

Article

A Faster and More Accurate Iterative Threshold Algorithm for Signal Reconstruction in Compressed Sensing

Jianxiang Wei ^{1,*} , Shumin Mao ², Jiming Dai ², Ziren Wang ³, Weidong Huang ⁴ and Yonghong Yu ⁵¹ School of Management, Nanjing University of Posts and Telecommunications, Nanjing 210003, China² School of Internet of Things, Nanjing University of Posts and Telecommunications, Nanjing 210003, China; 1020072115@njupt.edu.cn (S.M.); b19060238@njupt.edu.cn (J.D.)³ School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing 210023, China; b20032110@njupt.edu.cn⁴ Key Research Base of Philosophy and Social Sciences in Jiangsu-Information Industry Integration Innovation and Emergency Management Research Center, Nanjing 210003, China; huangwd@njupt.edu.cn⁵ College of Tongda, Nanjing University of Posts and Telecommunications, Nanjing 210003, China; yuyh@njupt.edu.cn

* Correspondence: jxwei@njupt.edu.cn

Abstract: Fast iterative soft threshold algorithm (FISTA) is one of the algorithms for the reconstruction part of compressed sensing (CS). However, FISTA cannot meet the increasing demands for accuracy and efficiency in the signal reconstruction. Thus, an improved algorithm (FIPITA, fast iterative parametric improved threshold algorithm) based on mended threshold function, restart adjustment mechanism and parameter adjustment is proposed. The three parameters used to generate the gradient in the FISTA are carefully selected by assessing the impact of them on the performance of the algorithm. The developed threshold function is used to replace the soft threshold function to reduce the reconstruction error and a restart mechanism is added at the end of each iteration to speed up the algorithm. The simulation experiment is carried out on one-dimensional signal and the FISTA, RadaFISTA and RestartFISTA are used as the comparison objects, with the result that in one case, for example, the residual rate of FIPITA is about 6.35% lower than those three and the number of iterations required to achieve the minimum error is also about 102 less than that of FISTA.

Keywords: FISTA; compressed sensing; threshold function; restart adjustment mechanism; parameter adjustment; signal reconstruction



Citation: Wei, J.; Mao, S.; Dai, J.; Wang, Z.; Huang, W.; Yu, Y. A Faster and More Accurate Iterative Threshold Algorithm for Signal Reconstruction in Compressed Sensing. *Sensors* **2022**, *22*, 4218. <https://doi.org/10.3390/s22114218>

Received: 7 May 2022

Accepted: 27 May 2022

Published: 1 June 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

At the turn of the twentieth century, the compressed sensing (CS) theory emerged in the field of signal processing [1,2]. According to studies, the CS is particularly well suited to wireless sensor networks (WSN) and has a promising future in wireless data communication [3–7].

To reconstruct the original signal with high precision, we require a faster convergence speed with less sampling information in the reconstruction algorithm [8] of CS. Iterative threshold algorithm (ITA), a convex optimization algorithm, converts the reconstruction problem into a convex optimization problem that can be solved by linear programming. To begin, the “threshold” in the algorithm refers to the hard [9] and soft threshold functions [10]. Second, the lasso optimization problem was studied with a gradient descent algorithm [11] and an iterative threshold function. The proximal gradient algorithm (PGA) [12] was also created by the researchers to solve the lasso problem [13]. Overall, the iterative soft threshold algorithm can be regarded as a combination of the PGA and the soft threshold function. The “soft threshold” is utilized as the gradient of the objective function and the “gradient descent” is employed to obtain the best value. Researchers have raised numerous improved FISTAs, such as AFISTA, [14] which fastens the FISTA by a continuation strategy,

and S-FISTA [15], which uses a scaling technique for gradient proximal step. EFISTA [16], monotonic FISTA [17], restart FISTA [18,19] and backtracking strategy [20] are also available. There is no doubt that those algorithms can play a larger role in a variety of fields [21–23].

This study will put forward a new improved FISTA. We will firstly suggest a better threshold function and demonstrate theoretically that it can overcome the discontinuity and constant deviation of the classical threshold function. Simultaneously, the study will integrate three parameters and the restart judgment mechanism. In this way, an improved iterative threshold algorithm with a higher convergence rate and better reconstruction performance would be formed.

2. Related Work

2.1. Basic Theory

In the sensor network structure module, compressed sensing is essential. Figure 1 depicts the basic steps of compressed sensing.

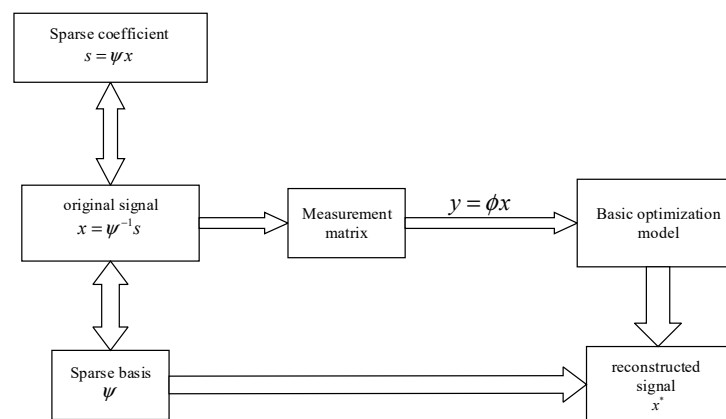


Figure 1. Basic framework of compressed sensing. (x denotes the original signal, and the x^* denotes the reconstructed signal).

