliability of signal reconstruction. The original signal is observed through a

measurement matrix and is affected by noise. The reconstruction goal is to recover the original signal based on the observation vector, and the reconstruction threshold is used to determine whether noise can be used for signal reconstruction. The correlation between different variables is shown in that the measurement matrix converts the original signal into the observation vector, and the noise affects the reconstruction effect. The reconstruction threshold is used to judge whether the noise can be ignored, so as to improve the accuracy of the signal reconstruction. The main difference of sparse signal recovery algorithms lies in their utilization of signal sparsity during the solving process. Compared with other algorithms such as greedy methods, they can better utilize the sparse characteristics of signals, thereby achieving faster convergence speed and higher recovery accuracy under the same conditions.

A cornerstone of SSR algorithms lies in the greedy tracking algorithm, which iteratively identifies the current optimal solution. However, it falls short of ensuring a globally optimal solution. In contrast, tree search matching and tracking algorithms can find the optimal branch through a large number of searches, but their computational complexity and complexity are relatively high. Therefore, based on greedy algorithms, this study proposes an OSRA, whose core problem is the SSR algorithm. Over-fitting refers to the phenomenon that the model over-fits the training data during machine learning, leading to poor performance on new unseen data. In the signal reconstruction problem, over-fitting may lead to an excessive dependence of the reconstruction results on the training data, thus creating errors in processing new signals. Compared with traditional greedy tracking algorithms, OSRA proposes a new approach based on global optimization, which solves the problem of over-fitting that may occur in greedy algorithms through support set pre-selection with backtracking and global optimization methods, thereby improving the accuracy and stability of signal reconstruction [17]. Before designing the algorithm, some symbol definitions based on the problem planning are first proposed, and the error function is defined as shown in equation (2).

$$r(\Lambda) = \|y - \Phi_{\Lambda} \hat{x}_{\Lambda}\|_2 \quad (2)$$

In equation (2), the error function is $r(\Lambda)$, and the optimal element composition support set is Λ . The sparse

reconstruction model is grounded in greedy algorithms and endeavors to identify the optimal support set, which comprises key elements, through the definition of error functions. This approach is designed to achieve efficient sparse reconstruction tailored to specific problems. In the process of support set estimation, greedy algorithms may over-fit, leading to support set estimation errors [18]. To address this issue, improvements were made to the greedy algorithm to enhance the performance of the signal reconstruction algorithm. The optimized OSRA flow is shown in Figure 1.

The optimized OSRA mainly includes two parts: support set pre-selection with backtracking and global optimization method. In the stage of pre-selection and backtracking of the support set, the greedy algorithm is first used to obtain the initial estimate of the support set. To avoid over-fitting, it is necessary to backtrack the initial estimated support set. If the error function after backtracking is still below the preset threshold value, the backtracking support set will be used as the final estimate. On the contrary, it will enter the global optimization step. Global optimization is mainly achieved through two stages: local search and global search. Local search mainly reduces the error function, which is the objective function, by estimating the support set that satisfies the conditions. If the conditions are met, then this support set estimate is considered the final estimate. Should the local search fail to locate an estimate of the supporting set that satisfies the given conditions, it will transition to the global search phase. During the global search, a global objective function will be devised to enable the algorithm to escape local optima and ultimately derive the final estimate of the support set. After obtaining the support set estimation, the least squares method is used to achieve signal estimation. Overall, support set pre-selection with backtracking can reduce computational complexity through greedy algorithms, while backtracking can test the support set, identify the best elements to form the support set, and calculate the corresponding error function. The domain definition of the support set is shown in equation (3).

$$\begin{aligned} N(\Lambda) = & \{\Lambda, \Lambda \cup \{\theta_j\} - \{\lambda_j\} : 1 \leq i \leq K, \\ & 1 \leq j \leq N - K\} \end{aligned} \quad (3)$$

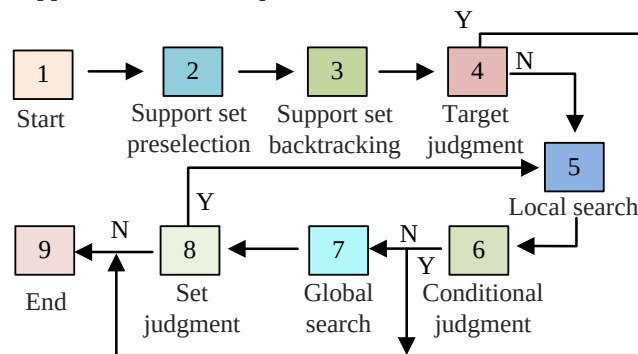


Figure 1: Optimized OSRA flow.

In equation (3), the domain of the support set is defined as $N(\Lambda)$, the number of optimal elements is K , the number of elements in the original signal subscript complete set is N , the complement of the support set in the complete set is Λ^c , and the element of the complement set is θ_j . The definition of the local optimal error function is shown in equation (4).

$$r(\Lambda^*) \leq r(\Lambda^o), \forall \Lambda^o \in N(\Lambda^*) \quad (4)$$

In equation (4), the local optimal error function is $r(\Lambda^*)$, the support set corresponding to the error threshold is Λ^o , and the support set estimated for the local search is Λ^* . The global objective function is a function used to balance the estimation error of the support set and the difference of elements. The global objective function is shown in equation (5).

$$g(\Lambda^o, \Lambda^*) = \mu [r(\Lambda^o)]^2 - [|\Lambda^o - \Lambda^*| + |\Lambda^* - \Lambda^o|]^2 \quad (5)$$

In equation (5), the global objective function is $g(\Lambda^o, \Lambda^*)$, and the constant μ is used to measure the difference between Λ^* and Λ^o . After optimizing OSRA, the SSR is shown in equation (6).

$$\hat{x}_T = \Phi_T^+ y \quad (6)$$

In equation (6), the best estimate of the signal support set is $\hat{T}$, and the best-reconstructed signal is $\hat{x}_T$. In summary, the optimization OSRA model achieves the reconstruction of sparse signals by defining support set domains, local optimal error functions, and global objective functions, ultimately obtaining the best estimate of the signal support set and the best reconstructed signal. The OSRA pseudocode is shown in Figure 2.

The convergence of the OSRA is ensured by two key mechanisms: the error-decreasing property inherent in the support set pre-selection backtracking process and the gradient descent mechanism employed during global optimization. Assuming the objective function adheres to Lipschitz continuity and the step size in the global search phase complies with the Armijo condition, the algorithm is guaranteed to converge to a local optimal solution within a finite number of iterations. Furthermore, if the measurement matrix fulfills the RIP condition, OSRA is capable of accurately reconstructing the signal under a sparsity level of K . The time complexity of OSRA is $O(K^{2N} + K^3)$, where N is the signal length and K is the sparsity. It is superior to the exponential complexity of the tree search algorithm, but slightly higher than $O(K^{2N})$ of the traditional OMP.

```

Initialize:
support_set = empty set
residual = y
best_support = empty set
index = argmax(abs(ΦT * residual))
support_set.add(index)
x_support = least_squares(Φ[:, support_set], y)
residual = y - Φ * x_support
if norm(residual) < epsilon:
    break
temp_support = support_set - {j}
temp_x = least_squares(Φ[:, temp_support], y)
temp_residual = y - Φ * temp_x
if norm(temp_residual) < norm(residual):
    support_set = temp_support
    residual = temp_residual
for neighbor in support_set.neighbors():
    candidate_support = support_set.update(neighbor)
    candidate_x = least_squares(Φ[:, candidate_support], y)
    if norm(y - Φ * candidate_x) < norm(residual):
        support_set = candidate_support
        residual = y - Φ * candidate_x
if no improvement:
    support_set = random_perturbation(support_set)
x = least_squares(Φ[:, support_set], y)
return x

```

Figure 2: The OSRA pseudocode.

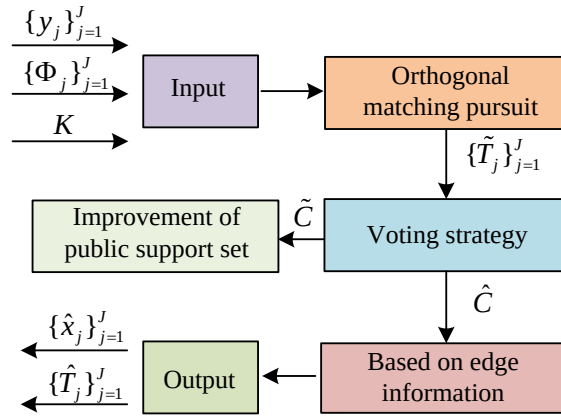


Figure 3: A centralized joint SSR algorithm based on improved public support set estimation.

