

Massive MIMO Channel Estimation Algorithm Based on Weighted Compressed Sensing

Zhiguo Lv* and Weijing Wang*

Abstract

Compressed sensing-based matching pursuit algorithms can estimate the sparse channel of massive multiple input multiple-output systems with short pilot sequences. Although they have the advantages of low computational complexity and low pilot overhead, their accuracy remains insufficient. Simply multiplying the weight value and the estimated channel obtained in different iterations can only improve the accuracy of channel estimation under conditions of low signal-to-noise ratio (SNR), whereas it degrades accuracy under conditions of high SNR. To address this issue, an improved weighted matching pursuit algorithm is proposed, which obtains a suitable weight value u_{op} by training the channel data. The step of the weight value increasing with successive iterations is calculated according to the sparsity of the channel and u_{op} . Adjusting the weight value adaptively over the iterations can further improve the accuracy of estimation. The results of simulations conducted to evaluate the proposed algorithm show that it exhibits improved performance in terms of accuracy compared to previous methods under conditions of both high and low SNR.

Keywords

Channel Estimation, Compressed Sensing, Matching Pursuit, Sparse Reconstruction

1. Introduction

Massive multiple-input multiple-output (MIMO) systems are equipped with a large number of antennas at a base station. This method can effectively improve transmission and spectrum efficiency. As the number of antennas at the base station tends to infinity, the influence of thermal noise on the system can be ignored [1-3]. To obtain these benefits, the base station must perform a precoding process on the transmitted signal, which requires the base station to know the downlink channel state information (CSI) in advance. The structure of the transmitted symbols in pilot-based channel estimation is shown in Fig. 1. To ensure the accuracy of channel estimation, the proportion of the pilot signal in the data frame should increase with the number of antennas, resulting in a lower communication efficiency. Furthermore, it is very difficult for mobile users to estimate the downlink channel with massive antennas in real time. Therefore, research on channel estimation algorithms with low pilot overhead and high accuracy at mobile user terminals has attracted considerable attention in recent years.

Experimental results on channel measurement of actual communication systems have shown that the channels of massive MIMO systems exhibit sparse or nearly sparse characteristics in many cases [4,5].

※ This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

Manuscript received January 22, 2021; first revision March 30, 2021; accepted May 4, 2021.

Corresponding Author: Weijing Wang (aoligeiwwj@163.com)

* Dept. of Computer and Information Engineering, Luoyang Institute of Science and Technology, Luoyang, China (lzg96wl@163.com, aoligeiwwj@163.com)

Leveraging these sparse characteristics, compressed sensing-based channel estimation algorithms can estimate the CSI of a massive MIMO system with shorter pilot sequences [6-8]. This approach has the advantage of low computational complexity. The matching pursuit (MP) and orthogonal matching pursuit (OMP) algorithms are commonly used to perform sparse channel recovery. Although their computational complexity is low, the accuracy of their estimations cannot meet the requirements of massive MIMO systems. Therefore, some modified algorithms have been proposed to improve estimation accuracy by exploiting the joint sparse feature of the channel [9-11]; however, their application environment involves very strict restrictions.

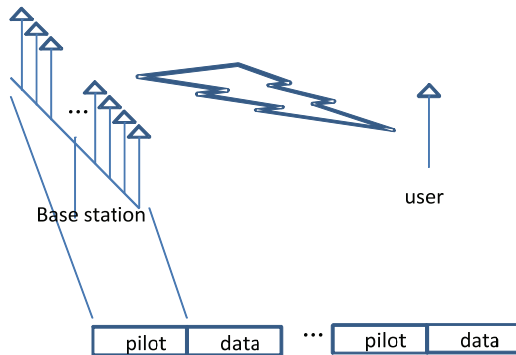


Fig. 1. Frame containing pilot and data symbols; the pilot symbols block is used to estimate channels.

MP is an iterative algorithm in which the estimated values obtained in different iterations provide different contributions to the final estimation result. However, the original MP algorithm does not distinguish the estimated values obtained in different iterations, resulting in a low estimation accuracy. To improve the accuracy of channel estimation, the iterations are divided into different stages [12]. The number of selected indices remains constant over the iterations of a stage and successively increases with a fixed step size from one stage to the next. Similarly, an algorithm called the variable metric method based on gradient pursuit (VMMGP) was proposed, which applies the gradient descent algorithm to channel estimation [13]. The estimated elements obtained in different iterations are multiplied by the weight value to build the final estimation result. The weight values are calculated based on the residual signal and the objective function of the gradient. In [14], an adaptive weighted OMP (AWOMP) algorithm was proposed. The reciprocals of the residual signal in each iteration were used as weight values to improve accuracy. After observing its performance in terms of accuracy over a larger range of signal-to-noise ratio (SNR) values, it was found that the weighted algorithm could improve the accuracy of channel estimation in the case of low SNR, but degraded it under conditions of high SNR. Therefore, an adaptive weighting and MP (AWMP) algorithm was proposed in [15] to weight the channel estimations obtained in each iteration under conditions of low SNR and use the original MP algorithm under conditions of high SNR.