In the reconstruction model, as shown in the above graph, the observation vector $y \in R^M$ can be expressed as:

$$y = \phi x = \phi \psi^{-1} S = \Theta S, \quad (1)$$

To solve such a linear inverse problem, the least square method is usually used and the form is as follows:

$$\hat{x}_{LS} = \underset{x}{\operatorname{argmin}} \|y - \phi x\|_2^2, \quad (2)$$

To attain decent results, such a morbid linear formula must avoid significant estimation variance caused by unbiased estimation and the Tikhonov regularization approach is employed. Simultaneously, Equation (3) is formed, which is the classical lasso problem [13] due to the benefit that the regular term of the L1 norm can produce sparse solutions,

$$\min_x \frac{1}{2} \|y - \phi x\|_2^2 + \lambda \|x\|_1, \quad (3)$$

Among this expression, $\lambda > 0$ is the regularization parameter.

2.2. Fast Iterative Soft Threshold Function Reconstruction Algorithm Based on Proximal Gradient Descent

It is clear that the basic optimization problem described in Equation (3) can be solved by the proximal gradient descent method.

After simplification and optimization, it can be calculated like this:

$$\begin{aligned} x^k &= \text{prox}_{t,\lambda|\cdot|_1}(x^{k-1} - t\nabla g(x^{k-1})) \\ &= S_{\lambda t}(x^{k-1} - t\phi^T(\phi x^{k-1} - y)) \end{aligned} \tag{4}$$

where $S_{\lambda t}(\cdot)$ is the soft threshold operator, $g(x)$ is the first half of Equation (3) and is described as $g(x) = \frac{1}{2}\|y - \phi x\|_2^2$. In addition, it has $\nabla g(x) = \nabla \frac{1}{2}\|y - \phi x\|_2^2 = \phi^T(y - \phi x)$ and t is achieved with the Lipschitz continuity in R^n of $\nabla g(x)$, which means $\forall x_1, x_2 \in R^n, \exists L > 0$ satisfies $|\nabla g(x_1) - \nabla g(x_2)| \leq L|x_1 - x_2|$. Then, set $t = \frac{1}{L}$.

Nevertheless, according to the reference [24], for the objective Equation (3), the convergence rate of $f(x_k) - f(x^*)$ is $O(\frac{1}{k})$ and so is the time complexity of ISTA. Fast iterative soft threshold algorithm (FISTA) is created using Nesterov acceleration technology to speed up the convergence speed of the algorithm. The difference between FISTA and ISTA lies in the Nesterov accelerating process, which only requires a few additional steps and brings great elevation to the convergence speed of the algorithm. ISTA mainly relies on the value of approximate function x_{k-1} from the previous iteration and the Function (4) is the only kernel of the algorithm. FISTA makes use of the Nesterov acceleration technology and it mainly consists of two parts. The first part is computing a new point along the direction of the previous two steps. In the second part, the cardinal point is processed by approximate gradient method. The core formula of the algorithm is shown in Formula (5).

$$\begin{cases} x_{k+1} = S_{\lambda t}(y_k - t_k \nabla f(y_k)) \\ \zeta_k = \frac{1 + \sqrt{1 + 4\zeta_{k-1}^2}}{2} \\ \gamma_k = \frac{\zeta_{k-1} - 1}{\zeta_k} \\ y_k = (1 - \gamma_k)x_k + \gamma_k x_{k-1} \end{cases}, \tag{5}$$

Therein, ζ and γ are two momenta, k represents the k -th iteration and we set $\zeta_1 = 1, y_1 = x_0 \in R^n$.

3. Proposed Method

3.1. Parameter Variation Based on Nesterov

As seen in the previous section, there are the following calculation rules in the FISTA

$$\zeta_k = \frac{1 + \sqrt{1 + 4\zeta_{k-1}^2}}{2}, \gamma_k = \frac{\zeta_{k-1} - 1}{\zeta_k}, \tag{6}$$

Obviously, each change of x_k is related to the value of the γ_k and the γ_k is constantly affected by the ζ_k . Two constants "1" and one constant "4" are required for each update in the calculation rules for ζ_k . In this paper, the three fixed constants are replaced by three parameters p, q and r to study the impact of those three parameters on the iteration process. Therefore, the calculation rules that integrate the three parameters p, q and r are modified as follows:

$$\zeta_k = \frac{p + \sqrt{q + r\zeta_{k-1}^2}}{2}, \gamma_k = \frac{\zeta_{k-1} - 1}{\zeta_k}, \tag{7}$$

First, analyzing the convergence is necessary and here is the specific analysis:

$$\begin{aligned} r \in (0, 4) : \zeta_k &\rightarrow \frac{2p+\Delta}{4-r} < +\infty, \gamma_k \rightarrow \frac{2p+\Delta-(4-r)}{2p+\Delta} < 1, \\ r = 4 : \zeta_k &\approx \frac{k+1}{2}p \rightarrow +\infty, \gamma_k \rightarrow 1 \end{aligned} \tag{8}$$

where $\Delta \stackrel{def}{=} \sqrt{rp^2 + (4-r)q}$. It indicates that only when the value of r is 4 can the γ_k converge to 1 and the convergence rate of $f(x_k) - f(x^*)$ is $O(\frac{1}{k^2})$. It also has the advantage of proving the convergence in the iterative process thanks to this constant. So we could set r as 4 for the algorithm. The values of p and q , on the other hand, can have a difference on the

convergence of the algorithm to a certain extent. Let x^* be the optimal solution of the lasso minimum problem and lead x^* and x_k into Formula (2) to get the solution of their respective minimum errors. According to relevant research, let $f(x) = \frac{1}{2}\|y - \phi x\|_2^2 + \lambda\|x\|_1^2$, then the convergence of $f(x_k) - f(x^*)$ is consistent with that of $\{x_k\}$.