The error threshold in the OSRA is dynamically adjusted based on the noise level, specifically set at 1.2 times the standard deviation of the noise. This configuration ensures that, during the backtracking process of the support set, only elements that effectively contribute to reducing the signal reconstruction error are retained, thereby preventing the inclusion of noise-induced interference. When the signal residual (the difference between the observed vector and the reconstructed signal) is less than the preset threshold or the number of iterations reaches 100 times, the algorithm is stopped to balance the reconstruction accuracy and computational efficiency and avoid meaningless excessive iterations. In the global optimization stage, by adjusting the difference weights of the elements in the support set (the optimal weight parameters are determined through cross-validation), it is ensured that the algorithm can not only utilize the fast search ability of the greedy algorithm, but also jump out of the local optimal solution through global search, thereby improving the accuracy of reconstruction.

3.2 Design of centralized joint SSR algorithm

In the problem of reconstructing sparse signals from multi-sources, it is necessary to consider the correlation between signals [19]. In the relevant signal source set, the support set between signals is jointly owned [20]. To address this issue, a CJSRA based on improved public support set estimation is proposed. Firstly, obtain a preliminary estimate of the public support set, then identify and remove erroneous elements from the preliminary estimate, retain correct elements, and improve the estimation of the public support set. The definition of a multi-source support set is shown in equation (7).

$$T_j = C \cup I_j, 1 \leq j \leq J \quad (7)$$

In equation (7), the support set for the multi-source signal x_j is T_j , the set of related signal sources is $\{x_j \in R^N\}_{j=1}^J$, the public support set is C , and the innovation support set is I_j . The sparse reconstruction model for multiple signal sources defines the signal support set, which is divided into related signal source set,

public support set, and new information support set to achieve effective reconstruction and analysis of multiple signal sources. The CJSRA based on improved public support set estimation is aimed at the SSR problem of multi-sources. The main content of the study is to improve the accuracy and stability of signal reconstruction by introducing an improved public support set method. The improvement of the public support set is a process of eliminating incorrect elements and retaining correct elements from the initial support set estimation through voting strategies and error function screening. The CJSRA using improved public support set estimation is shown in Figure 3.

The CJSRA using improved public support set estimation consists of three parts: preliminary estimation of public support set, improvement of public support set, and OMP algorithm based on edge information. The preliminary estimation of the public support set requires the use of OMP algorithm, and then a voting strategy is used to obtain the preliminary estimation of the public support set $\tilde{C}$. In the improvement of the public support set, it is necessary to delete the wrong elements, retain the correct elements, and finally obtain the improved public support set estimate $\hat{C}$. The input of an OMP based on edge information is the improved public support set, in order to obtain the final overall support set estimate $\{\hat{T}_j\}_{j=1}^J$ and the reconstructed original signal $\{\hat{x}_j\}_{j=1}^J$. The strategy for enhancing the preliminary estimation of the public support set involves retaining the elements that have been accurately estimated and eliminating those that have been inaccurately estimated. The number of elements with higher credibility in the public support set is shown in equation (8).

$$|\tilde{C}_R| = \left\lceil \frac{\max(s)}{J} \times |\tilde{C}| \right\rceil \quad (8)$$

In equation (8), the number of elements with higher reliability is estimated as $|\tilde{C}_R|$, and the addition function is s . The error function corresponding to the support set $\tilde{C}_R$ is shown in equation (9).

$$r_j(\tilde{C}_R) = \|z_{j, \tilde{C}_R}\|_2 \quad (9)$$

In equation (9), the error function of the support set $\tilde{C}_R$ is $r_j(\tilde{C}_R)$, and the error threshold value is $z_{j,\tilde{C}_R}$. The sum of the error functions corresponding to all signal observation vectors is shown in equation (10).

$$f(\tilde{C}_R) = \sum_{j=1}^J r(\tilde{C}_R) = \sum_{j=1}^J \|z_{j,\tilde{C}_R}\|_2 \quad (10)$$

In equation (10), the total error function is $f(\tilde{C}_R)$. To avoid over-fitting, the total error estimated by the public support set should be less than the total error function. The definition of the support set domain is shown in equation (11).

$$N(C^*) = \{C^o, C^o = C^* \setminus \{c\}, \forall c \in C^*\} \quad (11)$$

In equation (11), the support set domain of the multi-source support set is C^* . When the sum of the minimum error functions exceeds the threshold during iteration, the best estimate $\hat{C}$ of the public support set is set C^* . When initializing the OMP algorithm, the support set is an empty set. Next, the edge information can be used to estimate the subsequent support set to ensure that elements can be correctly identified during the iteration process. At this time, the overall support set estimation is shown in equation (12).

$$\hat{T}_j = mOMP(y_j, \Phi_j, 2K, \hat{C}) \quad (12)$$

In equation (12), the overall support set estimated by the OMP algorithm based on edge information is $\hat{T}_j$, and its size is $2K$. The non-zero term reconstruction result of the signal is shown in equation (13).

$$\hat{x}_{j,\hat{T}_j} = \Phi_{j,\hat{T}_j}^+ y_j \quad (13)$$

In equation (13), the reconstruction result of the non-zero term signal is $\hat{x}_{j,\hat{T}_j}$, which corresponds to the subscripts of the first K maximum amplitude terms to obtain the final estimation. The complexity of CJSRA

using an improved public support set includes preliminary set estimation, set improvement, and OMP algorithm based on edge information. The computational complexity of the improved algorithm of the public support set is shown in equation (14).

$$\begin{cases} O(JKMN + J(K-1)MN + \dots + JK'MN) \\ = O(JSMN) \\ S = K + (K-1) + \dots + K' \end{cases} \quad (14)$$

In equation (14), the computational complexity of the initial estimation of the public support set and the OMP algorithm based on edge information is consistent, which can be represented by $O(JKMN)$. The computational complexity is related to the number of iterations α , and the computational complexity of each iteration is $O(J\alpha MN)$. The evaluation indicators for algorithm reconstruction performance include relative reconstruction error and average error of support set estimation. The relative reconstruction error is shown in equation (15).

$$RER = E \left\{ \frac{\|x - \hat{x}\|_2^2}{\|x\|_2^2} \right\} \quad (15)$$

The relative reconstruction error in equation (15) is RER , and the average of all Monte Carlo experiments and all signals is calculated as $E\{\bullet\}$. The average error of support set estimation is shown in equation (16).

$$ASCE = E\{d(T, \hat{T})\} = 1 - E \left\{ \frac{|T \cap \hat{T}|}{|\hat{T}|} \right\} \quad (16)$$

In equation (16), the average error of support set estimation is $ASCE$, and the measure of support set estimation error is $d(T, \hat{T})$. The CJSRA pseudocode is shown in Figure 4.

```

support_candidates = empty set
for each y_j in Y:
    _, single_support = omp(Phi_j, y_j, K)
    support_candidates.add(single_support)
common_support = vote(support_candidates, threshold=tau)
for c in common_support:
    temp_support = common_support - {c}
    total_error = sum(norm(y_j - Phi_j[:, temp_support] *
least_squares(Phi_j[:, temp_support], y_j)) for y_j in Y)
    if total_error < current_total_error:
        common_support = temp_support
X = empty list
for each y_j in Y:
    Phi_j_support = Phi_j[:, common_support]
    x_j = least_squares(Phi_j_support, y_j)
    X.append(x_j)
return X

```

Figure 4: The CJSRA pseudocode.

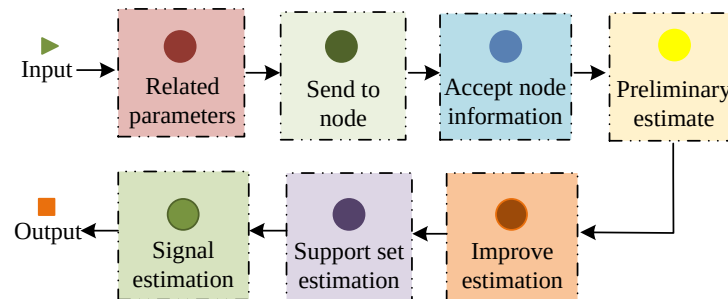


Figure 5: A DJSRA using improved public support set.

CJSRA converges through an iterative process that begins with an initial estimation by the public support set, followed by a refinement phase, and culminates in the OMP reconstruction of edge information. It is preliminarily estimated that the multi-source support set is aggregated by using the voting strategy. In the improvement stage, the support set is optimized by deleting the elements with large error contributions, so that the total error function monotonically decreases. Given that the elements within the support set are finite in number and the error possesses a lower bound, the algorithm converges to a local optimum within a finite sequence of steps. Assuming that the measurement matrices of each source satisfy the RIP conditions, the single-source OMP can accurately estimate the local support set. The voting strategy combined with the stability of RIP enables the estimation of the public support set to resist noise interference. Eventually, the precise reconstruction of multi-source signals is achieved through the edge information OMP. The computational complexity is linearly correlated with the number of sources, sparsity, observation length and signal length. It mainly comes from the preliminary estimation of multi-source OMP, improvement of support sets and joint reconstruction. It is suitable for medium-scale multi-source scenarios, and the computational amount increases linearly with the increase of the number of sources.

