Journal of Systems Engineering and Electronics
Vol. 29, No. 4, August 2018, pp.731 – 741

Tracking multiple targets in MIMO radar via adaptive asymmetric joint diagonalization with deflation

ZHANG Zhengyan* and ZHANG Jianyun

College of Electronic Countermeasure, National University of Defense Technology, Hefei 230037, China

Abstract: In view of the low performance of adaptive asymmetric joint diagonalization (AAJD), especially its failure in tracking high maneuvering targets, an adaptive asymmetric joint diagonalization with deflation (AAJDd) algorithm is proposed. The AAJDd algorithm improves performance by estimating the direction of departure (DOD) and direction of arrival (DOA) directly, avoiding the reuse of the previous moment information in the AAJD algorithm. On this basis, the idea of sequential estimation of the principal component is introduced to turn the matrix operation into a constant operation, reducing the amount of computation and speeding up the convergence. Meanwhile, the eigenvalue is obtained, which can be used to estimate the number of targets. Then, the estimation of signal parameters via rotational invariance technique (ESPRIT) algorithm is improved to realize the automatic matching and association of DOD and DOA. The simulation results show that the AAJDd algorithm has higher tracking performance than the AAJD algorithm, especially when the high maneuvering target is tracked. The efficiency of the proposed method is verified.

Keywords: MIMO radar, low complexity, angles tracking, adaptive asymmetric joint diagonalization with deflation (AAJDd), high maneuvering target.

DOI: 10.21629/JSEE.2018.04.08

1. Introduction

Multiple-input multiple-output (MIMO) radar brings many advantages compared with traditional phased array radar and multi-static radar, and has become a hot research topic in recent years [1 – 12]. MIMO radar can transmit orthogonal waveforms with multiple antennas, and simultaneously receive echoes from the targets by multiple antennas. Since the transmitted waveforms are independent between each other, the MIMO radar has a higher degree of freedom and can identify, locate and track more targets under the same configuration [13 – 16]. MIMO radar can be either equipped with widely separated antennas (statistics MIMO radar) [17] or collocated antennas (coherent MIMO radar) [18]. Bistatic MIMO radar is a common form of coherent MIMO radar, and its engineering implementation is strong. The transmit and receive arrays are respectively placed in bases far away from each other, combining the advantages of bistatic radar and MIMO radar. However, the signal structure is more complicated, which has brought unique problems. Therefore, the bistatic MIMO radar configuration is considered.

At present, most of the research focuses on solving the problem of target localization, and few studies target tracking problems. In the bistatic MIMO radar, the target's direction of departure (DOD) and direction of arrival (DOA) are the main parameters that need to be estimated [19 – 26]. The estimation algorithms contain estimation of signal parameters via rotational invariance technique (ESPRIT) algorithms [19 – 22], Capon algorithms [23], multiple signal classification (MUSIC) algorithms [24 – 26], which are always discussed in MIMO radar when targets are fixed. The above algorithms cannot solve the target tracking problem due to their batch mode.

At present, there are a few literature studies on the target tracking problem. Wu et al. improved parallel factor (PARAFAC) analysis and successfully solved the angle tracking problem [27]. Kalman combined the projection approximation subspace tracking with deflation (PASTd) algorithm, which is applied to monostatic MIMO radar, and can realize the automatic correlation of the target angle at different time [28]. However, PARAFAC, PASTd and Kalman-PASTd algorithms have higher computational complexity. Therefore, Yu et al. proposed a low complexity tracking algorithm for monostatic MIMO radar, but it had approximate operation, resulting in low tracking performance [29].