The data in the Figure 2 are made up of random signals and the findings are achieved through several experiments. It can be seen clearly that FISTA has an oscillation problem, which has a detrimental impact on the number of iterations, resulting in time waste and a drop in reconstruction efficiency.

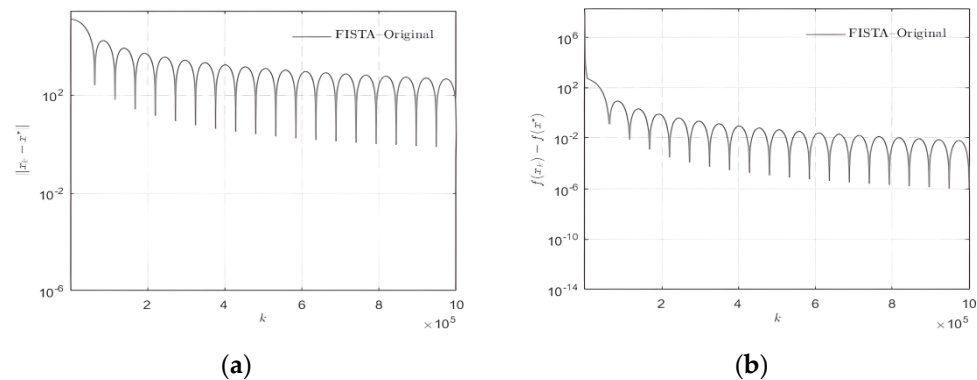


Figure 2. Iteration of conventional FISTA. (a) $\|x_k - x^*\|$. It depicts the reconstruction error diagram of x_k and x^* in each iterative process under the traditional fast iterative soft threshold algorithm; (b) $f(x_k) - f(x^*)$ shows the difference between the value of x_k obtained in each iteration and x^* in solving the lasso problem shown in Formula (2).

Relevant studies have proposed that some judgment restart mechanisms be implemented at the end of each iteration process to alleviate these problems. According to the scheme presented in reference [18], the restart mechanism can be incorporated and forms RadaFISTA, which means that FISTA is regarded as a generalized gradient scheme and Formula (9) is a generalized gradient step. By such mechanism, the new algorithm is able to achieve an almost monotonic convergence in terms of $f(x_k) - f(x^*)$ and can get a significantly faster speed.

$$x_{k+1} = S_{\lambda t}(y_k - t_k \nabla f(y_k)), \quad (9)$$

The gradient restart scheme is equivalent to making judgement on $(y_k - x_{k+1})^T (x_{k+1} - x_k) > 0$ before each iteration. At the same time, as resetting y_k in each iteration, it can never be neglected to regulate the value of r and add a factor between 0 and 1 to adjust and control the range of r value within 4.

When determining the value of these two parameters, p and q , we first fix that they must both be positive. Then, several data pairs composed of the values of p and q are formed and make up a data set. To achieve the suitable values for the parameters p and q , we use hundreds of groups of random signals as numerical instances and conduct 100 experiments on each group of signals. As shown in the Figures 3 and 4, the data in those figures are made up of random signals and the findings are achieved through those experiments.

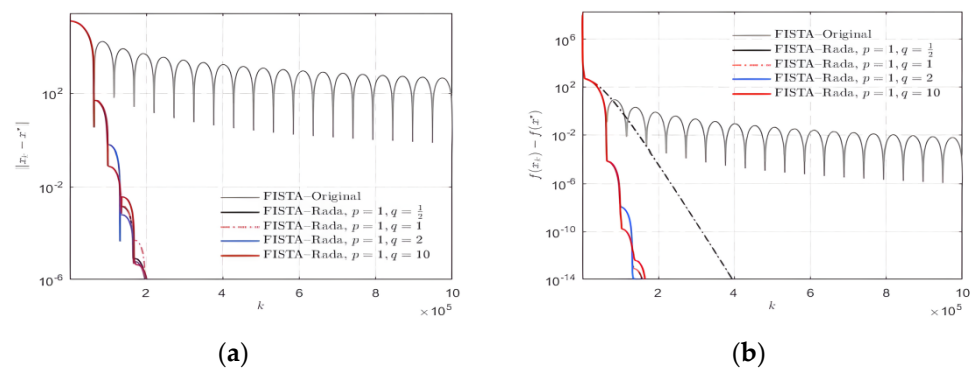


Figure 3. Selection of parameter q value for FISTA–Rada. (a) $\|x_k - x^*\|$; (b) $f(x_k) - f(x^*)$.

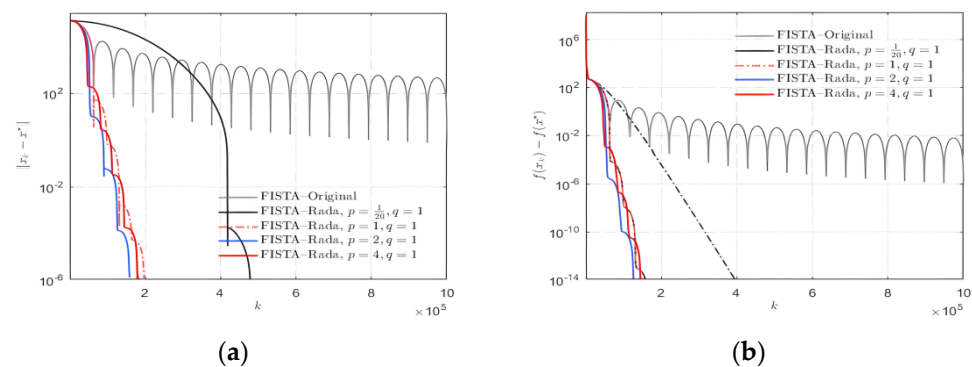


Figure 4. Selection of parameter p value for FISTA–Rada. (a) $\|x_k - x^*\|$; (b) $f(x_k) - f(x^*)$.