When making the preliminary estimation of the multi-source signal support set in the CJSRA algorithm, the "half-vote" rule (that is, an element is retained only when it appears in at least half of the single-source support set) is adopted to filter the random errors in the single-source estimation and improve the reliability of the public support set. By calculating the total change of the multi-source signal reconstruction error after removing a certain element, if the total error increases, the element is retained. Otherwise, it is removed, ensuring that the error of the public support set monotonically decreases after each iteration. During the support set improvement stage, the parameter R controls the proportion of elements deleted in each iteration.

3.3 Design of DJSRA

In the problem of joint SSR with multi-sources, two scenarios are mainly considered. The first scenario is CJSRA, in which each perception node transmits its observed signal values to the central node within the network, and then the central node reconstructs all signals.

In DJSRA scenarios, there is no such central node. Instead, each perception node must collaborate directly with one another to reconstruct the signal [21-22]. Centralized reconstruction scenarios can address issues such as energy consumption, latency, security, interference, and network robustness [23-25]. Nodes send information to the connected nodes based on the network topology, and then each node reconstructs itself, eliminating the problem of a unique central node. The network connection topology can be a fixed structure or a random structure, with the number of input nodes and output nodes corresponding to their in degree and out degree, respectively [26-27]. This can improve energy efficiency, reduce latency, enhance security, reduce interference, and enhance network robustness. The network connection topology consists of fixed and random structures, in which the entry and exit degrees of nodes are equal [28-29]. Random structure refers to the Watts-Strogatz network model, which contains many nodes. In network topologies with degree 1 and degree 2, nodes have one or two input connections and output connections, respectively. With 10 nodes and a degree of 9 for each node, the network topology exhibits a fully connected structure, resembling a centralized reconstruction model. However, in decentralized scenarios, when the degree of network connection topology is low, signals can only receive observation information from some nodes and cannot fully utilize the correlation between signals. In this case, it is necessary to consider the problem of estimating the public support set. To ensure reconstruction accuracy, each node needs to send signals through output connections and receive signals through input connections, receiving and sending information multiple times. But this method will lead to an increase in energy consumption.

To reduce energy consumption and ensure the accuracy of local node reconstruction, the study adopts a public support set improvement method for node signal reconstruction. In this way, each node only needs to receive and send information once to complete signal reconstruction, avoiding the process of receiving and sending information multiple times. The algorithm complexity of this method is similar to that of the CJSRA based on improved public support set estimation. The DJSRA using improved public support set is shown in Figure 5.

The DJSRA is an optimization algorithm used to solve large-scale sparse linear reconstruction problems. This algorithm gradually updates the support set through iteration to achieve efficient signal reconstruction. The

process of the algorithm is as follows: The first step is to transmit an initial signal to each node and then receive feedback from them. Subsequently, preliminary estimates are used to obtain a public support set, and improved estimates are made to obtain more accurate support set estimates. By calculating the estimated values of the support set and the estimated values of the signal, the algorithm can achieve better reconstruction results. During the implementation process, the algorithm provides flexible solvers and regularization options to meet the needs of different situations. In addition, the algorithm can accelerate the solution through communication between nodes and parallelization of computation to achieve efficient reconstruction.

The DJSRA is based on the single information interaction mechanism among distributed nodes. Each node collaboratively optimizes the local estimation by using the neighborhood support set information, and avoids duplicate transmission through the improved public support set method. Network topology ensures information diffusion, supports set estimation to tend to be consistent after limited interactions, and achieves global convergence. When the measurement matrices of each node satisfy RIP, the accuracy of local support set estimation is transferred through network connectivity. Even with low connectivity, the global support set can be inferred from neighborhood information to ensure the robustness of distributed reconstruction. The computational complexity is related to the number of nodes, connectivity and the computational cost of a single node. The distributed structure eliminates the bottleneck of the central node, and the reconfiguration can be completed with a single interaction. The computational cost is comparable to that of CJSRA but the communication cost is lower, making it suitable for large-scale distributed networks. The DJSRA pseudocode is shown in Figure 6.

In the DJSRA, the network connectivity adopts the "small-world" network model (Watt-Strogatz model), and the node connectivity is set to 5 by default (that is, each node is connected to 5 neighboring nodes), balancing the communication cost between nodes and the utilization efficiency of signal correlation. Experiments showed that when the connectivity degree was 9 (fully connected), the reconstruction accuracy was the highest, but the energy consumption increased significantly. Therefore, a medium

connectivity degree is selected as the optimal scheme. After a single information interaction, each node takes a local reconstruction error of less than 0.0001 as the convergence condition to avoid the time-consuming problem of multiple round-trip communications in traditional distributed algorithms and improve real-time performance. By taking advantage of the public support set characteristics of multi-source signals, the sparse structure is estimated through the fusion of neighborhood node information, without the need to preset the global sparsity parameter, which is adapted to the dynamic signal characteristics in distributed scenarios.

4 Performance analysis of SSR algorithm under CS

This chapter analyzes the performance of the proposed SSR algorithm. The first section of this chapter is the performance analysis of the optimization-oriented SSR algorithm. The second section is the performance analysis of the CJSRA. The third section is the performance analysis of the DJSRA.

4.1 Performance Analysis of SSR Algorithm Based on Optimization Orientation

Among the types of data signals used for simulation, the single-source signal was a sparse signal of fixed length, and the non-zero elements followed a Gaussian distribution. The degree of sparsity was controlled by the number of non-zero elements. Multi-source signals were multiple related signals, including shared public support sets and unique innovative support sets for each signal. The signal length was uniformly set to 1024, that is, each signal contained 1024 elements. The measurement matrix was a randomly generated Gaussian matrix, with the matrix size being the product of the number of measurements and the signal length. The column vectors were normalized. Some experiments adjusted the ratio of the number of measurements to the signal length, ranging from 0.10 to 0.20. Additive white Gaussian noise was added to the observation data. The noise level was controlled by the SNR, and the value range of the SNR was from 10 dB to 40 dB, corresponding to the simulation scenarios of different noise intensities.

```

for each node i in G:
    _, local_support = omp( $\Phi_i$ ,  $y_i$ , K)
    send(local_support) to neighbors in G
    received_supports = receive from neighbors
    global_support = union(received_supports + local_support)
    global_support = prune(global_support, method="error-based")
for each node i in G:
     $\Phi_i$ _support =  $\Phi_i[:, \text{global\_support}]$ 
     $x_i$  = least_squares( $\Phi_i$ _support,  $y_i$ )
return [ $x_i$  for all nodes]

```

Figure 6: The DJSRA pseudocode.

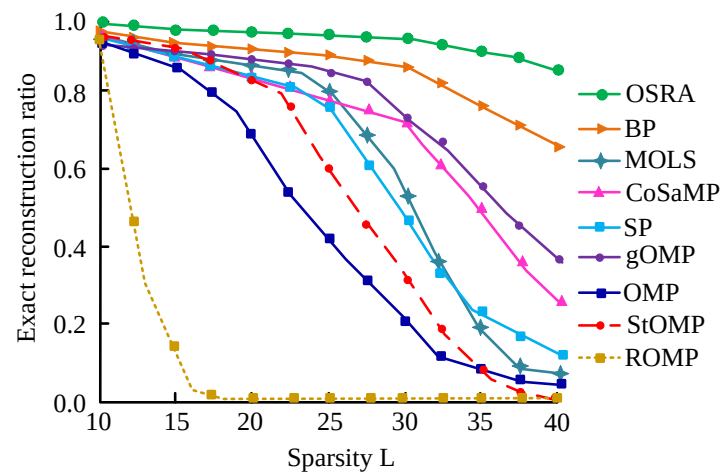


Figure 7: Comparison of successful reconstruction ratios of algorithms under different sparsity.

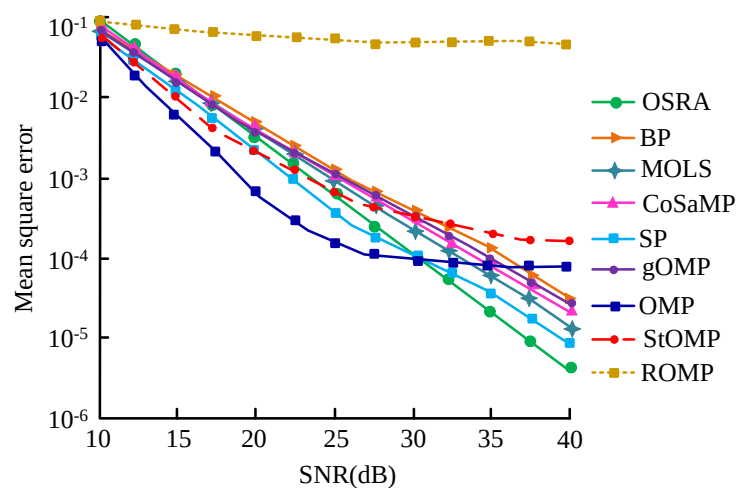


Figure 8: Comparison of mean square errors of various algorithms as signal-to-noise ratio increases.

To verify the performance of the OSRA proposed in the study, the study compared it with other algorithms. Comparison algorithms were the OMP [30], StOMP [31], subspace pursuit (SP) [32], regularized orthogonal matching pursuit (ROMP) [33], compressed sampling matching pursuit (CoSaMP) [34], gOMP, basis pursuit (BP) and orthogonal least square of multi-quadric algorithm (MOLS) [35-36]. The proportion of successful reconstruction under varying levels of sparsity is illustrated in Figure 7.