The tracking algorithms in the above literature are all about monostatic MIMO radar and cannot be directly ap-

Manuscript received December 14, 2017.
*Corresponding author.
This work was supported by the National Natural Science Foundation of China (61671453; 61201379), and Anhui Natural Science Foundation of China (1608085MF123).

plied to solve the tracking problem of bistatic MIMO radar. In bistatic MIMO radar, the DOD and DOA are different and the joint steering vector is more complicated. A new signal model of bistatic MIMO radar is proposed, and the fractional ambiguity function is applied to the projection approximation subspace tracking (PAST) algorithm, solving the problem of tracking the azimuth and elevation of the target [30]. However, the algorithm is to solve the problem of locating ground interference sources, which lacks generality. A fast angle tracking algorithm was proposed in [31]. The relation between the difference of the covariance matrix and the difference of the target angle was deduced. DOD and DOA were obtained by the least square method. The algorithm has low computational complexity and can automatically associate angles. However, after two approximations, the tracking performance is poor. Wu et al. introduced the PASTd algorithm into bistatic MIMO radar and successfully solved the target tracking problem [32]. The tracking performance is better than the algorithm in [31], but it requires additional data correlation operations and cannot track the target with the same DOD or DOA. In order to overcome the shortcomings in [32], Zhang et al. proposed a target tracking algorithm based on adaptive asymmetric joint diagonalization (AAJD) [33]. The algorithm divides the objective function optimization process into two steps: the first step obtains the joint steering vector by optimizing the objective function; the second step uses the joint steering vector and the previous angle to estimate the angle of target. And then the joint steering vector is updated as the initial value of the first step in the next moment. The algorithm needs no additional pairing process, solves the shortcoming of [32] and increases the adaptability. However, the performance is reduced by reusing the estimated angle of the previous time.

The performance and computational complexity of the AAJD algorithm need to be further optimized. To solve the problem, the adaptive asymmetric joint diagonalization with deflation (AAJDd) tracking algorithm is proposed. First of all, it is proved that each column of the eigenvector computed by the AAJD algorithm corresponds to a target. Based on this, the AAJDd algorithm avoids reusing the estimation angle of the previous moment to improve the tracking performance. Secondly, the eigenvalue variables are obtained, which can be used to estimate the target number and solve the tracking problem without knowing the target number. At the same time, the AAJDd algorithm turns the matrix operation into a constant operation, thereby reducing the amount of computation and speeding up the convergence. Finally, the ESPRIT algorithm is improved to realize the automatic matching and association of the DOD and DOA. The AAJDd algorithm has higher tracking performance, especially when the target is maneuvering.

The reminder of this paper is structured as follows. Section 2 develops the data model for bistatic MIMO radar. Section 3 establishes our angle tracking algorithm based on AAJDd. In Section 4, simulation results are presented to verify the effectiveness of the proposed algorithm, while the conclusions are made in Section 5.

Notation. $(\cdot)^*$, $(\cdot)^{\mathrm{T}}$, $(\cdot)^{\mathrm{H}}$, $\| \ \|_F$ and $(\cdot)^{-1}$ denote conjugate, transpose, conjugate transpose, Frobenius norm and inverse operations, respectively. $\mathrm{Diag}(\boldsymbol{v})$ stands for diagonal matrix whose diagonal is a vector $\boldsymbol{v}$; $\boldsymbol{I}_K$ is a $K \times K$ identity matrix; $\otimes$ and $\oplus$ are the Kronecker product and Hadamard product, respectively.

2. Signal model

The schematic diagram of MIMO radar is illustrated in Fig. 1. Suppose the number of transmit elements and receiving elements are M and N respectively, the spacing between adjacent array elements is not required (including the statistical and coherent MIMO radar system), and the coordinates of the transmitting and receiving array elements are (T_{x_m}, T_{y_m}) and (R_{x_n}, R_{y_n}) $(m = 1, \ldots, M; n = 1, \ldots, N)$. The orthogonal waveforms are emitted by different transmit elements.

Suppose that there are P far-field moving point targets in the airspace, (x_i, y_i), $\varepsilon(x_i, y_i)$ and v_i denote coordinate, scattering coefficient and speed $(i = 1, 2, \ldots, P)$. The angles between the moving direction and the direction of DOD and DOA are θ'_{mi} and φ'_{in}. Suppose the scattering coefficient satisfies the Swerling II model and remains constant during a certain observation time (within a pulse period). The propagation time $\tau(x_i, y_i, x_k, y_k) = d(x_i, y_i, x_k, y_k)/c$ can be obtained by the distance information $d(x_i, y_i, x_k, y_k)$ between (x_i, y_i) and (x_k, y_k), where c denotes the light speed.