On the premise that the value of p is fixed at 1, take the q values of $\frac{1}{2}$, 1, 2 and 10, respectively, for relevant experiments. As shown in Figure 3, the value of q has little effect on the results in those multiple experiments. Consequently, in order to facilitate the implementation of the following experiments, we set the value of q as 1 in the algorithm.

The choice of p value, on the other hand, has an impact on the convergence. Keep the q value constant and take p values of $\frac{1}{20}$, 1, 2 and 4, respectively, for the experiment. The results are exhibited in Figure 4.

Through repeated tests, it is seen that when the value of p is taken as 2, relatively still less data residuals can be obtained in fewer iteration times. The effect of taking p value around 2 cannot be achieved by other values. Therefore, in order to bring convenience for the following experiments, the value of p is set to 2 in the improved algorithm given in this paper.

Apparently, the parameters p and q appear to be able to control the reconstruction process after being added into the algorithm. In comparison to FISTA, this algorithm proposed in this paper can more easily control the convergence rate and the reconstruction speed by adjusting the parameters p and q , making it more versatile and adaptable to signals with different characteristics.

3.2. Improved Threshold Function

For the sake of seeking the “gradient” of the objective function containing the L1 norm, a soft threshold function is adopted to calculate the optimal solution. Evidently, the soft threshold function is widely used and has shown to be extremely reliable. However, both the soft threshold function and the hard threshold function have flaws. Consequently, this paper proposes a new threshold function that combines the features of soft and hard threshold functions and applies them throughout the reconstruction algorithm.

$$\hat{x}_i = \begin{cases} 0, & |x_i| < \lambda \\ x_i(1 - (\frac{\lambda}{|x_i|})^n), & |x_i| \geq \lambda \end{cases} \quad (10)$$

Therein, x_i is the processed values of y_k and n is a variable quantity that gives the enhanced threshold function a lot of flexibility. Furthermore, the threshold function proposed in this paper also has the advantage of a concise expression, which avoids the problems of multiple parameters and significant inconvenience that many threshold functions have to some extent. Additionally, additional advantages are as follows. In terms of continuity, it can be seen from Formula (10) that the function is continuous at $\pm\lambda$, which allows it to avoid the defects of the hard threshold function and to smooth out the signal. Additionally, when the coefficient x_i tends to infinity, the deviation of the soft threshold function will lead to distortion. However, in this improved function, it can be seen that $\lim_{x_i \rightarrow +\infty} \hat{x}_i - x_i = 0$,

$\lim_{x_i \rightarrow -\infty} \hat{x}_i - x_i = 0$. Hence, the improved threshold function described in this study can partially overcome the constant deviation of the soft threshold function and it is also verified that the deviation will have less and less influence on the updated threshold function.

Figure 5 illustrates that along with the increase of x_i , applying the improved threshold function described in this paper results in greater sparse constraints. The new function produces smaller deviations for large coefficients x_i than the soft threshold function. By keeping large coefficients x_i , we could overcome the problem of easy distortion of the soft threshold function and by shrinking the intermediate coefficients x_i , we could reduce the discontinuity of the hard threshold function. When the value of n is 1, the function is a soft threshold function. When the value of n approaches infinity, the value of x_i approaches the value of the hard threshold function. To some extent, the better function compensates for the shortcomings of constant deviation. Furthermore, after processing, the estimated coefficient \hat{x}_i obtained can approach the real value more quickly. Meanwhile, it also makes up for the discontinuity of the hard threshold function at the threshold $\pm\lambda$.

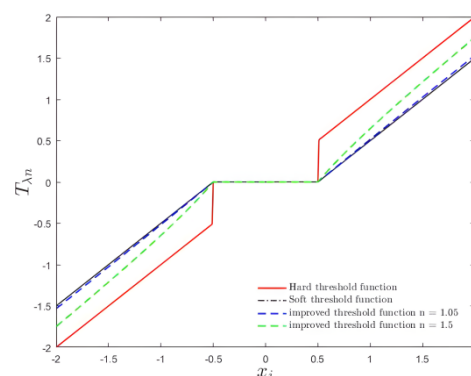


Figure 5. Threshold function diagram.

As a result, an improved threshold function is constructed for the lasso problem in the compressed sensing reconstruction algorithm. In the reconstruction algorithm, the iterative threshold algorithm attempts the solution to the lasso problem indicated in Formula (2). The soft threshold function, as seen in Section 2.2, is frequently utilized to solve such reconstruction problems. However, the improved function is used to displace the soft threshold function in this paper and is integrated into the FISTA core formula.

$$x_{k+1} = T_{\lambda n}(y_k - t_k \nabla f(y_k)), \quad (11)$$

In this formula, $T_{\lambda n}$ denotes the improved threshold function, λ is the threshold, n is the function variable and t_k is the step size. These new penalty functions can be designed flexibly by changing the value of n in order to obtain a better reconstruction effect.

3.3. Improved Fast Iterative Threshold Algorithm

Aiming at improving the convergence speed and the reconstruction performance of FISTA, this paper puts forward a new algorithm named fast iterative parametric improved

threshold algorithm (FIPITA), which substitutes the improved threshold function for the soft threshold function, uses a restart and self-adaptive adjustment mechanism to alleviate the oscillation problem in the reconstruction process and improve the convergence efficiency and integrates three parameters p , q and r to replace three previous constants. The final reconstructed signal can be obtained by combing the three elements above into the FIPITA.

