



Iterative Selection and Correction Based Adaptive Greedy Algorithm for Compressive Sensing Reconstruction

Ahmed Aziz^{1,2} · Walid Osamy^{1,3} · Ahmed M. Khedr^{4,5}

Accepted: 14 September 2020 / Published online: 3 October 2020
© Springer Science+Business Media, LLC, part of Springer Nature 2020

Abstract

Compressive sensing (CS) is a new sampling theory used in many signal processing applications due to its simplicity and efficiency. However, signal reconstruction is considered as one of the biggest challenge faced by the CS method. Therefore in this paper, we aim to address this challenge by proposing an Adaptive Iterative Forward–Backward Greedy Algorithm (AFB). AFB algorithm is different from all other reconstruction algorithms, as it depends on solving the least squares problem in the forward phase, which increases the probability of selecting the correct columns better than other reconstruction algorithms. In addition, AFB improves the selection process by removing the incorrect columns selected in the previous step. To evaluate the AFB’s reconstruction performance, we used two types of data: computer-generated data and real data set (Intel Berkeley data set). The simulation results show that AFB outperforms Forward–Backward Pursuit, Subspace Pursuit, Orthogonal Matching Pursuit, and Regularized OMP in terms of reducing reconstruction error.

Keywords IoT · WSNs · Compressive sensing · Reconstruction algorithms · Greedy algorithms

✉ Ahmed Aziz
a.mohamed@tsue.uz

Walid Osamy
W.elsherif@qu.edu.sa

Ahmed M. Khedr
amkhedr@zu.edu.eg

¹ Computer Science Department, Faculty of Computers and Artificial Intelligence, University of Benha, Benha, Egypt

² Tashkent State University of Economic, Tashkent, Uzbekistan

³ Department of Applied Natural Science, College of Community in Unaizah, Qassim University, Qassim, Kingdom of Saudi Arabia

⁴ Computer Science Department, University of Sharjah, Sharjah 27272, UAE

⁵ Faculty of Science, Zagazig University, Zagazig, Egypt

1 Introduction

Compressive sensing (CS) [1–5] method has been proposed as a novel data reduction method for reducing the size of data transmitted through the IoT network. According to CS method, the base station (BS) needs only $M \geq K \log N/K$, where M is the size of compressed samples, K is the sparsity level and N is the signal dimension, to recover the original signal $x \in R^N$ from only $y \in R^M$ measurements such that $y = \Phi x$ and Φ is CS matrix. On the other hand, the CS reconstruction process aims to recover N samples from only M measurements, where $M \ll N$, which makes it NP-hard problem [6]. The CS reconstruction problem can be expressed as follows:

$$\min_x \|x\|_0 \text{ s.t. } y = \Phi x \quad (1)$$

In Eq. 1, the CS reconstruction process aims to recover the sparsity level of the original signal x such that the CS matrix Φ and the measurements vector y are given. A lot of algorithms have been proposed to address this problem such as convex relaxation and greedy algorithm. In convex relaxation algorithms, the problem in (1) is relaxed by replacing L_0 to L_1 [7] as follows:

$$\min_x \|x\|_1 \text{ s.t. } y = \Phi x \quad (2)$$

and then use convex problem solvers such as L_1 -magic toolbox [8] to solve the problem in Eq. 2. Although the convex optimization based reconstruction algorithms have the stability and the ability to reconstruct the full signal correctly, they suffer from highly complex computations that make them unsuitable for IoT network. On the other hand, Greedy algorithms present themselves as sufficient reconstruction algorithms. During the greedy algorithm reconstruction process, one or more CS matrix Φ 's columns are iteratively selected based on their correlation with the current residual. There are different greedy algorithms that can be used, such as OMP [9] algorithm, in which one column is selected from Φ and then its orthogonality is removed from the current residual. This process is repeated until we obtain the estimated signal x' . Based on OMP algorithm, a lot of algorithms have been proposed such as ROMP [10] and StOMP [11]. In addition to this, algorithms such as CoSaMP [12], SP [13], IHT [14] and FBP [15], which use backward steps to prune the wrong elements that have been added during the forward step, also received significant attention. All of these algorithms are sufficient but cannot provide the optimal solution.

In this paper, to improve the signal reconstruction process, we propose a new iterative greedy algorithm called Adaptive Iterative Forward–Backward Greedy Algorithm (AFB). AFB is considered as a reversible greedy algorithm that follows a reversible construction so that, the support-set can be pruned (backward step) in order to remove the unreliable elements selected in the past (forward step).

The rest of the paper is organized as follows: sect. 2 briefly review the related works. In Sect. 3, we introduce our approach to carry out the proposed problem. In SecSect. 4, we give an example scenario. The simulation of our approach is presented in Sect. 5. In Sect. 6, we conclude our work.