The emission signal of the mth transmitting array element is $s_m(t)\exp(\mathrm{j}2\pi f_c t)$, where $s_m(t)$ and f_c denote baseband signal and carrier frequency respectively. The signal transmitted by the mth transmitting element reaches the nth receiving element after being scattered by P targets. The output can be depicted as

$$\overline{x}_{n,m}(t) = \sum_{i=1}^{P} s_m[t - \tau(T_{x_m}, T_{y_m}, x_i, y_i) - \tau(R_{x_n}, R_{y_n}, x_i, y_i)] \cdot \varepsilon(x_i, y_i)\exp(\mathrm{j}\omega_{nim}t) \cdot \exp\{\mathrm{j}2\pi f_c[t - \tau(T_{x_m}, T_{y_m}, x_i, y_i) - \tau(R_{x_n}, R_{y_n}, x_i, y_i)]\} \tag{1}$$

where $\omega_{nim} = 2\pi\dfrac{v_i(\cos\theta'_{mi}+\cos\varphi'_{in})}{\lambda}$, $f_{nim} = \dfrac{v_i(\cos\theta'_{mi}+\cos\varphi'_{in})}{\lambda}$ denotes Doppler shift. It is assumed that the delay difference between MN channels is negligible. And the signal is available as a narrowband signal, so

$$s_m[t-\tau(T_{x_m},T_{y_m},x_i,y_i)-\tau(R_{x_n},R_{y_n},x_i,y_i)]\approx$$

$$s_m[t-\tau(T_{x_1},T_{y_1},x_i,y_i)-\tau(R_{x_1},R_{y_1},x_i,y_i)]=$$

$$s_m(t-\tau_i) \tag{2}$$

where

$$\begin{cases}\tau_i=\tau(T_{x_1},T_{y_1},x_i,y_i)+\tau(R_{x_1},R_{y_1},x_i,y_i)\\ \tau_{i,m}^{\mathrm{T}}=2\pi f_c[\tau(T_{x_m},T_{y_m},x_i,y_i)-\tau(T_{x_1},T_{y_1},x_i,y_i)]\\ \tau_{i,n}^{R}=2\pi f_c[\tau(R_{x_n},R_{y_n},x_i,y_i)-\tau(R_{x_1},R_{y_1},x_i,y_i)]\end{cases}. \tag{3}$$

Substituting (2) and (3) into (1), we get

$$\overline{x}_{n,m}(t)=\sum_{i=1}^{P}\varepsilon(x_i,y_i)s_m(t-\tau_i)\exp(-\mathrm{j}(\tau_{i,m}^{T}+\tau_{i,n}^{R}))\cdot$$