At last, in order to facilitate the description and understanding of the execution steps of the above algorithm, the solving process is organized in the form of the following Algorithm 1.

Algorithm 1 Fast iterative parametric improved threshold algorithm(FIPITA)

Input :

Lipschitz constant: $L = L(f)$ ($L(f)$ -A Lipschitz constant of ∇f)

Initial value: x_0

Output:

Optimal value $f(x)$ with x .

1: **Begin**

2: Initialize momentum1 ζ , momentum2 γ ,

3: Set $\zeta_1 = 1; y_1 = x_0 \in R^n$;

4: **For** $k = 1, 2, 3 \dots$ **COMPUTE**

5: $x_{k+1} = T_{\lambda n}(y_k - t_k \nabla f(y_k));$

6: $\zeta_k = \frac{p + \sqrt{q + r\zeta_{k-1}^2}}{2}$

7: $\gamma_k = \frac{\zeta_{k-1} - 1}{\zeta_k}$

8: $y_k = (1 - \gamma_k)x_k + \gamma_k x_{k-1}$

9: **Restart if** $(y_k - x_{k+1})^T (x_{k+1} - x_k) > 0$

10: **Let** $r = \zeta r$

11: $y_k = x_k$

12: **if** $r < 3.99$

13: **Reset** ζ_k

14: **End For**

15: Obtain the optimal LASSO answer $f(x)$ and its corresponding x ;

16: **End**

4. Results and Discussions

Simulation experiments used to test and validate the fast iterative parametric improved threshold algorithm (FIPITA) are divided into several sections. These sections include comparing the signal before and after reconstruction to assess the reconstruction effectiveness, adopting the residual rate to assess the change in reconstruction accuracy with sparsity, adopting the required maximum number of iterations to assess the change in algorithm efficiency with sparsity and adopting the residual rate to assess the variation in reconstruction accuracy with the observed value. FISTA, the most classic and widely used algorithm, is added in the experiment as a comparable algorithm to further demonstrate the performance of the algorithm. Meanwhile, RestartFISTA [25] and RadaFISTA [18] are also added.

4.1. One Dimensional Signal Reconstruction Simulation Test

To demonstrate the property of the FIPITA, we take a Gaussian random signal x with length n of 256, observation value m of 128 and sparsity K of 10. The Gaussian random matrix is selected as the observation matrix of this simulation experiment. When the error of two adjacent iterations, $res = x_k - x_{k-1}$, is less than 10^{-16} or the value of $y - \phi x_k$ is less than 10^{-6} , the algorithm has to call off the iteration.

From Figure 6, it is be indicated that the signal can be rebuilt accurately by the mended algorithm.

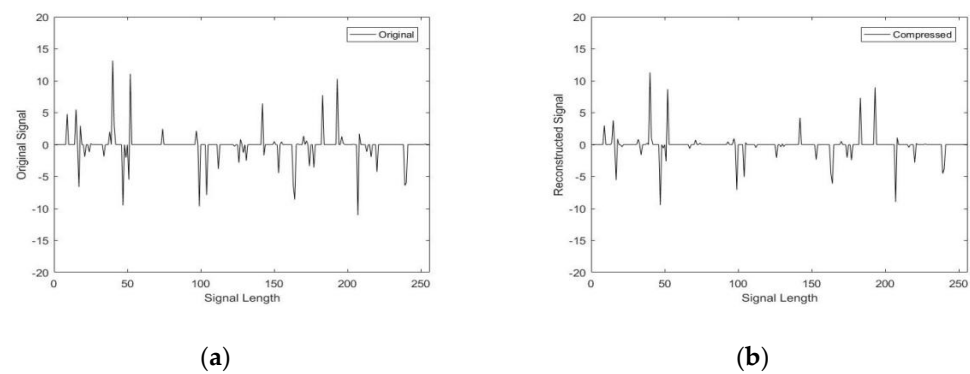


Figure 6. Signal reconstruction diagram. (a) Original signal; (b) the reconstructed signal.

4.2. Performance of the Algorithm under Different Sparsity

The residual rate is a key verification index for evaluating the algorithm performance and it is defined as follows:

$$\text{residualrate} = \frac{\|\hat{x} - x\|}{\|x\|}, \quad (12)$$

Among Formula (12), \hat{x} stands for the final reconstruction result, x is the original signal and the ratio of the norm of $\hat{x} - x$ to the norm of the original signal x is deduced as the index of signal reconstruction quality; the smaller the index the higher the signal reconstruction accuracy.

FISTA, RadaFISTA and RestartFISTA are drawn into the experiment to compare with the FIPITA. Gaussian and Hadamard matrices are selected as observation matrices for measurement experiments at the same time. The type, length of the signal and the observation value are identical to those stated in the previous section. The sparsity K is set between 21 and 70 and the step size is 1. The same group of signals is reconstructed by those four algorithms.

According to Figure 7a,b, the residual rate of the modified iterative threshold algorithm is the lowest and does not change with sparsity or observation matrix. Overall, the residual rates of the four algorithms are steadily increasing in tandem with the increase in sparsity, which is in keeping with the regular pattern of the algorithm. Moreover, as shown in the picture above, the residual rate of the FISTA, RadaFISTA and RestartFISTA are pretty close, that is, the reconstruction effects of these three algorithms are similar. Taking the Gaussian matrix as the observation matrix and the interval of sparsity between 21 and 70 as an example, the residual rate of the FIPITA proposed in this paper is approximately 6.35% lower than the other three algorithms. According to the definition of residual rate, the lower the residual rate the higher the reconstruction accuracy, which suggests the reconstruction accuracy of the FIPITA is supreme.