2 Related Research

During the past few years, CS signal reconstruction problem has gained significant attention from researchers and a lot of efforts have been done to enhance the reconstruction process at the BS side. Due to the high correlation between IoT sensor nodes, these sensors' data have high sparsity property. CS method utilizes this property to reduce the signal dimension from N to vector M such that $M \ll N$. On the other hand, during the CS reconstruction process, it is required to reconstruct N from M , which makes it NP-hard problem. To address this problem, many reconstruction algorithms have been proposed. CS reconstruction algorithms can be classified into two categories: Convex optimization and Greedy based algorithms. In convex optimization based reconstruction algorithms, the CS reconstruction problem formulated as L_1 norm is solved using any linear programming methods. Convex optimization based reconstruction algorithms have the ability to recover the original signal correctly. However, they suffer from high and complex computation processes which inhibit them from being the best reconstruction choice for the IoT networks. In contrast, GA based reconstruction algorithms could be more suitable for IoT networks as they provide the same efficiency and performance in reconstruction through moderate computations. In [16], Matching Pursuit algorithm (MP) is considered as the first GA based algorithm in which the support-set is initialized by the index of the largest magnitude entries in $\Phi'y$ during the forward step and then it solves the least squares problem. However, MP algorithm doesn't consider the non-orthogonality of the CS matrix, which leads to incorrect selection of columns corresponding to the nonzero values. This drawback has been solved by the OMP algorithm [9]. In each iteration of OMP, the algorithm selects the index of the largest magnitude entries in $\Phi'r$, where r is the residual of y , and then solves the least squares problem. Different algorithms have been proposed based on OMP algorithm as in [10, 11]. In [11], a faster and enhanced version of OMP is proposed which is called Stagewise OMP (StOMP). StOMP enhances the forward step of OMP by selecting a number of columns, instead of one column as in OMP, the magnitude values of the columns in $\Phi'r$ are greater than a threshold and then uses these columns to solve the least squares problem. In [10], in each iteration, the proposed algorithm finds the same inner-product magnitudes and then collects them in sets to select the maximum energy set.

The above algorithms are classified as irreversible GA class as they do not have a backward step. This prevents such algorithms to remove the elements selected wrongly during the forward step. I.e., in these algorithms, there is no option to remove any element that is already added to the support set. On the other hand, reversible greedy reconstruction algorithms such as CoSaMP [12], SP [13], IHT [14] and FBP [15] algorithms use the backward step to prune the wrong elements that have been added to the support set during the forward step. CoSaMP and SP algorithms initialize the support-set by adding the indices of b largest amplitude values of $\Phi'y$. The size of b is different in each algorithm, for example $b = K$ in SP and $b = 2K$ in CoSaMP, where the value of sparsity level K is known. On the other hand, FBP [15] algorithm has the ability to perform without the knowledge of K . It assigns forward and backward step size depending on the measurement size. In [14], IHT algorithm considers iterative gradient search algorithm which updates the estimate-set depending on e gradient of the residue and keeps only the largest K entries by removing the wrong selection.

3 Adaptive Iterative Forward–Backward Greedy Algorithm (AFB)

In this section, a new greedy based reconstruction algorithm called Adaptive Iterative Forward–Backward Greedy Algorithm (AFB) is proposed. AFB algorithm can be used by the BS to reconstruct the sensor readings again. Before describing these algorithms, we define the notations that we use to represent the operations in the proposed algorithm.

$$\text{resid}(y, x) \triangleq y - \Phi x \quad (3)$$

$$\text{supp}(x, k) \triangleq \{ \text{the set of indices corresponding to the } k \text{ largest amplitude components of } x \}, \quad (4)$$

Algorithm 1 AFB Algorithm

```

1: INPUT: CS matrix:  $\Phi$ , Compressed Samples vector:  $y$ 
2: OUTPUT: S: sparse approximation  $D^L$  of the original signal
Algorithm Parameter initialization:
3:  $r^0 = y$  {the residual initialization}
4:  $D^0 = 0$  {the approximation set initialization}
5:  $A = \phi$  {the estimated set initialization}
6:  $U = \phi$  {candidate set initialization}
7:  $Z = \phi$  {least squares signal approximation set}
8:  $L = 0$  {(number of rounds)}
9: while true do
10:   $L = L + 1$  {round increment}
    Selection and Estimation Step:
11:   $Z = \Phi_A^\dagger r^{L-1}$  (solve LS problem)
12:   $U = \text{supp}(Z, \eta)$  (Find  $\eta = M/2$  columns Eq.4)
13:   $A = U \cup \text{supp}(D^{L-1})$  (Union of old and new solution)
    Check and Remove Step:
14:   $R|A = \Phi_A^\dagger y$  (find the first solution)
15:   $R|A^c = 0$  (fill the index of non-selected column index with zero)
16:   $D^L = R_{M/3}$  (retaining only  $M/3$ )
    Update step:
17:   $r^L = \text{resid}(y, D^L)$  Eq.3
18:  if  $\|r^L\|_2 < \gamma$  or  $L = L_{max}$  then
19:    break
20:  end if
21: end while