In Figure 7, the vertical axis represents the numerical range of successful reconstruction ratios, and the horizontal axis represents different signal sparsity values and 9 algorithms. As the sparsity of the signal increased, the proportion of successful reconstructions by each algorithm showed a decreasing trend. When the signal sparsity was 15, the success rate of each algorithm in reconstruction was close to 1.0, indicating that the algorithms performed well when the sparsity was low. However, when the sparsity increased to 40, the success rate of the optimization OSRA proposed in the study was greater than 0.7, while the success rate of other algorithms ranged from 0 to 0.6. By comparison, with the continuous increase of sparsity, the performance advantages of optimization-oriented SSR algorithms became

increasingly apparent. As the SNR increased, the mean square error comparison of each algorithm is shown in Figure 8.

Figure 8 shows the comparison of mean square error between different algorithms without SNR. The smaller the mean square error value, the smaller the difference between the reconstructed signal and the original signal, indicating that the algorithm's reconstruction performance is better. From the graph, as the SNR gradually increased from 10dB to 40dB, the mean square error of the algorithm showed a decreasing trend. Among these algorithms, the OSRA showed significant advantages. At lower SNRs, the mean square error of OSRA was relatively small, and its advantages became more prominent as the SNR increased. When the noise ratio was 40dB, the mean square error of OSRA was less than 10^{-5} , which was 1-4 orders of magnitude smaller than other comparison algorithms. Compared with other algorithms, OSRA could more effectively reduce the mean square error of signals, providing more accurate and reliable results for signal processing. The comparison of running times of different algorithms in noisy scenarios is shown in Figure 9.

In Figure 9, the OSRA proposed in the study had a longer average running time due to its high computational complexity. As the SNR increased, its time consumption

first increased and then decreased. When the SNR was high, the loss of correct elements estimated by the pre-selected support set was reduced, and using global optimization to find missing elements was also faster. To quantify the complexity of the algorithm calculation, the experiment used statistical average iterations $E(P)$ for analysis. To measure the computational complexity and iterative efficiency of the algorithm in a noisy environment more accurately, Table 1 details the total number of iterations of local search and global search under different SNR conditions.

In Table 1, the average number of iterations $E(P)$ was relatively small, indicating that the OSRA had less additional computational complexity. When the signal sparsity was 25 and the SNR was 40, the average number of iterations $E(P)$ was 0.11. At this time, the computational complexity of the OSRA was about 35 times that of the orthogonal matching tracking algorithm. Using a comprehensive analysis of algorithm performance and complexity, the proposed OSRA could adjust the computational complexity to improve the effectiveness of signal reconstruction. However, OSRA still had some drawbacks. Firstly, this algorithm required a longer computation time, especially in Gaussian noise scenarios. Secondly, the algorithm was sensitive to noise, and its

reconstruction performance might be affected when the noise was large. In addition, the computational complexity of this algorithm increased with the increase of sparsity, which might limit its application in sparse signal processing scenarios.

The MSE comparison of OSRA and OMP in 50 Monte Carlo experiments is shown in Table 2. The mean MSE of OSRA was 3 to 4 orders of magnitude lower than that of OMP, and its standard deviation was smaller, indicating that it had higher reconstruction accuracy and stronger stability. All p values were less than 0.001, indicating that the differences between OSRA and OMP were of extremely significant statistical significance (rejecting the null hypothesis, the differences were not random).

4.2 Performance analysis of CJSRA

To verify the effectiveness of the CJSRA using the public support set, the experiments compared the performance of the OMP, SIPP, IOSSP, and the algorithm without adding the public support set improvement. The comparison of the performance of each algorithm in noise-free conditions is shown in Figure 10.

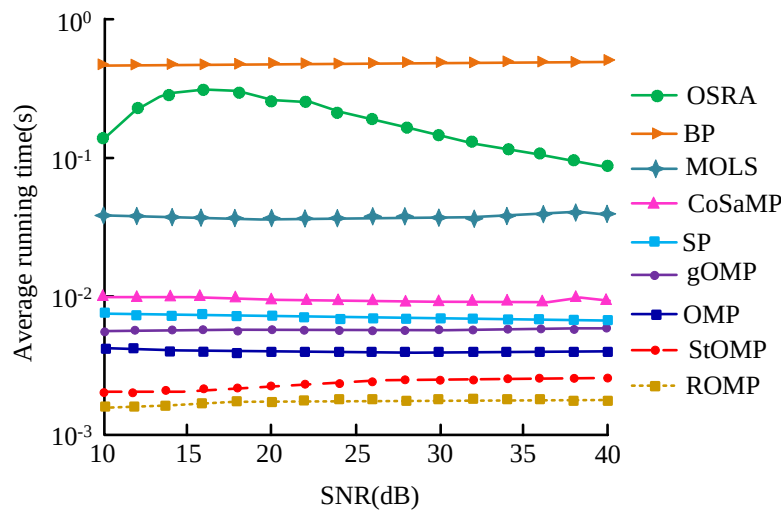


Figure 9: Comparison of runtime of different algorithms in noisy scenarios.

Table 1: The total number of iterations for local and global search under noisy conditions.

SNR	E(P)				
	K=21	K=23	K=25	K=27	K=29
SNR=32	0.17	0.20	0.22	0.25	0.28
SNR=34	0.15	0.17	0.19	0.23	0.27
SNR=36	0.13	0.14	0.15	0.21	0.26
SNR=38	0.11	0.12	0.13	0.18	0.25
SNR=40	0.08	0.10	0.11	0.17	0.24

Table 2: The comparison of MSE between OSRA and OMP.

Algorithm	SNR=20dB	SNR=30dB
OSRA	$2.5 \times 10^{-5} \pm 3.7 \times 10^{-6}$	$9.8 \times 10^{-6} \pm 1.1 \times 10^{-6}$
OMP	$1.2 \times 10^{-2} \pm 2.1 \times 10^{-2}$	$8.3 \times 10^{-3} \pm 1.5 \times 10^{-3}$
p	<0.001	

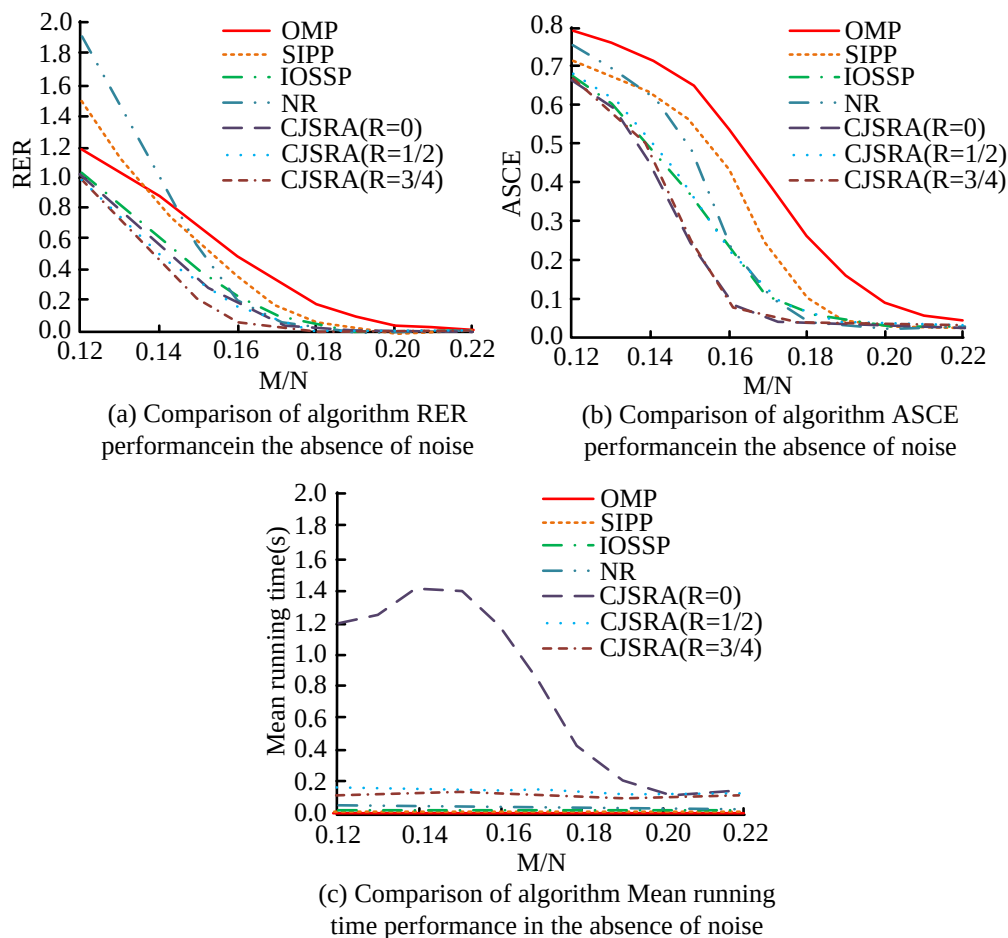


Figure 10: Comparison of performance of various algorithms in noise absence.