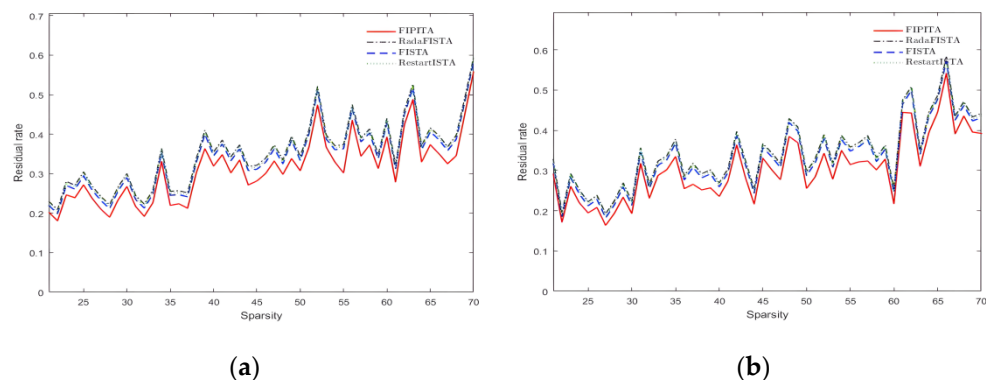


Figure 7. Signal residual rate under different sparsity. (a) Residual rate of Hadamard matrix; (b) residual rate of Gaussian matrix.

4.3. Performance of the Algorithm under Different Observation Numbers

The original signal is a Gaussian random signal with the same signal type and length as the ones above. The observation value m is set to range between 75 and 125, the step size is 1 and the sparsity K is set to 50. The FISTA, RadaFISTA and RestartFISTA are still used to reconstruct the same group of signals with FIPITA and the Gaussian matrix and Hadamard matrices are used as the observation matrices, respectively. The measured residual rate is shown in the graphs below.

Although the observation value is varied, the residual rate of the FIPITA is the minimum among the four algorithms under different measurement matrices, as is clearly shown in Figure 8a,b, and the residual rate of those three algorithms used for comparison is close. Taking the Hadamard matrix as the observation matrix and the interval of observation value between 75 and 125 as an example, the residual rate of FIPITA in this study is around 4.99% lower than the other three algorithms. In other words, the FIPITA has the maximum reconstruction accuracy, whereas the other three algorithms have similar reconstruction accuracy.

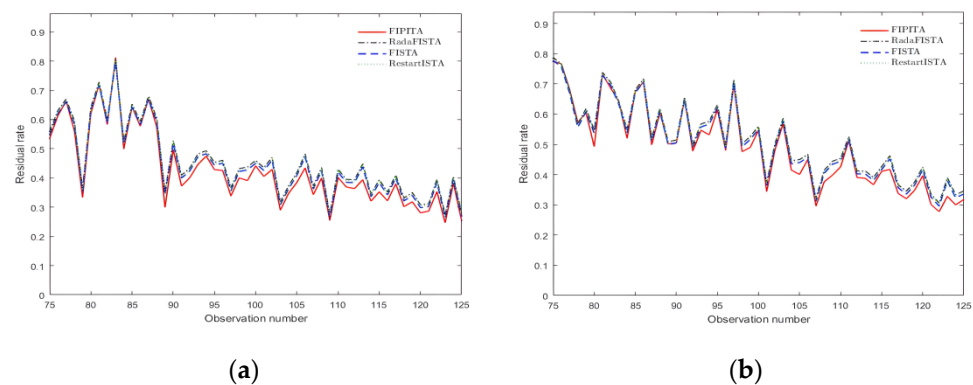


Figure 8. Signal residual rate under different observation numbers. (a) Residual rate of Hadamard matrix; (b) residual rate of Gaussian matrix.

4.4. Algorithm Efficiency Comparison

In this paper, the maximum number of iterations required by the algorithm is adopted as an index to measure the efficiency of the algorithm. The maximum number of iterations indicates the number of final iterations required when $res = x_k - x_{k-1}$ is less than 10^{-16} and the iteration is ceased. The efficiency of the algorithm improves as the number of iterations decreases, which means the reconstruction will be faster. Figure 9a,b shows the number of iterations for varied sparsity and observation matrices.

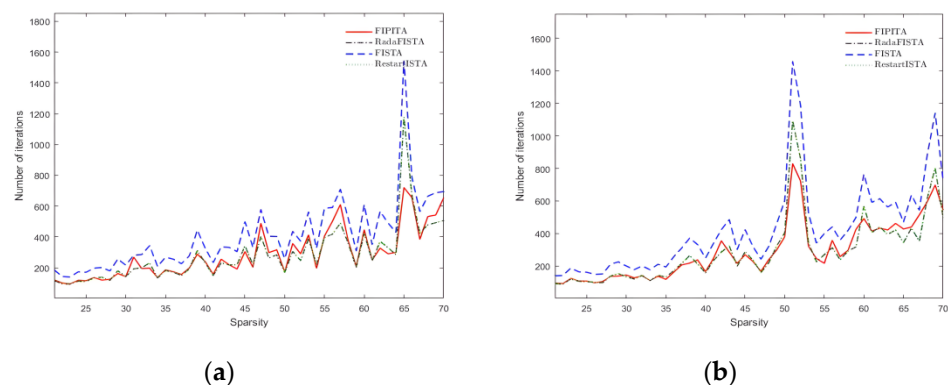


Figure 9. Required number of iterations under different sparsity. (a) Maximum number of iterations of Hadamard matrix; (b) maximum number of iterations of Gaussian matrix.