$$\exp(\mathrm{j}2\pi f_c(t-\tau_i))\exp(\mathrm{j}\omega_{nim}t). \tag{4}$$

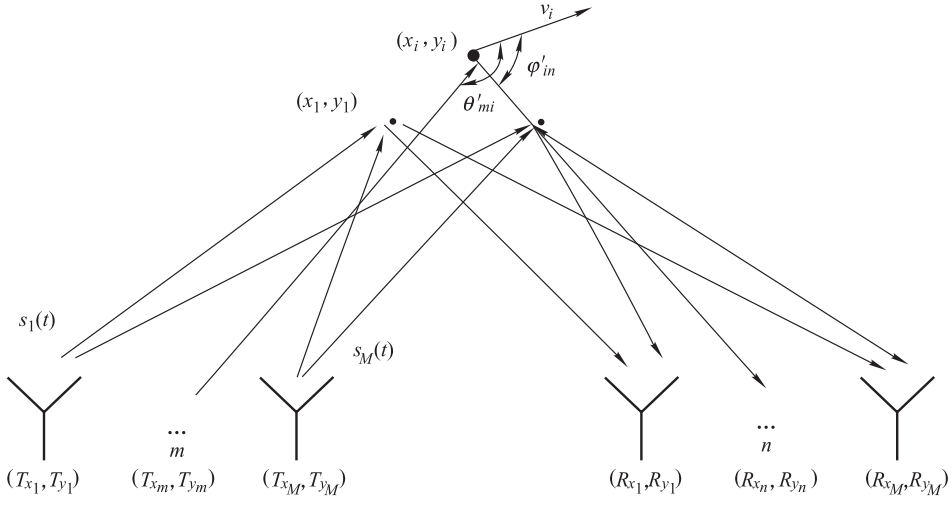

Fig. 1 General schematic diagram of MIMO radar

Suppose that the delay of the ith target matched filter is τ'_i. The output of signal in (4) after matched filtering is

$$x_{n,m}(t)=\overline{x}_{n,m}(t)\mathrm{e}^{-\mathrm{j}2\pi f_c(t-\tau'_i)}s_m(t-\tau'_i)^*=$$

$$\sum_{i=1}^{P}\varepsilon(x_i,y_i)\mathrm{e}^{\mathrm{j}2\pi f_c(\tau'_i-\tau_i)}\mathrm{e}^{\mathrm{j}\omega_{nim}t}\mathrm{e}^{-\mathrm{j}(\tau_{i,m}^{T}+\tau_{i,n}^{R})}\cdot$$

$$s_m(t-\tau_i)s_m(t-\tau'_i)^*. \tag{5}$$

When the delay estimation has no error, that is $\tau_i=\tau'_i$, (5) becomes

$$x_{n,m}(t)=\sum_{i=1}^{P}\varepsilon(x_i,y_i)\mathrm{e}^{\mathrm{j}\omega_{nim}t}\mathrm{e}^{-\mathrm{j}(\tau_{i,m}^{T}+\tau_{i,n}^{R})}. \tag{6}$$

The signal received by the nth receive element is $x_n(t)=\sum_{m=1}^{M}x_{n,m}(t)$. With noise, the output signal of the nth receiving array is

$$x_n(t)=\sum_{m=1}^{M}\sum_{i=1}^{P}\varepsilon(x_i,y_i)\mathrm{e}^{\mathrm{j}\omega_{nim}t}\mathrm{e}^{-\mathrm{j}(\tau_{i,m}^{T}+\tau_{i,n}^{R})}+v_n(t) \tag{7}$$

where $v_n(t)$ denotes the additive noise in the output signal of the nth element. $\varepsilon(x_i,y_i)\mathrm{e}^{\mathrm{j}\omega_{nim}t}$ and $\varepsilon(x_i,y_i)$ have the same statistical properties. We set the same spacing between the transceiver elements. Define the $M\times P$ dimensional transmit steering vector as $\boldsymbol{A}_t(\varphi)=[\boldsymbol{a}_t(\varphi_1),\boldsymbol{a}_t(\varphi_2),\ldots,\boldsymbol{a}_t(\varphi_P)]$. Among them, $\boldsymbol{a}_t(\varphi_i)=\boldsymbol{a}_t(x_i,y_i)$, $\boldsymbol{a}_t(x_i,y_i)=[1,\mathrm{e}^{-\mathrm{j}\tau_{i,2}^{T}},\ldots,\mathrm{e}^{-\mathrm{j}\tau_{i,M}^{T}}]^{\mathrm{T}}$, $\boldsymbol{a}_t(\varphi_i)=\{1,\exp(\mathrm{j}2\pi d_t\sin\varphi_i/\lambda),\ldots,\exp[\mathrm{j}2\pi(M-1)\cdot d_t\sin\varphi_i/\lambda]\}^{\mathrm{T}}$. Define the $N\times P$ dimensional receiver steering vector as $\boldsymbol{A}_r(\theta)=[\boldsymbol{a}_r(\theta_1),\boldsymbol{a}_r(\theta_2),\ldots,\boldsymbol{a}_r(\theta_P)]$. Among them, $\boldsymbol{a}_r(\theta_i)=\boldsymbol{a}_r(x_i,y_i)$, $\boldsymbol{a}_r(x_i,y_i)=[1,\mathrm{e}^{-\mathrm{j}\tau_{i,2}^{R}},\ldots,\mathrm{e}^{-\mathrm{j}\tau_{i,N}^{R}}]^{\mathrm{T}}$, $\boldsymbol{a}_r(\theta_i)=\{1,\exp(\mathrm{j}2\pi d_r\sin\theta_i/\lambda),\ldots,\exp[\mathrm{j}2\pi(N-1)d_r\sin\theta_i/\lambda]\}^{\mathrm{T}}$. According to (7), the receiving data of the lth pulse can be rewritten as