Figure 10 (a) shows the comparison of relative reconstruction errors among various algorithms. As the ratio of limited observation data to actual signal length M/N increased, the relative reconstruction error of the algorithm gradually decreased. When the public support set improvement method was not added, the performance of the algorithm was poor. When the M/N was 0.12, the relative reconstruction error was 1.91, and after adding the public support set improvement, the relative reconstruction error was about 1.0. After introducing the parameter R that reduced the number of iterations, the correct elements in the initial estimation were removed in the improvement of the public support set. The reconstruction performance of the algorithm at $R=3/4$ was worse than that of the algorithms at $R=0$ and $R=1/2$. Compared to other algorithms, the proposed CJSRA had the highest support set estimation accuracy. Figure 10 (b) shows the average error of support set estimation, which was similar to the comparison of relative reconstruction errors. Figure 10 (c) shows the calculation of the average running time. As the R -value increased, the algorithm running time gradually decreased, and when $R=0$, the algorithm running time was longer. As the M/N ratio increased, the computational complexity of the algorithm gradually decreased. Comprehensive analysis showed that the CJSRA based on the public support set had obvious advantages, and a value of R of 1/2 could balance the computational complexity and reconstruction

performance. In noisy scenarios, it was necessary to fix the M/N ratio and compare the impact of different SNRs on algorithm reconstruction performance. The M/N ratio was set to 0.18, and the performance comparison of each algorithm in noisy situations is shown in Figure 11.

In Figure 11 (a), after adding the public support set improvement, when the iteration parameters R were 0 and 1/2, and the SNR was 40dB, the relative reconstruction error of CJSRA was the smallest, about 0.08. As shown in Figure 11 (b), as the SNR increased, the average error of the algorithm's support set estimation gradually decreased. Compared to other algorithms, the proposed CJSRA had a minimum support set estimation average error of 0.04. Figure 11 (c) shows the comparison of the average running time of the algorithm. It can be seen that when the iteration parameters R were 1/2 and 3/4, the computational complexity of the algorithm was significantly lower than when R was 0. During the algorithm iteration process, more elements were deleted each time, reducing the number of iterations. Comprehensive analysis showed that improving the public support set could enhance the effectiveness of joint SSR. The introduction of parameter R was to reduce iterations. The larger its value, the more elements were deleted in the iteration, and the fewer the iterations, which could reduce the computational complexity. However, CJSRA performed poorly in noisy scenarios and required a fixed signal-to-noise ratio. In addition, the algorithm had a high computational

complexity and introduced the parameter R to reduce the number of iterations. However, when the value of R was large, the reconstruction performance of the algorithm was actually poor.

The comparison of ASCE between CJSRA and IOSSP in 30 Monte Carlo experiments is shown in Table 3. CJSRA verified the effectiveness of the voting strategy and error screening by improving the estimation of the public support set, with ASCE being more than 50% lower than IOSSP. When $M/N \geq 0.15$, all p was less than 0.01, indicating that the support set estimation accuracy of

CJSRA was significantly better than that of the comparison algorithm.

4.3 Performance analysis of DJSRA

To verify the performance of the DJSRA proposed in the study, the experiment compared OMP and DIPP, with different values set for the degree of network connectivity. Table 4 displays the average frequency at which DIPP network nodes receive and transmit information within a noise-free environment.

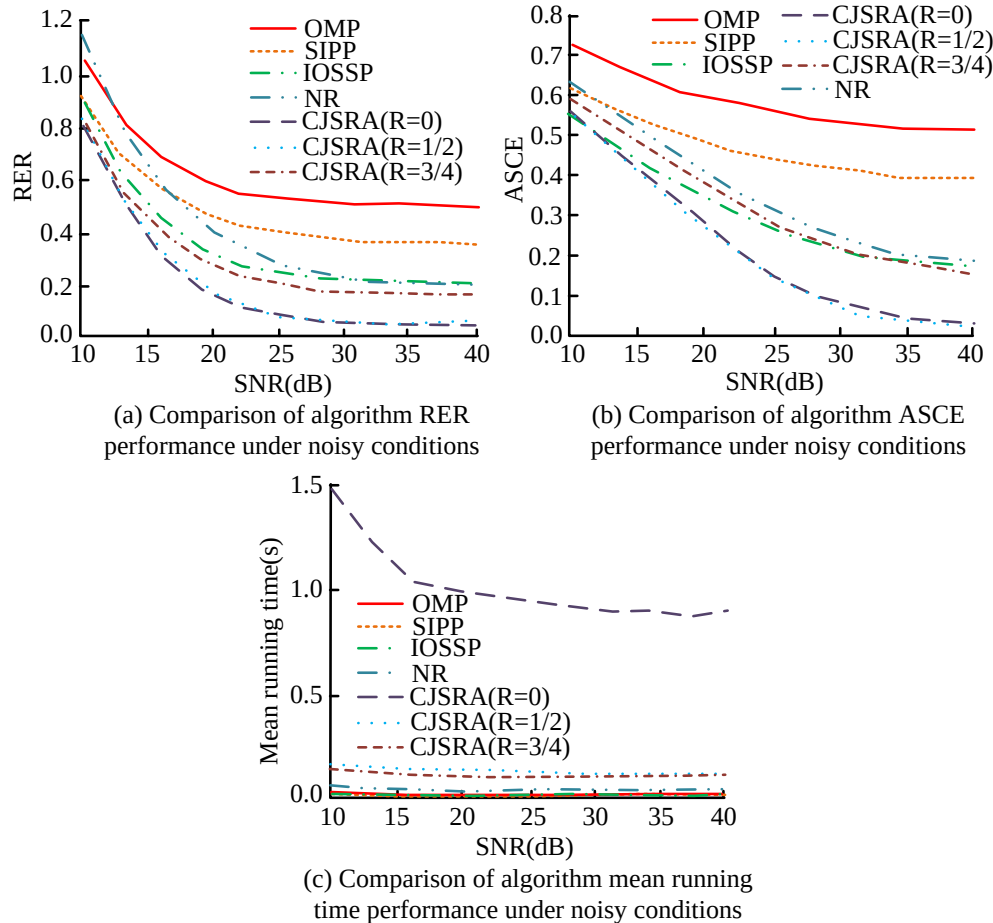


Figure 11: Performance comparison of various algorithms under noisy conditions.

Table 3: The comparison of ASCE between CJSRA and IOSSP.

Algorithm	M/N=0.15	M/N=0.20
CJSRA	0.13±0.02	0.08±0.01
IOSSP	0.27±0.05	0.19±0.03
p	0.003	0.001

Table 4: The average number of times DIPP network nodes receive and transmit information in a noise free environment.

M/N	Average number of messages received and transmitted						
	C3	C4	C5	C6	C7	C8	C9
M/N=0.10	7.41	36.25	120.21	198.19	136.23	100.06	62.46
M/N=0.12	7.55	20.36	65.12	13.53	80.15	80.25	141.45
M/N=0.14	3.67	15.45	15.12	7.12	56.12	32.54	5.63
M/N=0.16	2.57	8.36	8.36	5.86	15.36	7.68	4.68
M/N=0.18	2.42	3.72	5.21	4.62	8.45	6.54	4.88
M/N=0.20	2.43	3.69	4.63	4.95	8.23	6.32	4.53

As shown in Table 4, under different parameters, there was a large difference in the average number of times each node in the distributed parallel tracking algorithm receives and transmits information. Nodes may experience multiple repetitions of receiving and transmitting information, especially when the M/N ratio was less than 0.12. However, the DJSRA proposed in this study could reconstruct information after a single transmission at each node, saving time wasted on repeated transmissions. The performance comparison of each algorithm in a noiseless environment is shown in Figure 12.

In Figures 12 (a) and (b), as the M/N ratio increased, the error of the algorithm gradually decreased. As the network connectivity increased, compared to the other two algorithms, the DJSRA proposed in the study had a smaller initial error value and faster convergence speed. When the network connectivity was C9 and the M/N ratio

was 0.10, the relative reconstruction error and average support set estimation error of the DJSRA were 0.49 and 0.46, respectively. Figure 12 (c) shows the average running time of different algorithms reconstructing a single signal. The DIPP algorithm was greatly affected by the M/N ratio. When the M/N ratio was small, its running time could reach up to 1.1 seconds, which was proportional to the number of times the node received and sent information. In contrast, the average running time of the DJSRA proposed in the study did not change significantly. When the network connectivity was C9, the average running time remained within the range of less than 0.5 seconds. To further compare the communication efficiency of each algorithm in a noisy environment, Table 5 presents in detail the average number of times each node received and sent information in the DIPP algorithm and the DJSRA.