The weighted algorithms discussed above have the following disadvantages. First, the weight values used in these weighted algorithms are not optimal. Second, the method for calculating weight values based on residual signals is imprecise. The residual signal is related not only to noise but also to the representation error of the measurement signal. In the case of low SNR, noise plays a dominant role. A small weight value mainly suppresses noise, thus improving accuracy. In the case of high SNR, noise is

no longer the dominant factor; hence, a small weight value mainly suppresses the useful measurement signal and degrades the estimation accuracy. Finally, accurately estimating the SNR is very difficult in actual communication systems; it is thus challenging to select different channel estimation algorithms according to the SNR level.

To solve these problems, in this study, we propose an improved weighted MP algorithm based on compressed sensing. The algorithm assigns weight values to estimated elements obtained in different iterations and then builds the final estimated channel. The weight value is obtained by training the channel data and can be changed with an increase in the number of iterations. The result of simulations conducted to evaluate the proposed approach show that the channel estimation accuracy is improved compared to that of existing methods under conditions of both high and low SNR.

2. System Model

2.1 Massive MIMO

The channel between a mobile user and a base station in a massive MIMO system can be represented by a high-dimensional vector. The elements in the vector are often assumed to be random variables with zero mean and unit variance in a Gaussian distribution. Considering a massive MIMO system with a base station equipped with N_t antennas, the channel between the base station and a mobile user with a single antenna can be represented as an N_t -dimensional vector. If multiple users with a single antenna or a user with multiple antennas are considered, the channel is represented by a channel matrix. The channel matrix can be vectorized to form a channel vector. For simplicity, we use an N -dimensional vector \mathbf{h} to represent the channel, which may be an actual channel vector or a channel vector obtained by vectorization of a channel matrix. In the downlink, the pilot signals are transmitted from the base station to the mobile users. The pilot response signal \mathbf{y} received by the mobile user can be expressed as

$$\mathbf{y} = \Phi \mathbf{h} + \mathbf{n} \quad (1)$$

where $\Phi = [\Phi(1), \Phi(2), \dots, \Phi(L_p)]^T$ is an $L_p \times N$ dimensional pilot matrix, $\Phi(i)$ represents the N -dimensional pilot vectors transmitted by all N transmit antennas at time i , and L_p is the length of the pilot sequence. \mathbf{n} is an $L_p \times 1$ dimensional additive white Gaussian noise vector, and each element in vector \mathbf{n} obeys a Gaussian distribution with zero mean and σ^2 variance.

If the actual channel does not exhibit sparse features, it can be converted to a sparse form using a sparse dictionary. For convenience, only channels with a sparse form are considered herein. In traditional channel models, it is often assumed that the elements of the channel are subject to a Gaussian distribution; however, this assumption may not hold for the actual channels in practice. In the following subsection, a universal channel model is introduced, which can be applied to sparse channels with any probability distribution, including an unknown distribution.

2.2 Sparse Channel Model

If most elements of a signal vector are zero or close to zero, we call the signal vector sparse. The number of non-zero elements in the sparse signal vector is called the sparsity degree, which is represented by K . In addition to the sparsity degree, it is necessary to introduce the probability distribution of the

elements in the sparse signal vector. To adopt an identical model to describe sparse signals with different probability distributions, a universal model is introduced in Eq. (2) below.

$$\mathbf{h} = \mathbf{h}_a \odot \mathbf{h}_b \quad (2)$$

where \mathbf{h} is an N -length sparse signal column vector, \odot represents the Hadamard product, \mathbf{h}_a is a column vector of length N , and each element in \mathbf{h}_a is a random variable obeying a certain distribution. \mathbf{h}_b is also a random variable column vector of length N , and each element in \mathbf{h}_b obeys the Bernoulli distribution with the success probability parameter p . The value of parameter p is small, that is, the probability of elements in vector \mathbf{h} being non-zero is low, which ensures the sparsity of channel \mathbf{h} . The elements in \mathbf{h}_a , regardless of the probability distribution, do not affect the sparse characteristics of the channel.

According to the compressed sensing theory, the high-dimensional sparse vector \mathbf{h} can be reconstructed using the measurement vector \mathbf{y} , even in a noisy environment. Reconstruction requires the design of a measurement matrix and algorithm to estimate the non-zero positions of vector \mathbf{h} and the values at those positions. The non-zero positions are determined by vector \mathbf{h}_b , and the values at those positions are determined by vector \mathbf{h}_a . In the proposed improved weighted MP algorithm, the non-zero positions of \mathbf{h} are first determined, and then the values at these non-zero positions are estimated. This is equivalent to estimating \mathbf{h}_b first and then estimating \mathbf{h}_a . For simplicity, we no longer specifically emphasize the channel estimation of \mathbf{h}_a or \mathbf{h}_b in the subsequent sections.