$$\boldsymbol{y}(t)=\boldsymbol{A}_t(\varphi)\odot\boldsymbol{A}_r(\theta)vec(\mathrm{diag}(\boldsymbol{d}_l(t)))+\boldsymbol{v}(t)=$$

$$\boldsymbol{W}(\varphi,\theta)vec(\mathrm{diag}(\boldsymbol{d}_l(t)))+\boldsymbol{v}(t)=$$

$$\boldsymbol{W}(\varphi,\theta)\boldsymbol{d}_l(t)+\boldsymbol{v}(t) \tag{8}$$

where $\boldsymbol{y}(t)=[y_1(t),y_2(t),\ldots,y_N(t)]^{\mathrm{T}}$ is the $N\times 1$ receive signal vector. $\boldsymbol{W}(\varphi,\theta)=[\boldsymbol{a}_r(\theta_1)\otimes\boldsymbol{a}_t(\varphi_1),\boldsymbol{a}_r(\theta_2)\otimes\boldsymbol{a}_t(\varphi_2),\ldots,\boldsymbol{a}_r(\theta_P)\otimes\boldsymbol{a}_t(\varphi_P)]$ is the $MN\times$

P joint steering vector. $\boldsymbol{d}_l(t) = [d_{l,1}\mathrm{e}^{\mathrm{j}\omega_1 t}, d_{l,2}\mathrm{e}^{\mathrm{j}\omega_2 t}, \ldots, d_{l,P}\mathrm{e}^{\mathrm{j}\omega_P t}]$ is a diagonal matrix, where $d_{l,i} = \varepsilon_l(x_i, y_i)$ denotes the complex coefficient of the ith target in the lth transmit pulse period. $\boldsymbol{v}(t) = [v_1(t), v_2(t), \ldots, v_N(t)]^{\mathrm{T}}$ is the $N \times 1$ additive Gaussian white noise vector.

3. AAJDd angle tracking algorithm

3.1 Adaptive asymmetric joint diagonalization algorithm

In order to ensure the structural integrity of this paper, this section provides an overview of the AAJD angle tracking algorithm in [33].

The optimization function of the AAJD tracking algorithm is

$$\min_{\boldsymbol{W}(t)\in\boldsymbol{\omega}} J(\boldsymbol{W}(t)) = \sum_{i=1}^{t} \beta^{t-i}\|\boldsymbol{y}(i) - \boldsymbol{W}(t)\boldsymbol{d}(i)\|^2 \quad (9)$$

where $\boldsymbol{\omega} = \{\boldsymbol{W}|\boldsymbol{W} = \boldsymbol{A}_t \odot \boldsymbol{A}_r, \boldsymbol{A}_t \in \boldsymbol{v}_t, \boldsymbol{A}_r \in \boldsymbol{v}_r\}$ is the constraint set, in which $\boldsymbol{v}_t$ and $\boldsymbol{v}_r$ satisfy the Vandermonde matrix form, β is the forgetting factor.

Since the eigenvector $\boldsymbol{W}(t)$ obtained directly by solving the optimization function does not fully satisfy the standard form of the steering vector. The AAJD algorithm divides the solution process into two steps.