In light of Figure 9a,b, it is explicit that the number of iterations required by FISTA is the highest for various sparsity, implying that the algorithm requires more time cost and is inefficient. The FIPITA appears to be better than FISTA in terms of the number of iterations, but at the same time as sparsity increases the number of iterations required by the FIPITA is sometimes higher than RadaFISTA and RestartFISTA. For this reason, this paper integrates data from one particular experiment on the Hadamard matrix and displays them in the Table 1 below.

Table 1. Comparison of reconstruction residual rate and iteration numbers of four algorithms.

	Average Iteration Times under Different Sparsity	Sparsity (=30)		Sparsity (=45)		Sparsity (=50)	
		Number of Iterations	Residual Rate	Number of Iterations	Residual Rate	Number of Iterations	Residual Rate
FISTA	397.66	222	0.2861	346	0.2605	508	0.3609
RadaFISTA	273.82	150	0.2861	240	0.2602	357	0.3612
RestartFISTA	273.82	149	0.2860	237	0.2608	359	0.3610
FIPITA	295.92	145	0.2592	253	0.2423	358	0.3427

The statistics from Table 1 leads us to the conclusion that, while FIPITA will result in marginally higher iteration times than RadaFISTA and RestartFISTA, such iteration times are inherently unstable. When the sparsity is 30, for example, the number of its iterations is much less than the other three algorithms, including FISTA, and when the sparsity is 50, its iteration times are similar to RadaFISTA and RestartFISTA. At the same time, when it comes to residual rate, the FIPITA has a clear advantage, which can improve the accuracy of the compressed sensing reconstruction algorithm.

It is also important to combine the number of iterations with convergence. Several hundreds of experiments are done on the same group of signals to study the convergence.

The horizontal axis of Figure 10 denotes the number of iterations. We can see that the results of the figure are consistent with those analyzed above. The convergence rate and the iterations of FIPITA are similar to RadaFISTA and RestartFISTA and superior to FISTA. In addition, the convergence rate of $f(x_k) - f(x^*)$ is $O(\frac{1}{k^2})$, which is coincident with those three.

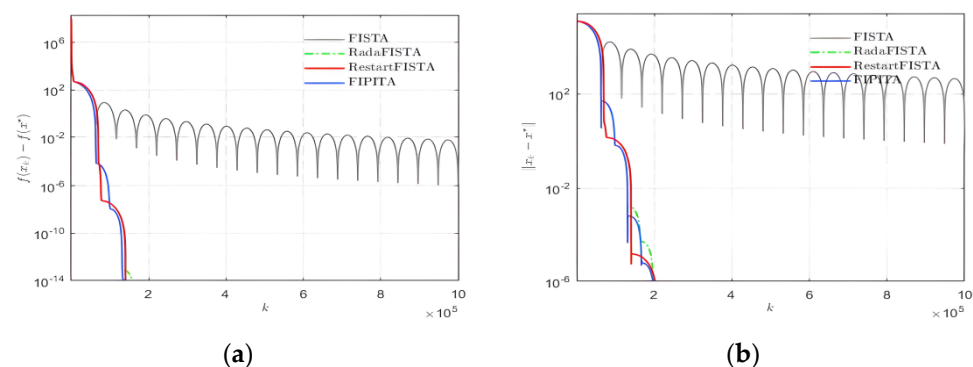


Figure 10. Iteration times and convergence. (a) $f(x_k) - f(x^*)$; (b) $\|x_k - x^*\|$.

Generally speaking, while the number of iterations required by the suggested technique is significantly fewer than that of FISTA, it is slightly more than RadaFISTA and RestartFISTA. However, after thoroughly evaluating and judging the unstable nature of the number of iterations, lower residual rate and superior reconstruction performance of FIPITA in a comprehensive way, we can get a conclusion that the FIPITA not only has relatively rapid reconstruction speed but also can increase signal reconstruction accuracy.

5. Conclusions

The fast iterative parametric improved threshold algorithm (FIPITA) has combined the restart mechanism with the idea of backtracking and added three parameters so that it can converge faster than the FISTA. Furthermore, instead of using the soft threshold function to lower the residual rate of the algorithm, this paper uses the enhanced threshold function to improve the reconstruction accuracy. Through experiments, the effectiveness of the algorithm is tested and confirmed. The results reveal that this algorithm is not only better than several comparable algorithms in terms of reconstruction accuracy but also considerably superior to FISTA in terms of algorithm efficiency. Therefore, the next research project should focus on how to further improve the efficiency of the reconstruction algorithm while still ensuring reconstruction accuracy.