```

3.1 AFB Algorithm Description

In this section, we describe the newly proposed reconstruction algorithm called Adaptive Iterative Forward–Backward Greedy Algorithm (AFB), for successful reconstruction of sensor readings. AFB is a greedy algorithm and it consists of two-processes. In the first process (Selection and Estimation), the estimated set U will be updated by adding η (where η denotes the Selection step size) columns from the CS matrix. These columns have largest

entries in the least-squares signal approximation set Z from Φ as the correct solution of least-squares problem $Z = \Phi^\dagger r^{L-1}$. In the second process (Check and Remove), the estimated set U is clipped by removing v columns which were wrongly selected in the Selection and Estimation process. This accurately identifies the true support of U , where v is called the Removable step size. The described details of AFB algorithm is presented in Algorithm 1.

3.1.1 Algorithm Description

The algorithm includes four steps: Parameter Initialization, Selection and Estimation, Check and Remove, and Update.

Parameter Initialization Residual $r^0 = y$, approximation $D^0 = 0$, Estimated set $A = \phi$, candidate set $U = \phi$ and number of rounds $L = 0$ (Algorithm 1, Lines 3 – 8).

Selection and Estimation In this step, set U will be updated by adding the η ($\eta = M/2$) largest elements from the solution Z of the least squares problem ($Z = \Phi_A^\dagger r^{(L-1)}$). Then extend the set A by adding the components of U to the approximation set $D^{(L-1)}$ (Algorithm 1, Lines 11 – 13).

Check and Remove In this step, the projection coefficients R are computed by the orthogonal projection of y onto submatrix of Φ whose columns are listed in the set A . Then (correction step), the proposed algorithm removes the indices of incorrect columns which were wrongly selected by the Selection and Estimation step, i.e., the algorithm updates the approximation set $D^L = R_{M/3}$ by removing $v = \eta - M/3$ columns indices which have the smallest values in set R (Algorithm 1, Lines 14 – 16).

Update The samples are updated as $r^L = y - \Phi D^L$ and it will be terminated in two situations: (1) the residue set $\|r^L\|_2$ is smaller than γ , the termination parameter. The termination parameter selection process is based on the level of noise (Algorithm 1, Lines 17–18) (2) the number of iterations reached the maximum L_{max} (for example, $L_{max} = M$). At the end of the algorithm, D^L contains the corresponding nonzero values, which is the approximate solution of the original signal.

4 Example Scenario

In this section, we provide a simple example, with step by step explanation of our algorithm.

1. Given the parameters $M = 5, N = 10, S \leq M/2 = 2$, and a randomly generated $M \times N$ sampling matrix Φ from the standard i.i.d. Gaussian ensemble with mean 0 and standard deviation $1/N$, i.e.

$$\Phi_{5 \times 10} = \begin{pmatrix} 0.052 & -0.013 & -0.029 & 0.030 & -0.133 & 0.045 & -1.136 & 0.103 & -0.019 & 0.082 \\ -0.002 & -0.071 & -0.084 & -0.125 & -0.233 & -0.013 & 0.045 & -0.111 & -0.021 & 0.152 \\ -0.003 & 0.135 & -0.112 & -0.086 & -0.144 & -0.018 & -0.084 & 0.126 & -0.030 & 0.046 \\ -0.079 & -0.022 & 0.252 & -0.017 & 0.033 & -0.047 & -0.033 & 0.066 & 0.002 & -0.021 \\ 0.101 & -0.058 & 0.165 & 0.079 & 0.039 & 0.086 & 0.055 & -0.006 & 0.005 & 0.062 \end{pmatrix}$$

2. Select a set G of size $|G| = S$ uniformly at random, and generate the sparse signal vector x by setting all entries of x (such that x is an $N \times 1$ vector) supported on G to ones, referred to as a binary signal, i.e.

$$x = (0 \ 1 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0)$$

3. Finally, compute the measurement y (such that y is an $M \times 1$ vector)

$$y = \Phi x = (-0.0427 \ -0.1562 \ 0.0231 \ 0.2301 \ 0.1066)$$

By applying AFB algorithm with sampled measurement vector y and measurement matrix Φ as inputs, the algorithm executes as follows:

1. *Parameter Initialization* The AFB algorithm is initialized as: $r^0 = y$, $D^0 = 0$, $A = \phi$, $Z = \phi$, $U = \phi$, $\gamma = 10^{-6}$, $\eta = M/2 = 3$, $v = \eta - M/3 = 2$ and $L_{max} = M = 5$.