(i) Recursion phase

$$\widehat{\boldsymbol{W}}(t) = \min_{\widehat{\boldsymbol{W}}(t)} J(\widehat{\boldsymbol{W}}(t)) = \sum_{i=1}^{t} \beta^{t-i}\|\boldsymbol{y}(i) - \widehat{\boldsymbol{W}}(t)\boldsymbol{d}(i)\|^2. \quad (10)$$

Let $\nabla J(\widehat{\boldsymbol{W}}(t))$ be the conjugate gradient of $J(\widehat{\boldsymbol{W}}(t))$ with respect to $\widehat{\boldsymbol{W}}(t)$, then we have

$$\nabla J(\widehat{\boldsymbol{W}}(t)) = \sum_{i=1}^{t} \beta^{t-i}[\widehat{\boldsymbol{W}}(t)\boldsymbol{d}(i)\boldsymbol{d}^{\mathrm{H}}(i) - \boldsymbol{y}(i)\boldsymbol{d}^{\mathrm{H}}(i)]. \quad (11)$$

Let $\nabla J(\widehat{\boldsymbol{W}}(t)) = 0$, we can get the following equations:

$$\widehat{\boldsymbol{W}}(t) = \boldsymbol{C}_{yd}(t)\boldsymbol{C}_{dd}^{-1}(t) \quad (12)$$

$$\boldsymbol{C}_{dd}(t) = \sum_{i=1}^{t} \beta^{t-i}\boldsymbol{d}(i)\boldsymbol{d}^{\mathrm{H}}(i) = \beta\boldsymbol{C}_{dd}(t-1) + \boldsymbol{d}(t)\boldsymbol{d}^{\mathrm{H}}(t) \quad (13)$$

$$\boldsymbol{C}_{yd}(t) = \sum_{i=1}^{t} \beta^{t-i}\boldsymbol{y}(i)\boldsymbol{d}^{\mathrm{H}}(i) = \beta\boldsymbol{C}_{yd}(t-1) + \boldsymbol{y}(t)\boldsymbol{d}^{\mathrm{H}}(t). \quad (14)$$

At this point, only $\boldsymbol{d}(t)$ is unknown. When the angle changes slowly, $\boldsymbol{d}(t)$ satisfies the following relationship:

$$\boldsymbol{d}(t) = \widehat{\boldsymbol{W}}^{-1}(t)\boldsymbol{y}(t) \approx \boldsymbol{W}^{-1}(t-1)\boldsymbol{y}(t). \quad (15)$$

From (12) to (15), the relationship between $\widehat{\boldsymbol{W}}(t)$ and $\boldsymbol{W}(t-1)$ is obtained and the tracking process is completed.

(ii) Regularization phase

The relationship between $\widehat{\boldsymbol{W}}(t)$ and $\boldsymbol{W}(t)$ satisfies the following constraints:

$$\boldsymbol{W}(t) = \min_{\boldsymbol{W}(t)\in\boldsymbol{\omega}} J'(\boldsymbol{W}(t)) = \|\widehat{\boldsymbol{W}}(t) - \boldsymbol{W}(t)\|^2 \quad (16)$$

$$\boldsymbol{W}(t) = \min_{\boldsymbol{W}(t)\in\boldsymbol{\omega}} J'(\boldsymbol{W}(t)) = \min_{\boldsymbol{W}(t)\in\boldsymbol{\omega}} \|\widehat{\boldsymbol{W}}(t) - \boldsymbol{W}(t)\|^2 = \min_{\boldsymbol{A}_r\in\boldsymbol{v}_r, \boldsymbol{A}_t\in\boldsymbol{v}_t} \|\widehat{\boldsymbol{W}}(t) - \boldsymbol{W}(t)\|^2 = \min_{\boldsymbol{A}_r\in\boldsymbol{v}_r, \boldsymbol{A}_t\in\boldsymbol{v}_t} \|\widehat{\boldsymbol{W}}(t) - \boldsymbol{A}_t(\varphi(t)) \odot \boldsymbol{A}_r(\theta(t))\|^2 = \min_{\boldsymbol{a}_r, \boldsymbol{a}_t} \sum_{p=1}^{P} \|\widehat{\boldsymbol{W}}_p(t) - \boldsymbol{a}_r(\theta(t))\boldsymbol{a}_t^{\mathrm{T}}(\varphi(t))\|^2. \quad (17)$$