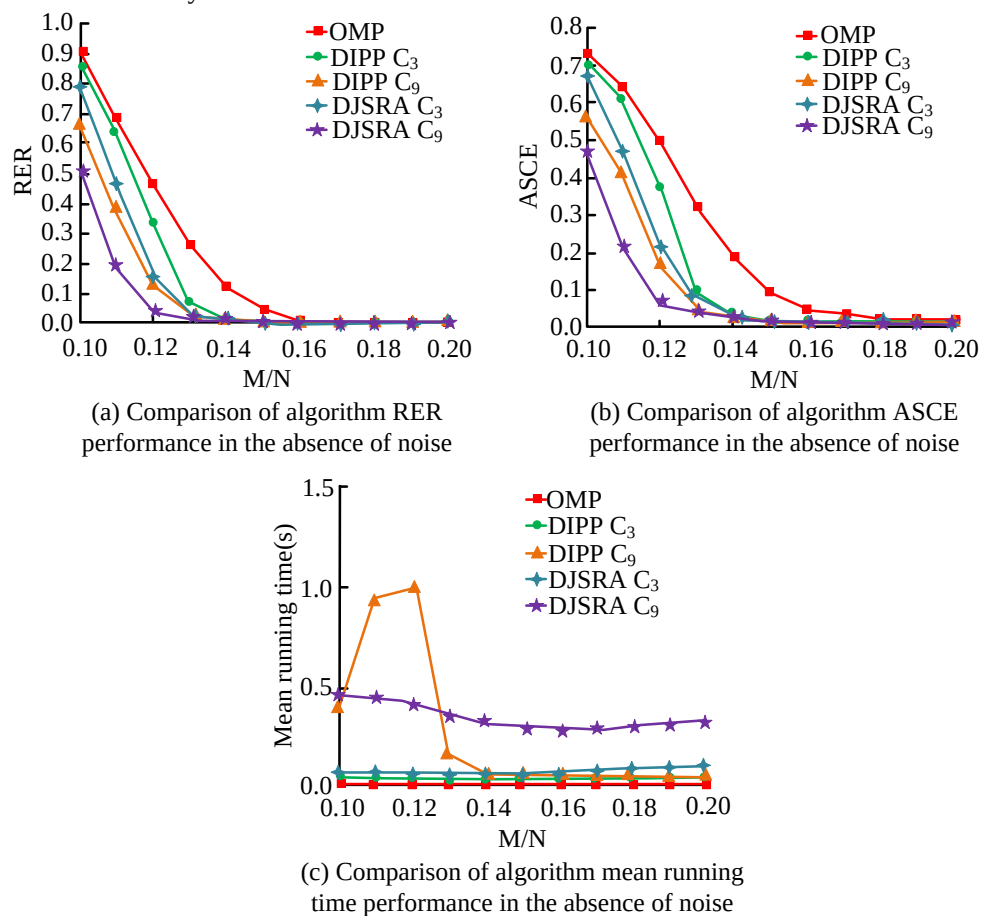


Figure 12: Comparison of performance of various algorithms in noiseless environments.

Table 5. The average number of times DIPP received and sent messages per node in noisy environments.

SNR(dB)	Average number of messages sent and received						
	C3	C4	C5	C6	C7	C8	C9
SNR=10	6.86	52.35	102.35	196.62	183.25	173.24	169.00
SNR=15	7.69	60.12	85.14	101.38	123.54	162.45	199.56
SNR=20	5.95	9.65	12.36	28.66	85.34	125.46	173.36
SNR=25	4.16	5.63	10.46	14.27	22.36	45.37	56.34
SNR=30	3.52	5.24	6.25	8.26	18.36	25.36	28.54
SNR=35	3.16	5.21	6.03	9.48	10.23	12.35	15.36
SNR=40	2.85	5.01	6.01	4.94	8.36	10.14	13.16

Table 5 shows that there was a significant difference in the average number of times DIPP nodes received and sent messages in noisy environments, especially when the SNR was less than 20, the number of times DIPP nodes received and sent messages in the network varied significantly. The DJSRA proposed in the study could complete signal reconstruction by receiving and sending information once per section, avoiding the time consumption of nodes repeatedly sending information multiple times. The performance comparison of each algorithm in noisy environments is shown in Figure 13.

As shown in Figure 13 (a), as the network connectivity increased, the relative reconstruction error of the DJSRA proposed in the study gradually decreased, and the error value was smaller than that of other algorithms. When the SNR was 10, the relative reconstruction error using the orthogonal matching tracking algorithm was 0.57. The minimum relative reconstruction error of the DJSRA proposed in the study was 0.34. Figure 13 (b) shows the average error of the algorithm's support set estimation, which was similar to the result of relative reconstruction error. As the SNR increased, the average

error of support set estimation for DIPP and the proposed algorithm gradually approached, with an error value of less than 0.1 when the SNR was 40. Figure 13 (c) shows the comparison of the average running time of the algorithm. When the network connectivity was low, the average running time of DIPP changed significantly, which was positively correlated with the number of times nodes received and sent information. The average running time of the DJSRA proposed in the study was relatively stable. Comprehensive analysis showed that the DJSRA proposed in the study could reduce computational complexity while ensuring the effectiveness of signal reconstruction.

The RER comparison between DJSRA and DIPP in 40 Monte Carlo experiments with network connectivity C5 is shown in Table 6. DJSRA achieved reconfiguration through a single interaction, with an RER that was over 40% lower than DIPP, and a smaller standard deviation (higher stability). In the low SNR scenario (SNR=10dB), $p < 0.05$, indicating that the distributed collaborative mechanism significantly improved the anti-noise performance.

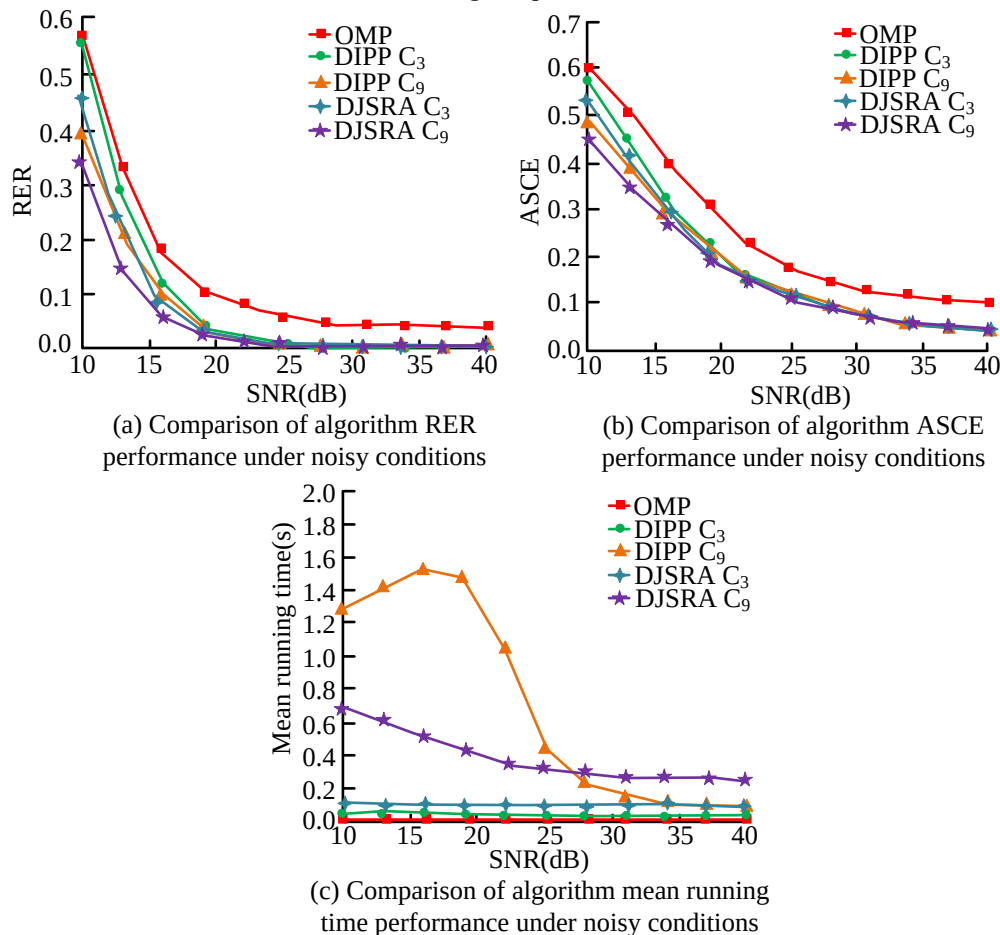


Figure 13: Comparison of performance of various algorithms in noiseless environments.

Table 6: The comparison of RER between DJSRA and DIPP.

Algorithm	SNR=20dB	SNR=30dB
DJSRA	0.34±0.06	0.18±0.03
DIPP	0.57±0.12	0.31±0.08
p	0.012	0.008

Table 7: Comparison experimental results between sparse signal recovery algorithm and other methods.