At iteration $L = 1$: The algorithm will execute the following steps:

2. *Selection and Estimation* This is the first stage of AFB in which:

1. AFB solves the least squares problem:

- $Z = \Phi^\dagger r^0 = (-0.047 \ 0.564 \ 0.843 \ -0.272 \ 0.160 \ 0.130 \ 0.231 \ 0.451 \ -0.043 \ -0.004)$

2. The column indices(2,3,8) will be added to expand A , since these indices have the largest magnitude elements in Z :

- $U = \text{supp}(Z, \eta) = \{2, 3, 8\}$
- $A = U \cup \text{supp}(D^0) = \{2, 3, 8\}$

3. *Check and Remove*:

The projection coefficients R are computed by the orthogonal projection of y onto sub-matrix of Φ whose columns are listed in the set A :

- $R|A = \Phi_A^\dagger y = (1 \ 1 \ 0)$
- $R|A^c = 0$, then $R = (0 \ 1 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0)$

Then, the algorithm produces a new approximation by retaining only $M/3 = 2$ (the largest entries in R) entries in R :

- $D^1 = R_{M/3} = (0 \ 1 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0)$

4. *Update* Finally, the samples are updated so that they reflect the residual, the part of the signal that has not been approximated:

- $r^1 = y - \Phi D^1 = 0$

Now, AFB checks the termination rule to decide whether to break the iteration or to begin a new one:

- $(((\|r^1\|_2 = 0 < \gamma = 10^{-6}) = true) | ((L = L_{max}) = true)) = true$ then break the iteration.

The algorithm executes one iteration and returns $D^1 = (0 \ 1 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0)$ as output. It's clear that, the AFB algorithm correctly yields the optimal solution.

5 Simulation Results

In this section, using MATLAB R2016b, we evaluate the overall performance of the proposed reconstruction algorithm (AFB) and we compare the performance of the proposed algorithm with the performance of OMP, ROMP, SP and FBP algorithms (with Selection size= $M/2$ and Removable size= $M/3$). First, we test the proposed AFB algorithm by reconstructing the signals collected from 54 sensors located in Intel Berkeley Research Lab [17]. The experiments also cover reconstruction of computer-generated signals for diverse non-zero coefficient distributions, which include uniform and Gaussian distributions along with binary non-zero coefficients. The reconstruction performance has been evaluated using Gaussian and Bernoulli observation matrices.

Second, we evaluate the performance of the proposed reconstruction algorithm in the presence of noisy observations and compare the results with FBP, SP, OMP and ROMP algorithms. Reconstruction accuracy is expressed in terms of Average Normalized Mean Squared Error (ANMSE) for 500 test-samples. For every test-sample, we utilize specific observation matrix which is drawn from Gaussian distribution.

5.1 Experiments Over Real Dataset

In this section, we experiment the proposed algorithm by reconstructing the signals collected from the WSN centrally located at the Intel Berkeley Research lab. We use the Discrete Cosine Transform (DCT) as the sparsifying domain. It is noted that the environmental signals in the detected area possess sparse DCT coefficient vectors as demonstrated in Fig. 1. Figure 2a–d. It can be noted from the figures that our algorithm succeeds to achieve high performance in recovering temperature and humidity.

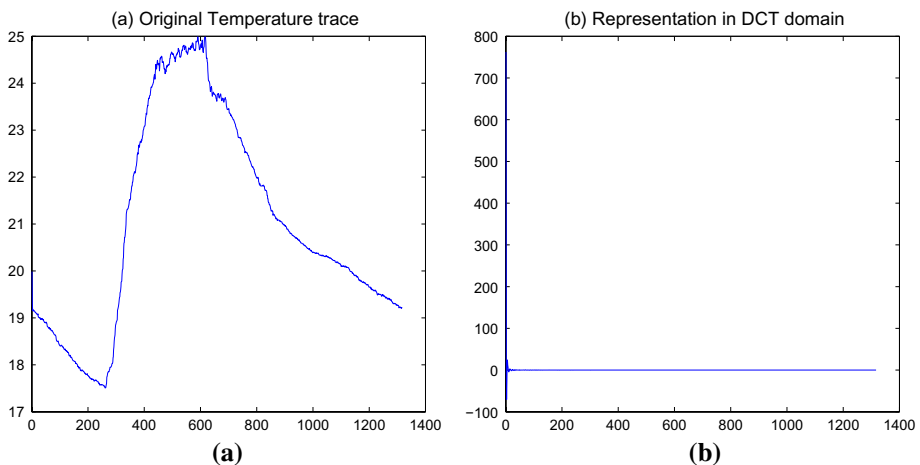


Fig. 1 **a** Intel temperature trace. **b** Intel temperature trace in DCT domain