3. Weighted MP Algorithm

Compressed sensing techniques can estimate a sparse channel with short pilot sequences. An N -dimensional sparse signal \mathbf{h} can be reconstructed from M -dimensional random sample values $\mathbf{y} = \Phi \mathbf{h}$, where Φ is an $M \times N$ -dimensional measurement matrix ($M < N$). To ensure the accuracy of the reconstruction, the measurement matrix is required to satisfy the condition of the restricted isometry property (RIP). When the condition $M \geq K \log(\frac{N}{K})$ is satisfied, K is the sparsity degree of the sparse signal, and the original sparse signal can be reconstructed using equation (3) below even in a noisy environment [16].

$$\hat{\mathbf{h}} = \underset{\mathbf{h}}{\operatorname{argmin}} \|\mathbf{h}\|_{\ell_0} \quad \text{s.t.} \quad \|\mathbf{y} - \Phi \mathbf{h}\|_{\ell_2}^2 \leq \varepsilon, \quad (3)$$

where $\|\cdot\|_{\ell_0}$ denotes ℓ_0 -norm, and $\|\cdot\|_{\ell_2}$ denotes ℓ_2 -norm.

The MP algorithm can be used to obtain a solution to Eq. (3). Only one element of the sparse signal is estimated in each iteration. All the estimated elements obtained in the current and previous iterations are stacked together according to their respective positions to construct the final estimated signal.

3.1 Weight Value Analysis

If the estimated element obtained in the initial iteration contains an error, this discrepancy causes the estimated elements obtained in the subsequent iterations to contain a larger error; this phenomenon is referred to as error propagation. The estimated element $\hat{\mathbf{h}}_i$ obtained in the i -th iteration consists of two parts: the true value \mathbf{h}_i and the estimation error \mathbf{e}_i . According to the proportional relationship between

the true value and the estimation error, a weight value is assigned to the estimated element. The weight value is calculated as follows.

$$u_i = \frac{|\mathbf{h}_i|^2}{|\mathbf{h}_i|^2 + |\mathbf{e}_i|^2} = \frac{1}{1 + \frac{|\mathbf{e}_i|^2}{|\mathbf{h}_i|^2}} \quad (4)$$

Because the true value and error cannot be accurately known in practice, it is impossible to calculate the weight value u_i . However, the range of the weight value can be determined. When the SNR is high, the error caused by noise is relatively small; thus, the weight value should be larger. The maximum value of the weight, $u_i = 1$, is obtained when the noise approaches 0. When the SNR is low, the error caused by noise is large, and the weight should therefore be smaller. The minimum value of the weight, $u_i = 0$, is obtained when the noise tends to infinity. Therefore, the actual weight value falls within the range 0–1. In addition, the number of iterations affects the weight value. The non-zero elements in the sparse vector are estimated in descending order according to the modulus value of the original MP algorithm. The estimated value of each element is calculated as the inner product of the residual signal and the atom (the column of the measurement matrix Φ). As the number of iterations increases, the residual signal sequentially removes the contribution of the estimated elements. Therefore, the modulus of the true value and the error both decrease gradually with the number of iterations.

The estimation error \mathbf{e}_i consists of two parts and can be expressed as

$$\mathbf{e}_i = \mathbf{a}_i + \mathbf{b}_i \quad (5)$$

where \mathbf{a}_i is the error caused by using atoms to approximate the i -th residual signal, \mathbf{b}_i is the error caused by the noise in the i -th iteration. The elements in the measurement matrix are independent and identically distributed Gaussian variables. Hence, the projections of noise on each atom are approximately equal, that is, \mathbf{b}_i can be considered to be independent of iteration i . In comparison, \mathbf{a}_i decreases as the number of iterations increases. By observing the variation curves of the actual signals \mathbf{a}_i and \mathbf{h}_i , it is found that \mathbf{a}_i exhibits a faster attenuation speed than \mathbf{h}_i . Therefore, the weight value should increase with successive iterations.

The growth curve of the weight value over successive iterations is affected by the probability distribution and attenuation speed of the channel element, as well as by noise contamination. As it is difficult to obtain an accurate expression of the growth of the weight value, it is assumed that the weight value increases linearly for convenience of analysis. Therefore, the initial value and step of increase are sufficient to obtain the required weight values for each iteration. These parameters can be easily obtained from an actual channel environment. After training the channel data, a suitable weight value u_{op} can be obtained, which implies that the loss of estimation accuracy after weighting in the high-SNR region is reduced, whereas the gain of estimation accuracy in the low-SNR region is increased. Given a suitable weight u_{op} , the weight increase at each step is calculated as $\Delta u = 2(1 - u_{op})/K$, where K is the number of iterations. The initial weight value is calculated as $u_0 = 2u_{op} - 1$.

3.2 Algorithm Description

To improve the accuracy of the estimation, an improved weighted MP algorithm is proposed, in which

the weight value of u_i is set to the newly added non-zero element. A flowchart of the algorithm is shown in Fig. 2.