Algorithm	Relative reconstruction error	Successful refactoring rate	Reference
OSRA	0.763	0.793	This study
CJSRA	1.026	0.818	This study
DJSRA	0.332	0.856	This study
ISTA	1.687	0.787	Lv et al. [37]
FISTA	1.986	0.716	Guo et al. [38]
AMP	1.674	0.806	Gerbelot et al. [39]

The experimental results comparing the SSR algorithm with other methods are shown in Table 7. Iterative Shrinkage-Thresholding Algorithm (ISTA) adopted the default configuration in the literature. The step size was set to 0.9 times the inverse of the Lipschitz constant, the maximum number of iterations was 200 times, and the termination condition was that the change in reconstruction error in five consecutive iterations was less than 10^{-5} . Fast Iterative Shrinkage-Thresholding Algorithm (FISTA) was an accelerated version based on literature optimization, introducing a momentum term to enhance the convergence speed. The step size was dynamically adjusted, and the maximum number of iterations was set at 150 times. The termination condition was consistent with that of ISTA. The Approximate Message Passing (AMP) was implemented with reference to the original literature, including an adaptive denoising module. The number of iterations was set at 50 times, and the measurement matrix adopted the same column-normalized random Gaussian matrix preprocessing method as the algorithm in this paper. All baseline methods used the same signal model (Gaussian sparse signal), measurement matrix (column-normalized random Gaussian matrix), and noise model (additive Gaussian white noise) as the algorithm in this paper, and the parameter adjustments strictly referred to the optimal practices in the original literature to ensure consistent comparison conditions. As shown in Table 7 the SSR algorithm proposed in the study had a relatively low reconstruction error and a high reconstruction success rate. Among them, DJSRA performed the best and FISTA performed the worst. This was mainly due to the introduction of an iterative threshold selection strategy in the DJSRA, which enabled it to achieve high reconstruction accuracy with fewer iterations. FISTA used an acceleration strategy, which could accelerate convergence but sacrificed reconstruction accuracy to some extent.

Performance data revealed the inherent trade-off among accuracy, efficiency and robustness in algorithm design. By introducing the backtracking verification of support sets and the global optimization mechanism, OSRA significantly outperformed the traditional greedy algorithm in terms of reconstruction accuracy. However, the computational complexity increased due to the increase in the number of iterations and the number of least squares solution times, reflecting the design logic of "accuracy for efficiency". As a centralized algorithm, CJSRA improved the estimation accuracy of the support set through the multi-source voting strategy and edge information OMP. However, all source data needed to be transmitted to the central node, and the communication

cost increased linearly with the number of sources. While DJSRA reduced the communication overhead through distributed single information interaction (only the summary of the support set was transmitted between adjacent nodes). However, due to the reliance on local information, the reconstruction error was slightly higher (RER was about 10%-15% higher), demonstrating the "trade-off between the accuracy of multi-source collaboration and communication cost". Furthermore, in scenarios with low SNR or low observation rate, complex algorithms (such as OSRA, CJSRA) were more sensitive to noise, while traditional algorithms (such as OMP) exhibited stronger robustness due to their simple logic, reflecting the "trade-off between noise robustness and observation rate". Distributed topology analysis showed that DJSRA had the best accuracy in fully connected networks, but the error increased significantly in low-connectivity topologies, reflecting the "dependency relationship between network structure and reconstruction performance". These trade-offs provided clear guidance for practical applications: OSRA was preferred in high-precision scenarios, DJSRA was applicable in distributed scenarios with limited communication, and lightweight algorithms such as OMP could be adopted in scenarios with high real-time requirements.

5 Conclusion

CS is widely used in applied mathematics, remote sensing imaging, and computer science, with its core problem being SSR. The research proposed SSR algorithms suitable for different scenarios for single-source and multi-source joint SSR. For SSR from a single-source, research was conducted to improve the greedy algorithm support set estimation and propose an OSRA. For the reconstruction of sparse signals from multi-sources, a CJSRA based on improved public support set estimation was proposed by studying and utilizing the improved public support set. Through node signal reconstruction, a DJSRA using an improved public support set was proposed. Experimental data showed that the mean square error of the OSRA was less than 10^{-5} , which was 1-4 orders of magnitude smaller than OMP and other comparative algorithms. When the signal sparsity was 25 and the SNR was 40, the average number of iterations was 0.11. At this time, the computational complexity of the OSRA was about 35 times that of the OMP algorithm. Based on comprehensive analysis, it can be concluded that the OSRA proposed in the study could adjust the computational complexity to improve the effectiveness of signal reconstruction. For the joint SSR algorithm with multi-sources, the performance of the algorithm was poor

without the addition of the public support set improvement method. When the M/N was 0.12, the relative reconstruction error was 1.91, and the relative reconstruction error after the addition of the public support set improvement was about 1.0. After introducing the parameter R that affected the number of iterations, the correct elements in the initial estimation were removed in the improvement of the public support set. The reconstruction performance of the algorithm was worse when $R=3/4$ than when $R=0$ and $R=1/2$. Compared to other algorithms such as OMP, the proposed CJSRA had the highest support set estimation accuracy. When the SNR was 10, the relative reconstruction error using OMP was 0.57. The minimum relative reconstruction error of the DJSRA proposed in the study was 0.34. Based on the above content, the superiority and contribution value of this study can be concluded. Firstly, a greedy algorithm-based SSR algorithm was proposed for the problem of single-source SSR. This algorithm could achieve high signal reconstruction performance gains at low complexity. Secondly, for the problem of joint SSR from multi-sources, a joint SSR algorithm based on an improved public support set was proposed, which could achieve good signal reconstruction performance at low complexity. Finally, the study focused on decentralized reconstruction scenarios and hybrid support set models, analyzed the impact of network connection topology on the estimation of public support sets and the energy consumption of nodes receiving and transmitting information, and proposed corresponding improvement strategies. Through the above research, better reconstruction results can be achieved in the field of SSR, and better performance can be achieved in different scenarios. However, OSRA and CJSRA algorithms performed poorly in noisy scenarios and require a fixed SNR. In addition, when introducing the parameter R to reduce the number of iterations, if the value of R was large, the reconstruction performance of the algorithm was actually poor. Therefore, these algorithms were sensitive to noise, and their computational complexity increased with increasing sparsity, limiting their application in sparse signal processing scenarios. The limitation of this study was that the computational complexity of the single-source SSR algorithm was about 35 times that of the orthogonal matching tracking algorithm, and the algorithm ran relatively long. In response to the above issues, future research directions can focus on improving the performance of algorithms in noisy scenarios and reducing computational complexity to adapt to a wider range of sparse signal processing applications, and start with the finite equidistant property to further improve the performance of SSR.

References

- [1] Shenghang Zhou, Xiubao Sui, Wenhui Zhang, Yingzi Hua, Qian Chen, and Guohua Gu. High-resolution image transmission through scattering media based on compressive encoding. *Optical Engineering*, 60(3):35108-35123, 2021. <https://doi.org/10.1117/1.OE.60.3.035108>
- [2] Jianfeng Wang, Zhiyong Zhou, and Jun Yu. Statistical inference for block sparsity of complex-valued signals. *IET Signal Processing*, 14(4):154-161, 2020. <https://doi.org/10.1049/iet-spr.2019.0200>
- [3] Sheikh Rafiul Islam, Santi P. Maity, and Ajoy Kumar. On learning based compressed sensing for high resolution image reconstruction. *IET Image Processing*, 15(2):393-404, 2020. <https://doi.org/10.1049/ipr2.12029>
- [4] Chaoang Xiao, Hesheng Tang, and Yan Ren. Compressed sensing reconstruction for axial piston pump bearing vibration signals based on adaptive sparse dictionary model. *Measurement & Control*, 53(3-4):649-661, 2020. <https://doi.org/10.1177/0020294019898725>
- [5] Jia Wu. A fast-iterative reconstruction algorithm for sparse angle CT based on compressed sensing. *Future Generation Computer Systems*, 126(1):289-294, 2022. <https://doi.org/10.1016/j.future.2021.08.013>
- [6] Chao Wang, Xiaohong Shen, Haiyan Wang, and Haodi Mei. Energy-efficient collection scheme based on compressive sensing in underwater wireless sensor networks for environment monitoring over fading channels. *Digital Signal Processing*, 127(1):103530-103537, 2022. <https://doi.org/10.1016/j.dsp.2022.103530>
- [7] Xingyuan Wang, Cheng Liu, and Donghua Jiang. A novel triple-image encryption and hiding algorithm based on chaos, compressive sensing and 3D DCT. *Information Sciences*, 574(1):505-527, 2021. <https://doi.org/10.1016/j.ins.2021.06.032>
- [8] Baohong Bai, Xiaodong Li, Tao Zhang, and Dakai Lin. Nonconvex $L_{1/2}$ minimization based compressive sensing approach for duct azimuthal mode detection. *AIAA Journal*, 58(4):3932-3946, 2020. <https://doi.org/10.2514/1.J059341>
- [9] Yuki Kato, Soma Watahiki, and Masayoshi Otaka. Full-field measurements of high-frequency micro-vibration under operational conditions using sub-Nyquist-rate 3D-DIC and compressed sensing with order analysis. *Mechanical Systems and Signal Processing*, 224(1):112179-112208, 2025. <https://doi.org/10.1016/j.ymssp.2024.112179>
- [10] Qutaiba Kadhim, and Waleed Ameen Mahmoud Al-Jawher. A new multiple-chaos image encryption algorithm based on block compressive sensing, swin transformer, and wild horse optimization. *Multidisciplinary Science Journal*, 7(1):2025012-2025025, 2025. <https://doi.org/10.31893/multiscience.2025012>
- [11] Alireza Zeynali, and Mohammad Ali Tinati. Signal reconstruction based on time-varying sliding window in WSNs using compressed sensing. *International Journal of Communication Systems*, 38(1):6080-6099, 2025. <https://doi.org/10.1002/dac.6080>
- [12] You Zhao, Xiaofeng Liao, Xing He, Rongqiang Tang, and Weiwei Deng. Smoothing inertial neurodynamic approach for sparse signal reconstruction via L_p -norm minimization. *Neural Networks*, 140(17):100-