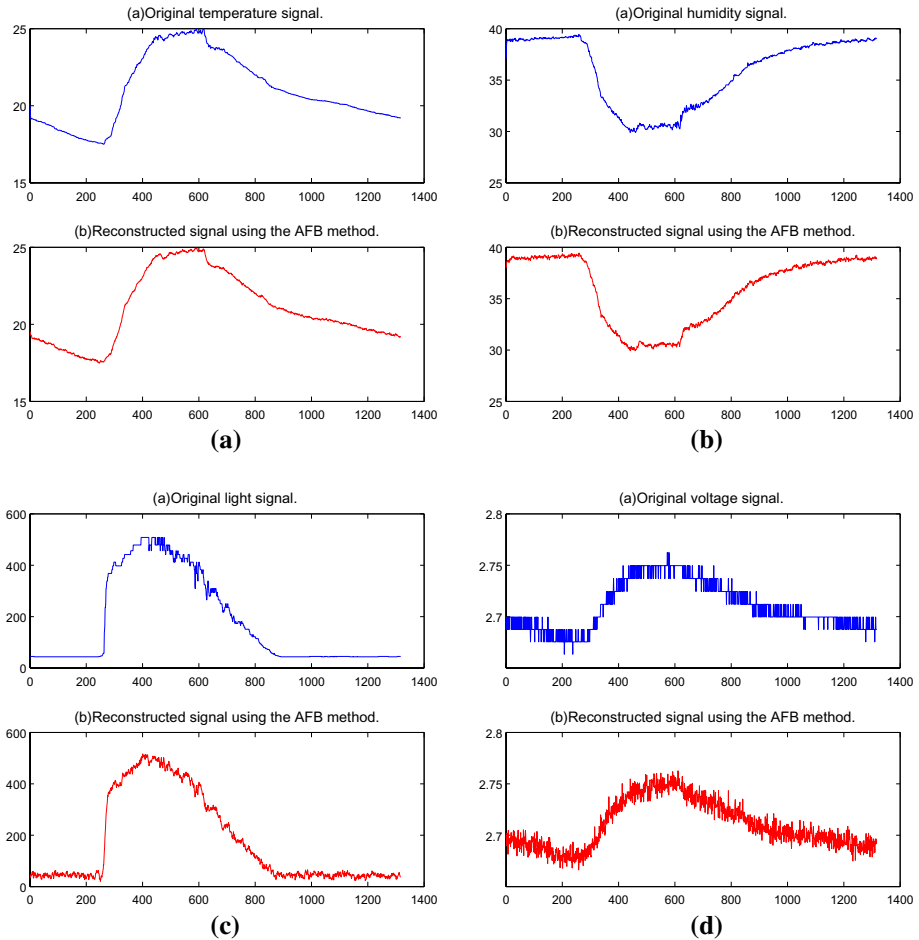


Fig. 2 Reconstruction results over **a** Intel temperature trace, **b** Intel humidity trace, **c** Intel light trace and **d** Intel voltage trace employing Gaussian observation matrices

5.2 Different Coefficient Distributions

Here, we executed three tests: the first test utilizes uniform sparse data with nonzero values collected from the Uniform distribution $U[-1, 1]$. Figure 3 shows that the proposed AFB algorithm provides lower ANMSE than FBP, SP, ROMP and OMP. The second test utilizes Gaussian sparse values, whose non-zero entries are taken from the standard Gaussian distribution. Figure 4 shows that the proposed algorithm has considerably better reconstruction performance than OMP, ROMP, SP and FBP. Finally, the third test utilizes sparse binary vectors and the non-zero coefficients are chosen to be 1. Figure 5, shows that the AFB algorithm outperforms OMP, ROMP, SP and FBP methods in this case too. In all the three cases, the performance of proposed AFB algorithm exceeds the performance of the existing algorithms because of the correct selection of columns in each round, where the selection depends on solving the least squares problem which gives the ability to find the correct columns in each round.

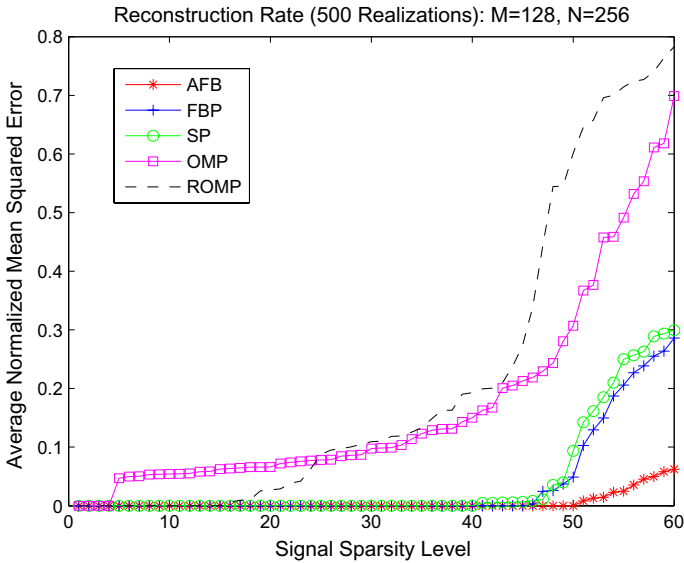


Fig. 3 Reconstruction results over sparsity for Uniform sparse signals employing Gaussian observation matrices