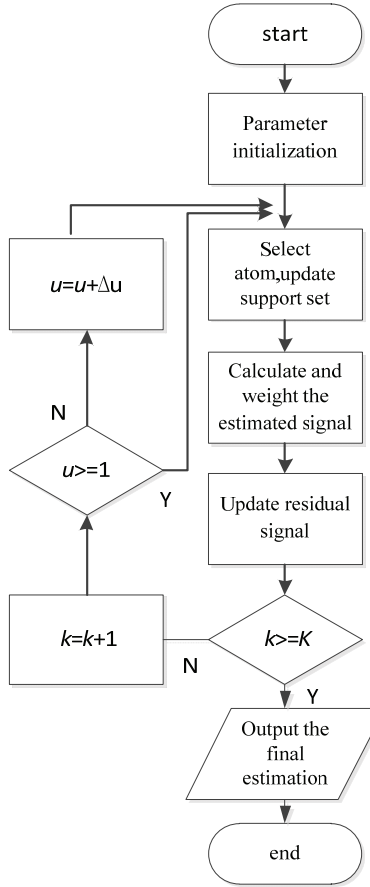


Fig. 2. Flowchart of the proposed algorithm.

The main steps of the algorithm are described as follows.

(1) Parameter initialization

The parameter initialization operation is performed first. The residual signal $\mathbf{r}_0 = \mathbf{y}$, the number of iterations $k = 1$, the support set (a set of non-zero positions of the sparse signal) is initialized as an empty set, i.e., $\Lambda_0 = \emptyset$, the sparsity degree is set as K , the set of indices of the atoms $\Omega = \{1, 2, \dots, N\}$, and the estimated signal $\hat{\mathbf{h}}$ is initialized as the zero vector $\hat{\mathbf{h}}_0 = \mathbf{0}$.

(2) Select atom

By calculating the absolute value of the inner product between the residual signal and the atom, the corresponding index of the atom, denoted as $\lambda(k)$, which has the maximum correlation with the current residual signal \mathbf{r}_{k-1} , is selected.

$$\lambda(k) = \underset{i \in \Omega}{\operatorname{argmax}} |\langle \mathbf{r}_{k-1}, \Phi_i \rangle| \quad (6)$$

Here, $\langle \mathbf{r}_{k-1}, \Phi_i \rangle$ represents the inner product of the two vectors.

(3) Update support set

Combine the selected atom index $\lambda(k)$ with the support set Λ_{k-1} obtained in previous iterations to form a new support set Λ_k , i.e., $\Lambda_k = \Lambda_{k-1} \cup \{\lambda(k)\}$.

(4) Calculate the value of $\hat{\mathbf{h}}_k$ at position $\lambda(k)$

Calculate the estimated value of $\hat{\mathbf{h}}_k$ at position $\lambda(k)$, $\hat{\mathbf{h}}_k(\lambda(k)) = \langle \mathbf{r}_{k-1}, \Phi_{\lambda(k)} \rangle$.

(5) Update the estimated signal $\hat{\mathbf{h}}_k$ after weighting the estimated value of $\hat{\mathbf{h}}_k$ at the position of $\lambda(k)$

Multiply $\langle \mathbf{r}_{k-1}, \Phi_{\lambda(k)} \rangle$ by the weight value u_k and substitute it into the corresponding position of $\hat{\mathbf{h}}_k$ to complete the update of the estimated signal $\hat{\mathbf{h}}_k$.

$$\hat{\mathbf{h}}_k = \hat{\mathbf{h}}_{k-1} + u_k \hat{\mathbf{h}}_k(\lambda(k)) \quad (7)$$

(6) Update the weight value

Update the weight value u_k based on adjustment value Δu .

$$u_k = u_{k-1} + \Delta u \quad (8)$$

The weight adjustment value Δu in the algorithm may take a positive or a negative value. A positive value corresponds to an increase in weight value, while a negative value corresponds to a decrease in weight value. If the weight value is increased, the maximum value cannot be greater than 1, and if it is decreased, the minimum value cannot be less than 0.

(7) Update the residual signal \mathbf{r}_k based on $\hat{\mathbf{h}}_k$.

$$\mathbf{r}_k = \mathbf{r}_{k-1} - \Phi \hat{\mathbf{h}}_k \quad (9)$$

If $k \geq K$, the iterative process is complete; otherwise, $k = k + 1$, and repeat steps (2)–(7) to perform the next iteration.

(8) Output the final estimation $\hat{\mathbf{h}}_K$.

4. Weighted OMP Algorithm

The concept of weighting is introduced into the OMP algorithm to verify whether the weighting method is valid for other MP algorithms. The weighted OMP algorithm with a fixed weight value is called the fixed-weighted OMP (FWOMP). The weighted OMP algorithm using a varying weight value is referred to as varying-weighted OMP (VWOMP). The OMP algorithm differs from the MP algorithm in terms of the signal estimated in the current iteration. In the OMP algorithm, the sparse signal estimated in the current iteration already contains non-zero elements obtained in all previous iterations. The final estimated signal is not a superposition of the estimations obtained in different iterations. Therefore, the signal estimated in the current iteration is multiplied by the weight value first, and then removed from the previous residual signal to obtain the residual signal of the current iteration. The key steps of the weighted OMP algorithm are as follows.