- 112, 2021. <https://doi.org/10.1016/j.neunet.2021.02.006>
- [13] Huiizu Lin, Shuai Sun, Liang Jiang, Longkun Du, Hongkang Hu, and Weitao Liu. Optimal parameters for image reconstruction in ghost imaging via sparsity constraints. *Optical Engineering*, 59(12):123101-123109, 2020. <https://doi.org/10.1117/1.OE.59.12.123101>
- [14] Yixuan Li, Xiaopeng Yang, Tian Lan, Renjie Liu, and Xiaodong Qu. Parameter inversion by a modified reflected signal reconstruction method for thin-layered media. *IEEE Antennas and Wireless Propagation Letters*, 21(5):958-962, 2022. <https://doi.org/10.1109/LAWP.2022.3152843>
- [15] Na Li, Xinghui Yin, Huanyin Guo, Sulan Zong, and Wei Fu. Variable step-size matching pursuit based on oblique projection for compressed sensing. *IET Image Processing*, 14(4):766-773, 2020. <https://doi.org/10.1049/iet-ipr.2019.0916>
- [16] Shengheng Liu, Yimin D. Zhang, Tao Shan, and Ran Tao. Structure-aware Bayesian compressive sensing for frequency-hopping spectrum estimation with missing observations. *IEEE Transactions on Signal Processing*, 66(8):2153-2166, 2018. <https://doi.org/10.1109/TSP.2018.2806351>
- [17] Shirshendu Roy, Debiprasad P. Acharya, and Ajit K. Sahoo. Fast OMP algorithm and its FPGA implementation for compressed sensing-based sparse signal acquisition systems. *IET Circuits, Devices & Systems*, 15(6):511-521, 2021. <https://doi.org/10.1049/cds2.12047>
- [18] P. Sudhakar Reddy, B. S. Raghavendra, and A. V. Narasimhadhan. Root-free annihilating filter method for sparse signal reconstruction. *Circuits, Systems, and Signal Processing*, 44(1):670-683, 2025. <https://doi.org/10.1007/s00034-024-02871-3>
- [19] Cheng Zhang, Qianwen Chen, Meiqin Wang, and Sui Wei. Optimised two-dimensional orthogonal matching pursuit algorithm via singular value decomposition. *IET Signal Processing*, 14(4):717-724, 2021. <https://doi.org/10.1049/iet-spr.2019.0090>
- [20] Abbas Ali Sharifi, and Ghanbar Azarnia. Compressive sensing for PAPR reduction of DC-biased optical OFDM signals with exploiting joint sparsity for signal reconstruction. *Optical Engineering*, 59(9):96105-96116, 2020. <https://doi.org/10.1117/1.OE.59.9.096105>
- [21] You You, and Li Zhang. Exploiting angular spread in channel estimation of millimeter wave MIMO system. *IET Communications*, 16(3):195-205, 2022. <https://doi.org/10.1049/cmu2.12329>
- [22] Maysara Ghaith, Ahmad Siam, Zhong Li, and Wael El-Dakhkhni. Hybrid hydrological data-driven approach for daily streamflow forecasting. *Journal of Hydrologic Engineering*, 25(2):4019063-4019071, 2020. [https://doi.org/10.1061/\(ASCE\)HE.1943-5584.0001866](https://doi.org/10.1061/(ASCE)HE.1943-5584.0001866)
- [23] Shruthi Khadri, Naveen K Bhoganna, and Madam Aravind Kuma. IoT driven joint compressed sensing and shallow learning approach for ECG signal-reconstruction. *Indonesian Journal of Electrical Engineering and Computer Science*, 34(1):666-676, 2024. <https://doi.org/10.11591/ijeecs.v34.i1.pp666-676>
- [24] Jie Zan. Research on robot path perception and optimization technology based on whale optimization algorithm. *Journal of Computational and Cognitive Engineering*, 1(4):201-208, 2023. <https://doi.org/10.47852/bonviewJCCE597820205514>
- [25] Ghous Ali, and Masfa Nasrullah Ansari. Multiattribute decision-making under Fermatean fuzzy bipolar soft framework. *Granular Computing*, 7(2):337-352, 2022. <https://doi.org/10.1007/s41066-021-00270-6>
- [26] A. Guleria, and R. K. Bajaj. T-spherical fuzzy soft sets and its aggregation operators with application in decision-making. *Scientia Iranica*, 28(2):1014-1029, 2021. <https://doi.org/10.24200/SCI.2019.53027.3018>
- [27] Qilei Zhang, Lei Yu, Feng He, and Yifei Ji. Modified complex multitasks Bayesian compressive sensing using Laplacian scale mixture prior. *IET signal processing*, 16(5):601-614, 2022. <https://doi.org/10.1049/sil2.12134>
- [28] Omar Abu Arqub, Jagdev Singh, and Mohammed Alhodaly. Adaptation of kernel functions-based approach with Atangana-Baleanu-Caputo distributed order derivative for solutions of fuzzy fractional Volterra and Fredholm integrodifferential equations. *Mathematical Methods in the Applied Sciences*, 46(7):7807-7834, 2023. <https://doi.org/10.1002/mma.7228>
- [29] Zihan Luo, Jing Liang, and Jie Ren. Deep learning based compressive sensing for UWB signal reconstruction. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1-10, 2022. <https://doi.org/10.1109/TGRS.2022.3181891>
- [30] Quan Sun, Fei-Yun Wu, Kunde Yang, and Chunlong Huang. Sparse signal recovery from noisy measurements via searching forward OMP. *Electronics Letters*, 58(3):124-126, 2022. <https://doi.org/10.1049/ell2.12365>
- [31] Yichen Chen, Bingru Chen, Wu Guo, Jia Jia, and Yijun Wang. Data evaluation algorithm for compensation litigation for ecotope pollution damage in the Yellow River valley caused by industrial wastewater from the perspective of green development. *Water Science & Technology*, 88(3):631-644, 2023. <https://doi.org/10.2166/wst.2023.215>
- [32] Jian Zhang, Bin Chen, Ruiqin Xiong, and Yongbing Zhang. Physics-inspired compressive sensing: Beyond deep unrolling. *IEEE Signal Processing Magazine*, 40(1):58-72, 2023. <https://doi.org/10.1109/MSP.2022.3208394>
- [33] Yue Xiao, Lei Yuan, Junyu Wang, Wenxin Hu, and Ruimin Sun. Sparse reconstruction of sound field using bayesian compressive sensing and equivalent source method. *Sensors*, 23(12):5666-5685, 2023. <https://doi.org/10.3390/s23125666>

- [34] Zezheng Qin, Yiming Ma, Lingyu Ma, Guangxing Liu, and Mingjian Sun. Convolutional sparse coding for compressed sensing photoacoustic CT reconstruction with partially known support. *Biomedical Optics Express*, 15(2):524-539, 2024. <https://doi.org/10.1364/BOE.507831>
- [35] David L. Donoho, Arian Maleki, and Andrea Montanari. Message-passing algorithms for compressed sensing. *Proceedings of the National Academy of Sciences*, 106(45):18914-18919, 2009. <https://doi.org/10.1073/pnas.0909892106>
- [36] Amir Beck, and Marc Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183-202, 2009. <https://doi.org/10.1137/080716542>
- [37] Dan Lv, Yan Zhang, Danjv Lv, Jing Lu, Yixing Fu, and Zhun Li. Combining CBAM and iterative shrinkage-thresholding algorithm for compressive sensing of bird images. *Applied Sciences*, 14(19):8680-8694, 2024. <https://doi.org/10.3390/app14198680>
- [38] Xinrong Guo, Fengkai Liu, and Darong Huang. Migration through resolution cell correction and sparse aperture ISAR imaging for maneuvering target based on whale optimization algorithm-Fast iterative shrinkage thresholding algorithm. *Sensors*, 24(7):2148-2163, 2024. <https://doi.org/10.3390/s24072148>
- [39] Cédric Gerbelot, and Raphaël Berthier. Graph-based approximate message passing iterations. *Information and Inference*, 12(4):2562-2628, 2024. <https://doi.org/10.48550/arXiv.2109.11905>