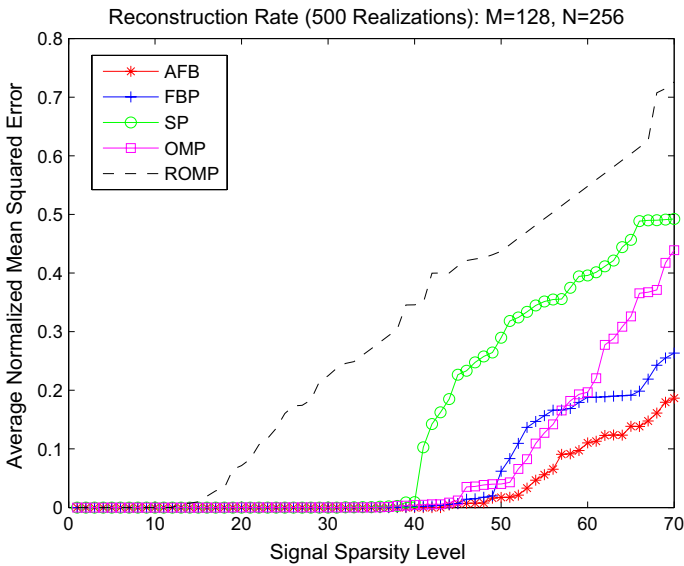


Fig. 4 Reconstruction results over sparsity for Gaussian sparse vectors using Gaussian observation matrices

5.3 Performance Evaluation Under Different Observation Lengths

We have conducted another important test to evaluate the reconstruction capability by changing the observation length M and fixing the sparse ratio S . Figure 6a depicts the

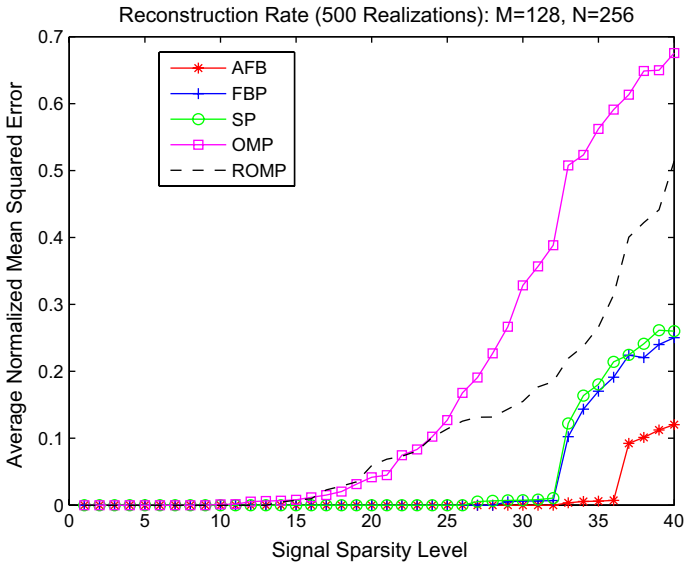


Fig. 5 Reconstruction results over sparsity for sparse binary signals using Gaussian observation matrices

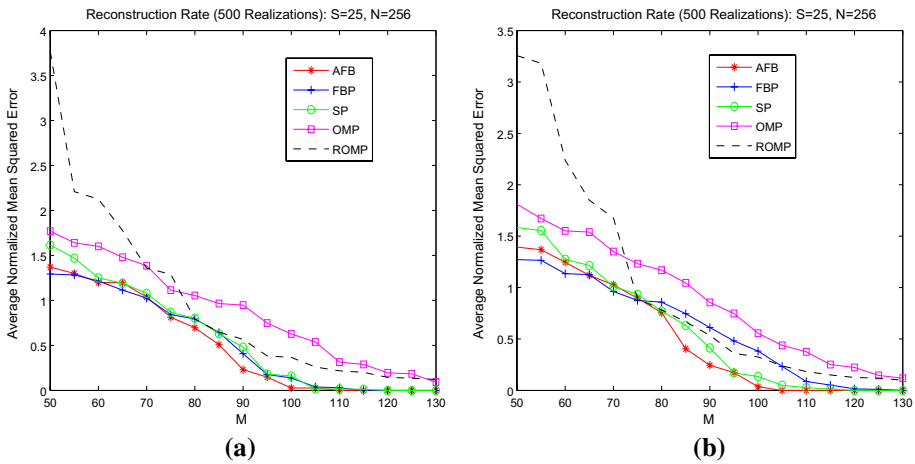


Fig. 6 **a** Reconstruction results over observation lengths for Binary sparse signals where $S = 20$ using a single Gaussian observation matrix for each M . **b** Reconstruction results over observation lengths for binary sparse signals where $S = 20$ using a Bernoulli distribution matrix for each M