(1) Parameter initialization

First, a parameter initialization operation is performed, in which the residual signal $\mathbf{r}_0 = \mathbf{y}$, the number of iterations $k = 1$, the support set is initialized as an empty set, i.e., $\Lambda_0 = \emptyset$, the sparsity degree is set as K , the set of indices of the atoms $\Omega = \{1, 2, \dots, N\}$, and the estimated signal $\hat{\mathbf{h}}$ is initialized as the zero vector $\hat{\mathbf{h}}_0 = \mathbf{0}$.

(2) Select atoms

By calculating the absolute value of the inner product between the residual signal and the atom, the corresponding index of the atom, denoted as $\lambda(k)$, which has the maximum correlation with the current residual signal \mathbf{r}_{k-1} , is selected.

$$\lambda(k) = \underset{i \in \Omega}{\operatorname{argmax}} |\langle \mathbf{r}_{k-1}, \Phi_i \rangle| \quad (10)$$

(3) Update support set

Combine the selected atom index $\lambda(k)$ with the support set Λ_{k-1} obtained in the previous iteration to form a new support set Λ_k , that is, $\Lambda_k = \Lambda_{k-1} \cup \{\lambda(k)\}$.

(4) Calculate the value of $\hat{\mathbf{h}}_k$ at the position of Λ_k

$$\hat{\mathbf{h}}_k = (\Phi_{\Lambda_k}^T \Phi_{\Lambda_k})^{-1} \Phi_{\Lambda_k}^T \mathbf{y} \quad (11)$$

The estimated value of $\hat{\mathbf{h}}_k$ at the position of Λ_k is calculated using Eq. (10).

(5) Update the estimated signal after the weighting.

Multiply $\hat{\mathbf{h}}_k$ by the weight value u_k to update $\hat{\mathbf{h}}_k$.

$$\hat{\mathbf{h}}_k = u_k \hat{\mathbf{h}}_k \quad (12)$$

(6) Update the weight value

Update the weight value u_k based on adjustment value Δu .

$$u_k = u_{k-1} + \Delta u \quad (13)$$

The weight adjustment value Δu in the algorithm may take a positive or a negative value. A positive value corresponds to an increase in weight value, while a negative value corresponds to a decrease in weight value. If the weight value is increased, the maximum value cannot be greater than 1, and if it is decreased, the minimum value cannot be less than 0.

(7) Update the residual signal \mathbf{r}_k based on the estimated signal $\hat{\mathbf{h}}_k$.

$$\mathbf{r}_k = \mathbf{r}_{k-1} - \Phi \hat{\mathbf{h}}_k \quad (14)$$

If $k \geq K$, the iterative process is complete; otherwise, $k = k + 1$, and repeat steps (2)–(7) to perform the next iteration.

(8) Output the final estimation $\hat{\mathbf{h}}_K$.

5. Simulation Results

In this section, we present the results of simulations performed to evaluate the proposed approach. The accuracy of channel estimation was evaluated in terms of the normalized mean square error (NMSE), which was calculated using the following equation:

$$\mu_{NMSE} = 10 \log_{10} \left(\frac{\|\hat{\mathbf{h}} - \mathbf{h}\|^2}{\|\mathbf{h}\|^2} \right) \quad (15)$$

where $\hat{\mathbf{h}}$ and \mathbf{h} are the estimated and actual channels, respectively.

The proposed algorithm with a fixed weight value is called the fixed-weighted MP (FWMP) algorithm. A suitable value of u_{op} was obtained by training the channel data. The influence of different weight values on the channel estimation accuracy was determined first. The simulation parameters were set as follows. The length of vector \mathbf{h}_a was 256. Each element of \mathbf{h}_a was a normally distributed random variable with zero mean and unit variance. The length of vector \mathbf{h}_b was also 256. Each element obeyed the Bernoulli distribution with a success probability of $p = 0.05$.

The estimation accuracy achieved by the FWMP algorithm using different weight values is shown in Fig. 3. The FWMP algorithm with a weight value $u = 1$ is the original MP algorithm. It can be observed that the NMSE first decreased and then increased as the weight value increased from 0.4 to 1.2. Compared with the original MP algorithm, it can be further observed that the NMSE performance of the FWMP algorithm in the low-SNR regime can be improved with $u_{op} = 0.8$. In contrast, the NMSE performance was degraded in the high-SNR regime, because the channel noise has less influence on the measured signal under such conditions. A non-one weight value distorts the estimated signal and degrades the estimation accuracy. After training with different weight values, the suitable value $u_{op} = 0.8$ was obtained in the specific channel with the model parameters given above.

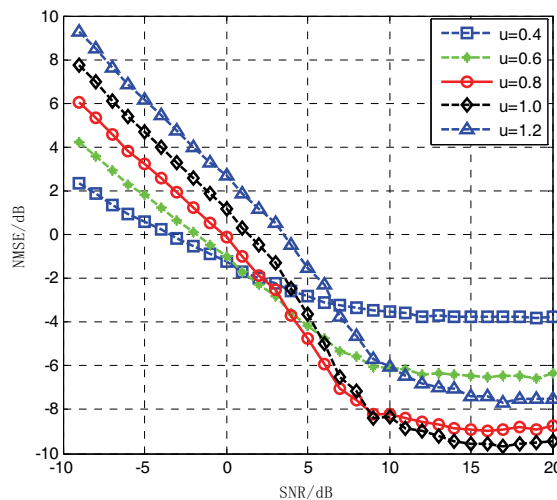


Fig. 3. Comparison of NMSE performance of FWMP algorithm with different weight values.