Author Contributions: Conceptualization, J.W.; methodology and software, S.M.; formal analysis, S.M. and J.D.; writing—original draft preparation, S.M. and Z.W.; writing—review and editing, W.H. and Y.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported in part by the Major Project of Philosophy and Social Science Research in Jiangsu Universities of China (2020SJZDA102), the Future Network Scientific Research Fund Project (FNSRFP-2021-YB-54) and Tongda College of Nanjing University of Posts and Telecommunications (XK203XZ21001).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data of this study are available from the corresponding author upon request.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Donoho, D.L. Compressed sensing. *IEEE Trans. Inf. Theory* **2006**, *52*, 1289–1306. [\[CrossRef\]](#)
2. Candes, E.J.; Romberg, J.K.; Tao, T. Stable signal recovery from incomplete and inaccurate measurements. *Commun. Pure Appl. Math. J. Issued Courant Inst. Math. Sci.* **2006**, *59*, 1207–1223. [\[CrossRef\]](#)
3. Wei, P.; He, F. The compressed sensing of wireless sensor networks based on Internet of Things. *IEEE Sens. J.* **2021**, *21*, 25267–25273. [\[CrossRef\]](#)
4. Xifilidis, T.; Psannis, K.E. Correlation-based wireless sensor networks performance: The compressed sensing paradigm. *Clust. Comput.* **2022**, *25*, 965–981. [\[CrossRef\]](#)
5. Lin, D.; Min, W.; Xu, J.; Yang, J.; Zhang, J. An Energy-efficient Routing Method in WSNs Based on Compressive Sensing: From the Perspective of Social Welfare. *IEEE Embed. Syst. Lett.* **2020**, *13*, 126–129. [\[CrossRef\]](#)
6. Sekar, K.; Suganya Devi, K.; Srinivasan, P. Energy efficient data gathering using spatio-temporal compressive sensing for WSNs. *Wirel. Pers. Commun.* **2021**, *117*, 1279–1295. [\[CrossRef\]](#)
7. Prabha, M.; Darly, S.S.; Rabi, B.J. A novel approach of hierarchical compressive sensing in wireless sensor network using block tri-diagonal matrix clustering. *Comput. Commun.* **2021**, *168*, 54–64. [\[CrossRef\]](#)
8. Li, L.; Fang, Y.; Liu, L.; Peng, H.; Kurths, J.; Yang, Y. Overview of compressed sensing: Sensing model, reconstruction algorithm, and its applications. *Appl. Sci.* **2020**, *10*, 5909. [\[CrossRef\]](#)
9. Blumensath, T.; Davies, M.E. Iterative Thresholding for Sparse Approximations. *J. Fourier Anal. Appl.* **2008**, *14*, 629–654. [\[CrossRef\]](#)
10. Wright, S., Jr.; Nowak, R.D.; Figueiredo, M.A. Sparse reconstruction by separable approximation. In Proceedings of the 33rd IEEE International Conference on Acoustics, Speech and Signal Processing, Las Vegas, NV, USA, 30 March–4 April 2008. [\[CrossRef\]](#)
11. Netrapalli, P. Stochastic gradient descent and its variants in machine learning. *J. Indian Inst. Sci.* **2019**, *99*, 201–213. [\[CrossRef\]](#)
12. Mathew, A.; Amudha, P.; Sivakumari, S. Deep Learning Techniques: An Overview. In Proceedings of the International Conference on Advanced Machine Learning Technologies and Applications, Jaipur, India, 13–15 February 2020. [\[CrossRef\]](#)
13. Tibshirani, R.J. The lasso problem and uniqueness. *Electron. J. Stat.* **2013**, *7*, 1456–1490. [\[CrossRef\]](#)
14. Babapour, S.; Lakestani, M.; Fatholahzadeh, A. AFISTA: Accelerated FISTA for sparse signal recovery and compressive sensing. *Multimed. Tools Appl.* **2021**, *80*, 20707–20731. [\[CrossRef\]](#)
15. Lazzaretti, M.; Rebegoldi, S.; Calatroni, L.; Estatico, C. A scaled and adaptive FISTA algorithm for signal-dependent sparse image super-resolution problems. In Proceedings of the International Conference on Scale Space and Variational Methods in Computer Vision, Virtual Event, Cabourg, France, 16–20 May 2021. [\[CrossRef\]](#)
16. Tong, C.; Teng, Y.; Yao, Y.; Qi, S.; Li, C.; Zhang, T. Eigenvalue-free iterative shrinkage-thresholding algorithm for solving the linear inverse problems. *Inverse Probl.* **2021**, *37*, 065013. [\[CrossRef\]](#)

17. Beck, A.; Teboulle, M. Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems. *IEEE Trans. Image Process.* **2009**, *18*, 2419–2434. [[CrossRef](#)] [[PubMed](#)]
18. O’donoghue, B.; Candes, E. Adaptive restart for accelerated gradient schemes. *Found. Comput. Math.* **2015**, *15*, 715–732. [[CrossRef](#)]
19. Aujol, J.F.; Dossal, C.; Labarrière, H.; Rondepierre, A. FISTA restart using an automatic estimation of the growth parameter. *HAL Sci. Ouvert.* 2021; *preprint*.
20. Beck, A.; Teboulle, M. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM J. Imaging Sci.* **2009**, *2*, 183–202. [[CrossRef](#)]
21. Mao, A.; Li, Y.; Feng, D.; Pan, W.; Li, M. Scanning Measurement Based on FISTA Phase Calibration. In Proceedings of the 2021 4th International Conference on Information Communication and Signal Processing (ICICSP), Shanghai, China, 24–26 September 2021. [[CrossRef](#)]
22. Li, C.; Liu, X. Seismic Reflectivity Inversion Using an Adaptive FISTA. In Proceedings of the Second EAGE Conference on Seismic Inversion, Porto, Portugal, 7–9 February 2022. [[CrossRef](#)]
23. Liu, Z.; Liao, X.; Wu, J. Image reconstruction for low-oversampled staggered SAR via HDM-FISTA. *IEEE Trans. Geosci. Remote Sens.* **2021**, *60*, 1–14. [[CrossRef](#)]
24. Molinari, C.; Liang, J.; Fadili, J. Convergence rates of forward–douglas–rachford splitting method. *arXiv* **2018**, arXiv:1801.01088. [[CrossRef](#)] [[PubMed](#)]
25. Liang, J.; Luo, T.; Schönlieb, C.B. Improving “Fast Iterative Shrinkage-Thresholding Algorithm”: Faster, Smarter and Greedier. *arXiv* **2018**, arXiv:1811.01430. [[CrossRef](#)]