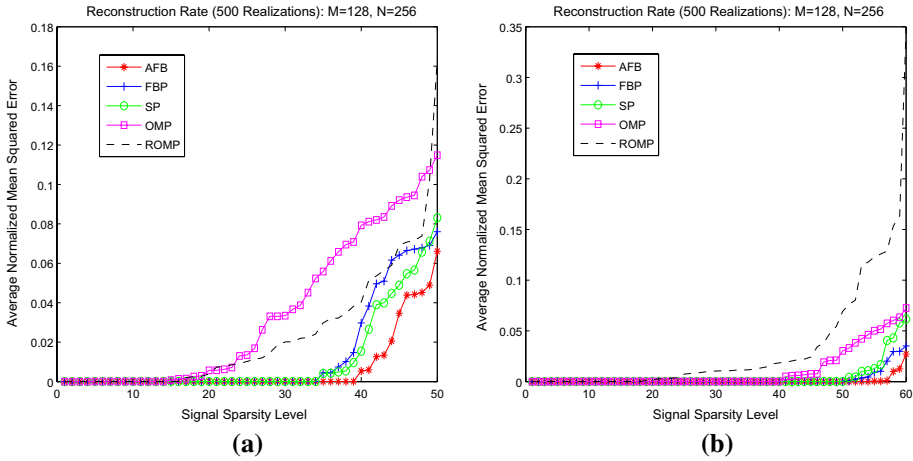


Fig. 7 **a** ANMSE for reconstruction of Uniform sparse signals from noisy observations using Gaussian observation matrices. **b** ANMSE for reconstruction of binary sparse signals from noisy observations using Gaussian observation matrices

reconstruction performance over M (M ranges from 50 to 130 and increments by 5 and $S = 25$). Φ is Gaussian Measurement matrix. It is clear that the ANMSE of the proposed algorithm is lower than that of OMP, ROMP, SP and FBP algorithms. In Fig. 6b, we replicate the last scenario with Φ drawn from the Bernoulli distribution. We can observe that the ANMSE for the proposed algorithm is still lower than, SP, FBP, OMP and ROMP algorithms.

5.4 Reconstruction performance for Noisy Observations

Here, we aim to evaluate the reconstruction performance of the proposed AFB algorithm when simulated using noisy observations Fig. 7a, b show the reconstruction error for noisy binary and uniform sparse signals. It can be noted that the proposed algorithm produces less error than OMP, ROMP, SP and FBP.

6 Conclusion

In this paper, we have proposed a greedy based CS reconstruction algorithm called Adaptive Iterative Forward–Backward Greedy Algorithm (AFB). AFB consists of four steps: Parameters initialization, Selection and Estimation, Check and Remove, and Update step. AFB initializes its parameters in the Parameters initialization step. Then, during the Selection and Estimation step, AFB proceeds to update the support set by solving the least squares problem and selects $M/2$ columns. In the Check and Remove step, AFB checks the estimated set and removes the incorrect columns. Finally in the Update step, AFB decides whether to terminate or to repeat the previous steps. AFB is evaluated in comparison with existing baseline algorithms in terms of Average Normalized Mean Squared Error. AFB exhibits superior performance and significantly reduces the reconstruction error, indicating that it is a promising approach for CS reconstruction.