Because the estimation error varies between iterations, assigning varying weight values for the estimated elements obtained in each iteration is more reasonable. This reconstruction algorithm is called

varying weighted MP (VWMP). The VWMP algorithm is similar to the FWMP algorithm, except that the weight value changes between iterations. A comparison of the performance of the FWMP, VWMP, and original MP algorithms in the same simulation environment is shown in Fig. 4. The fixed-weight value in the FWMP algorithm was set to 0.8. In the VWMP1 algorithm, the initial value of the weight was set as $2 \cdot 0.8 - 1 = 0.6$. The step of the weight increase was calculated as $\Delta u = 2(1 - 0.8)/13 = 0.03$. To ensure that the weight value of the final iteration could be increased to 1, Δu was increased to 0.05. In the VWMP2 algorithm, the initial value of the weight was set to 1. The weight value decreased by 0.05 for each iteration. There was an approximately equal mean weight value between the VWMP1 and VWMP2 algorithms for fairness. It can be observed from Fig. 4 that the VWMP1 algorithm exhibited superior performance compared to the MP algorithm in terms of the NMSE. The VWMP1 algorithm was able to achieve a gain of approximately 1 dB in both the low- and high-SNR regimes compared with the original MP. Compared with the FWMP algorithm ($u = 0.8$), the VWMP1 algorithm could achieve a gain of approximately 5 dB under conditions of high SNR and similar NMSE performance in the low-SNR regime. The VWMP2 algorithm showed the worst performance in most cases, except for a weak performance improvement in the low-SNR regime. This shows that the proposed method of initializing the weight value and increasing it iteratively can improve the accuracy of the estimation under both low- and high-SNR conditions.

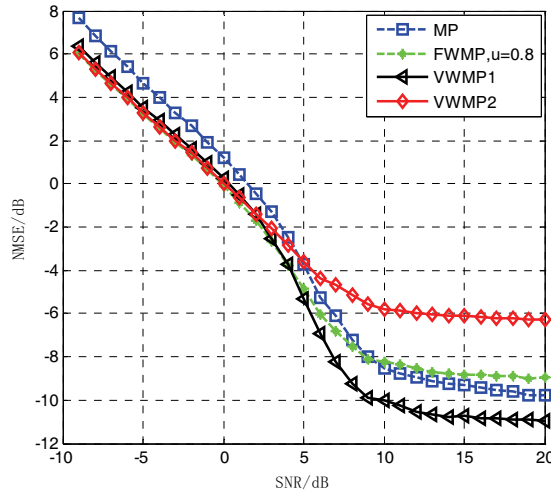


Fig. 4. Comparison of NMSE performance of different MP algorithms.

A suitable weight value was also obtained using the WOMP algorithm by learning the sparse channel data. Parameters of the sparse channel model were the same as those in the WMP algorithm, except for the success probability $p = 0.07$. The estimation accuracy of the FWOMP algorithm with different weight values is shown in Fig. 5. It can be observed that the accuracy of the FWOMP algorithm first improved and then decreased as the weight value increased from 0.7 to 1.0. Then, a suitable weight value of $u_{op} = 0.9$ was obtained. It may also be noted from the simulation results that the FWOMP algorithm performed better than the original OMP at a low SNR, whereas it performed worse with a high SNR, because the weight value does not change with iterations. The accuracy of the FWOMP algorithm with the weight value $u = 1.1$ was inferior to that of the OMP algorithm in both the high- and low-SNR regimes. This was due to the fact that the weight value exceeded the maximum value of 1.

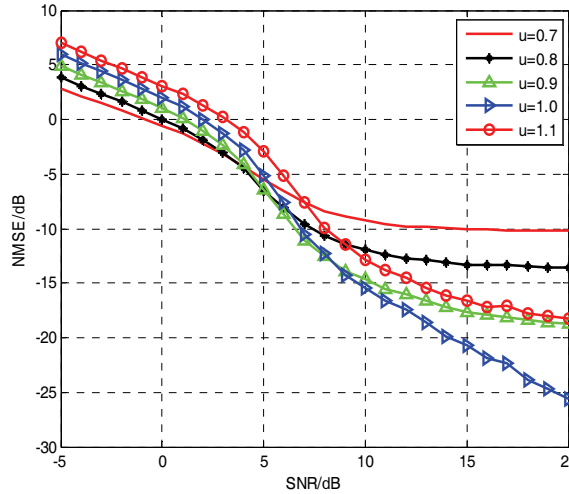


Fig. 5. Comparison of NMSE performance of FWOMP algorithm with different weights.

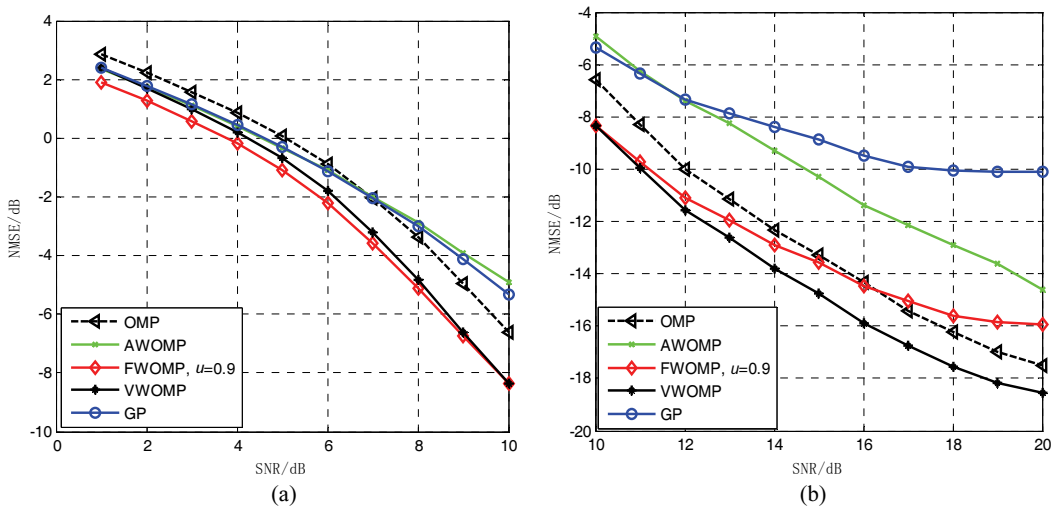


Fig. 6. Comparison of NMSE performance of different OMP algorithms: (a) low SNR and (b) high SNR.

A comparison of the performance of the FWOMP, VWOMP, gradient pursuit (GP) [13], AWOMP [14], and the original OMP algorithm is shown in Fig. 6. The FWOMP algorithm used a weight value of 0.9. In the VWOMP algorithm, the weight value was initialized as 0.8, and the weight increase step Δu was set to 0.01. It can be observed from the simulation results that the GP and AWOMP algorithms showed slightly improved performance compared with the original OMP algorithm under low-SNR conditions, but exhibited reduced estimation accuracy under high-SNR conditions. Compared with the GP, AWOMP, and FWOMP algorithms with a weight value of 0.9, the VWOMP algorithm showed significantly improved accuracy at high SNR and almost the same performance at low SNR. This shows that the weighting method can be successfully applied to the OMP algorithm.

To further illustrate the performance of the proposed algorithm, a comparison of the bit error rate (BER) of different algorithms is shown in Fig. 7. The parameter of success probability $p = 0.09$. Quadrature

phase-shift keying modulation and a half-rate convolutional code with generator sequences $G = [1011011, 1111001]$ were utilized. As shown in Fig. 7, the performance of the proposed weighted algorithms exceeded that of other algorithms in terms of BER. This shows that the proposed algorithm was able to obtain a more accurate estimated channel and thus a better BER.

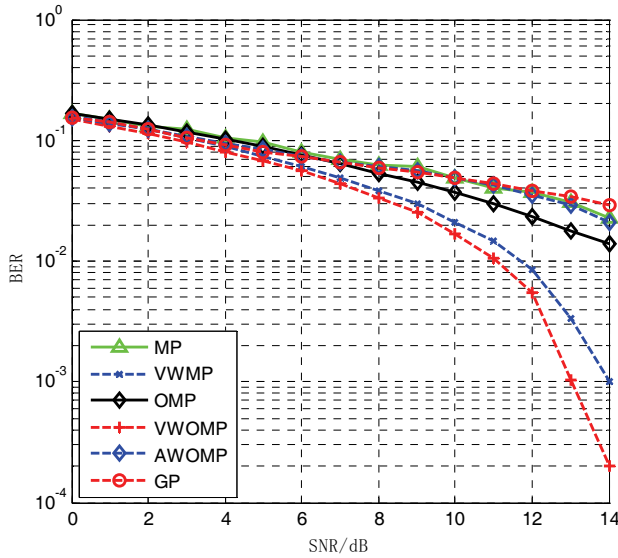


Fig. 7. Comparison of BER performance of different algorithms.

6. Conclusion

To solve the problem of poor channel estimation accuracy in massive MIMO systems, we proposed an improved weighted channel estimation algorithm, in which the estimated signals obtained in each iteration are weighted to improve the estimation accuracy. The estimated signal is weighted to find the correct atom with greater probability and achieve a balanced performance between the low- and high-SNR regimes. Because the residual signal becomes increasingly smaller with successive iterations, increasing the weight values iteratively can further improve the estimation accuracy under both low- and high-SNR conditions. The validity of the improved weighted MP algorithm was verified by the simulation results.