References

1. Donoho, D. L. (2006). Compressed sensing. *IEEE Transactions on Information Theory*, 52(4), 1289–1306.
2. Cands, E. J. (2006). Compressive sampling. In *Proceedings of the International Congress of Mathematicians*, Vol. III, pp. 1433–1452, Madrid, Spain.
3. Meenu, R., Sanjay Dhok, B., & Deshmukh, A. R. B. (2018). Systematic review of compressive sensing: Concepts, implementations and applications. *IEEE Access*, 6, 4875–4894.
4. Lv, C., Wang, Q., Yan, W., & Shen, Y. (2016). Energy-balanced compressive data gathering in wireless sensor networks. *Journal of Network and Computer Applications*, 61, 102–114.
5. Khedr, A. M. (2015). Effective data acquisition protocol for multi-hop heterogeneous wireless sensor networks using compressive sensing. *Algorithms*, 8(4), 910–928.
6. Xinpeng, D., Cheng, L., & Chen, D. (2014). A simulated annealing algorithm for sparse recovery by l_0 minimization. *Neurocomputing*, 131, 98–104.
7. Davenport, M., Boufounos, P., Wakin, M., & Baraniuk, R. G. (2010). Signal processing with compressive measurements. *IEEE Journal of Selected Topics in Signal Processing*, 5(5), 445–460.
8. Venkataramani, R., & Bresler, Y. (1998). Sub-nyquist sampling of multiband signals: Perfect reconstruction and bounds on aliasing error. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 12–15.
9. Tropp, J. A., & Gilbert, A. C. (2007). Signal recovery from random measurements via orthogonal matching pursuit. *IEEE Transactions on Information Theory*, 53(12), 4655–4666.
10. Donoho, D. L., Tsaig, Y., Drori, I., & Starck, J. L. (2012). Sparse solution of underdetermined systems of linear equations by stagewise orthogonal matching pursuit. *IEEE Transactions on Information Theory*, 58(2), 1094–1121.
11. Needell, D., & Vershynin, R. (2009). Uniform uncertainty principle and signal recovery via regularized orthogonal matching pursuit. *Foundations of Computational Mathematics*, 9(3), 317–334.
12. Needell, D., & Tropp, J. A. (2009). CoSaMP: Iterative signal recovery from incomplete and inaccurate samples. *Applied and Computational Harmonic Analysis*, 26(3), 301–321.
13. Wei, D., & Olgica, M. (2009). Subspace pursuit for compressive sensing signal reconstruction. *IEEE Transactions on Information Theory*, 55(5), 2230–2249.
14. Cevher, V., & Jafarpour, S. (2010). Fast hard thresholding with nesterovs gradient method. In *Neuronal information processing systems, workshop on practical applications of sparse modeling*, Whistler, Canada
15. Burak, N., & Erdogan, H. (2013). Compressed sensing signal recovery via forward-backward pursuit. *Digital Signal Processing*, 23, 1539–1548.
16. Davenport, M. A., Boufounos, P. T., Wakin, M. B., & Baraniuk, R. G. (2010). Signal processing with compressive measurements. *IEEE Journal of Selected Topics in Signal Processing*, 4, 445–460.
17. Intel Berkeley Research Lab Data Set. <http://db.csail.mit.edu/labdata/labdata.html>. Accessed 9 October 2017.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Ahmed Aziz received his B.Sc. degree with honor in Computer science in June 2007, the M.S. degree in OCT 2014, in area of computer science from the Faculty of Computers and Informatics, Benha University, Benha, Egypt. He received his Ph.D. in computer science from school of computer and system science, Jawaharlal Nehru University, New Delhi, India in 2019. From December 2007 till 2014 he has been working as demonstrator and from 2014 till 2018 he has been working as Assistant Lecturer at the department of Computer Science, College of computers and informatics, Benha University, Benha, Egypt (www.fci.bu.edu.eg). From 2019 till now he has work as Assistant professor in the collage. His research interests include sensor networks, Compressive sensing, computing, wireless networks and IoT.



Walid Osamy received his B.Sc. degree with honor in computer science in June 2000, the M.S. degree in June 2006, and the Ph.D. degree in September 2010 all in the area of computer science and from the Faculty of Science, Zagazig University, Egypt. From July 2001 to March 2009 he has been involved in the projects of Network Infrastructure, and Management information systems at Communication Information Technology Center (CITC) and a member of Information and Communication Technology Projects (ICTP) at Zagazig University, Egypt. From December 2010 to January 2019 he has been working as an assistant professor at the Department of Computer Science, College of Computers and Informatics, Benha University, Egypt. From February 2019 till now he is working as an associate professor at the Department of Computer Science, College of Computers and Artificial Intelligence, Benha University, Egypt. From March 2015 until now, he has been working as an assistant professor at Qassim University, Buridah, KSA. His research interests include Computational Intelligence and in the field of IoT (Mobile Computing, Wireless Sensor Networks (WSNs)).



Ahmed M. Khedr received his B.Sc degree in Mathematics in June 1989 and the M.Sc degree in the area of optimal control in July 1995, both from Zagazig University, Egypt. In July 1999, he received his M.Sc and in March 2003, he received his Ph.D. degrees, both in Computer Science and Engineering, from University of Cincinnati, Ohio, USA. From March 2003 to January 2004, he was a research assistant professor at ECECS Department University of Cincinnati, USA. From January 2004 to May 2009, he worked as an assistant professor at Zagazig University, Egypt. From September 2009 to September 2010 he worked as an associate professor at the Department of Computer Science, College of Computers and Information Systems, Taif University, KSA. Since December 2014, he is a Professor at Zagazig University, Egypt. From September 2010 till December 2019, he worked as an Associate Professor and since January 2020, is a Professor at the Department of Computer Science, College of Computing and Informatics, University of Sharjah, UAE. He was awarded the State Prize of distinction in advanced technology, Sharjah Islamic Bank prize of distinction in research and the University of Sharjah prize of distinction in research, in June 2009, May 2013 and April 2014, respectively. His research interests include Wireless Sensor Networks, Internet of Things, and Distributing Computing.

tinction in research and the University of Sharjah prize of distinction in research, in June 2009, May 2013 and April 2014, respectively. His research interests include Wireless Sensor Networks, Internet of Things, and Distributing Computing.