Acknowledgement

This work was supported in part by the science and technology breakthrough project of the Henan Science and Technology Department (No. 192102210249, 192102210116, and 212102210470), Key projects of colleges and universities in Henan Province (No. 19B510007 and 19A520006). It was also supported in part by the Science and Technology Development Plan of Henan Province (No. 202102310625).

References

- [1] E. G. Larsson, O. Edfors, F. Tufvesson, and T. L. Marzetta, "Massive MIMO for next generation wireless systems" *IEEE Communications Magazine*, vol. 52, no. 2, pp. 186-195, 2014.
- [2] M. Pappa, C. Ramesh, and M. N. Kumar, "Performance comparison of massive MIMO and conventional MIMO using channel parameters," in *Proceedings of 2017 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)*, Chennai, India, 2017, pp. 1808-1812.
- [3] J. Lee, K. J. Choi, and K. S. Kim, "Massive MIMO full-duplex for high-efficiency next generation WLAN systems," in *Proceedings of 2016 International Conference on Information and Communication Technology Convergence (ICTC)*, Jeju, Korea, 2016, pp. 1152-1154.
- [4] W. U. Bajwa, A. Sayeed, and R. Nowak, "Sparse multipath channels: Modeling and estimation," in *Proceedings of 2009 IEEE 13th Digital Signal Processing Workshop and 5th IEEE Signal Processing Education Workshop*, Marco Island, FL, 2009, pp. 320-325.
- [5] G. Gui, N. Liu, L. Xu, and F. Adachi, "Low-complexity large-scale multiple-input multiple-output channel estimation using affine combination of sparse least mean square filters," *IET Communications*, vol. 9, no. 17, pp. 2168-2175, 2015.
- [6] S. Hou, Y. Wang, T. Zeng, and S. Wu, "Sparse channel estimation for spatial non-stationary massive MIMO channels," *IEEE Communications Letters*, vol. 24, no. 3, pp. 681-684, 2020.
- [7] X. Lv, Y. Li, Y. Wu, and H. Liang, "Kalman filter based recursive estimation of slowly fading sparse channel in impulsive noise environment for OFDM systems," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 3, pp. 2828-2835, 2020.
- [8] W. Zhang, T. Kim, and S. H. Leung, "A sequential subspace method for millimeter wave MIMO channel estimation," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 5, pp. 5355-5368, 2020.
- [9] Z. Gao, L. Dai, W. Dai, B. Shim, and Z. Wang, "Structured compressive sensing-based spatio-temporal joint channel estimation for FDD massive MIMO," *IEEE Transactions on Communications*, vol. 64, no. 2, pp. 601-617, 2016.
- [10] W. Ding, F. Yang, W. Dai, and J. Song, "Time-frequency joint sparse channel estimation for MIMO-OFDM systems," *IEEE Communications Letters*, vol. 19, no. 1, pp. 58-61, 2015.
- [11] J. Si, X. Hou, and Y. Cheng, "Joint multi-signal reconstruction based on block pruning multipath matching pursuit," *Systems Engineering and Electronics*, vol. 38, no. 9, pp. 1993-1999, 2016.
- [12] Y. Zhang, R. Venkatesan, O. A. Dobre, and C. Li, "Efficient estimation and prediction for sparse time-varying underwater acoustic channels," *IEEE Journal of Oceanic Engineering*, vol. 45, no. 3, pp. 1112-1125, 2020.
- [13] P. P. Liu, L. I. Lei, and H. Y. Wang, "Research on greedy reconstruction algorithms of compressed sensing based on variable metric method," *Journal on Communications*, vol. 35, no. 12, pp. 98-105, 2014.
- [14] X. Zhu, Y. M. Li, and X. Q. Liu, "Impulse radio ultra-wideband signal detection based on compressive sensing," *Journal of Microwaves*, vol. 30, no. 5, pp. 76-81, 2014.
- [15] X. Yu, H. Zheng, and Y. Zeng, "Adaptive weighting matching pursuit algorithm based on compressed sensing," *Journal of Chongqing University of Posts and Telecommunication (Natural Science Edition)*, vol. 28, no. 5, pp. 707-712, 2016.
- [16] M. F. Duarte and Y. C. Eldar, "Structured compressed sensing: from theory to applications," *IEEE Transactions on Signal Processing*, vol. 59, no. 9, pp. 4053-4085, 2011.



Zhiguo Lv <https://orcid.org/0000-0002-2747-3487>

He received the B.S. degree in applied electronic technology from Henan Normal University in 2000. He received the M.S. degree in communication and information systems from Guilin University of Electronic Technology in 2008. He received the Ph.D. degree in communication and information systems from Xidian University in 2019. He is currently a lecturer at Luoyang Institute of Science and Technology. His research interests include compressive sensing, channel estimation, and MIMO techniques.



Weijing Wang <https://orcid.org/0000-0001-5710-9873>

She received the B.S. degree in mathematics and applied mathematics from Henan University of Science and Technology in 2003. She received the M.S. degree in computer science from Henan University of Science and Technology in 2006. She is currently a lecturer at Luoyang Institute of Science and Technology. Her research interests focus on image processing and artificial intelligence.