The specific solution process is

$$\begin{cases} \boldsymbol{a}_{pt}(\varphi(t)) = \widehat{\boldsymbol{W}}_p^{\mathrm{T}}(t)\boldsymbol{a}_{pr}^*(\theta(t-1)) \\ \boldsymbol{a}_{pt}(\varphi(t)) = \boldsymbol{a}_{pt}(\varphi(t))/\|\boldsymbol{a}_{pt}(\varphi(t))\| \\ \boldsymbol{a}_{pr}(\theta(t)) = \widehat{\boldsymbol{W}}_p(t)\boldsymbol{a}_{pt}^*(\varphi(t)) \\ \boldsymbol{a}_{pr}(\theta(t)) = \boldsymbol{a}_{pr}(\theta(t))/\|\boldsymbol{a}_{pr}(\theta(t))\| \end{cases}. \quad (18)$$

The transmitting and receiving steering vectors of the target are obtained by (18). On this basis, DOD and DOA of the targets are obtained by combining the ESPRIT algorithm.

Finally, the steering vectors of t moments are updated by the DOD and DOA.

$$\boldsymbol{W}(t) = \boldsymbol{A}_t(\varphi(t)) \odot \boldsymbol{A}_r(\theta(t)) \quad (19)$$

$\boldsymbol{W}(t)$ is closer to the true signal steering vectors than $\widehat{\boldsymbol{W}}(t)$.

3.2 AAJDd algorithm

The AAJD algorithm in [22] obtains the joint steering vectors by solving the minimum value of the objective function which is used as the initial vector at the next moment. The algorithm overcomes the problem that the PASTd algorithm cannot track the targets of the same DOD or DOA, and can realize angle automatic association and matching. The reason is that the target angle at time t in the second step is obtained from the target angle information at time $t-1$ and the estimated joint steering vector at time t. The AAJD algorithm has a hypothetical precondition that the transmit and receive steering vectors $\boldsymbol{A}_t(\varphi(t))$ and $\boldsymbol{A}_r(\theta(t))$ may either be kept unchanged or be slowly time varying as $\boldsymbol{A}_t(\varphi(t)) = \boldsymbol{A}_t(\varphi(t-1))$ and $\boldsymbol{A}_r(\theta(t)) = \boldsymbol{A}_r(\theta(t-1))$.

Therefore, when observing fast moving targets, especially high maneuvering targets, the rate of change of angle

is large, resulting in low tracking accuracy of the AAJD algorithm. Otherwise, in the first step, the AAJD algorithm does not have the variable that can represent the eigenvalues which can be used to estimate the number of target. Aiming at these deficiencies of the AAJD algorithm, this paper proposes the AAJDd algorithm.

3.2.1 Estimating eigenvalue variables and steering vector

The AAJD algorithm takes the joint steering vector as a whole, so there is no variable in the AAJD algorithm that can characterize the eigenvalues. And it increases computational complexity at the same time.

The idea of sequential estimation of the principal component in [34,35] is introduced into the AAJD algorithm, which is based on the deflation technique. First, the AAJD algorithm with $p = 1$ is used to update the most dominant eigenvector (the most dominant eigenvector corresponding to the largest eigenvalue). Then the projection of the received data vector onto the most dominant eigenvector is removed, resulting in a new data vector. In this case, the second dominant eigenvector becomes the most dominant eigenvector in the new data vector. It can be extracted from the new data vector in the same way. Applying this procedure repeatedly, all desired eigencomponents are estimated sequentially.

Table 1 describes the AAJDd algorithm process. When $p = 1$, the main part of the AAJDd algorithm corresponds to the AAJD algorithm with a single target. The quantity $g_i(t)$ plays the same role as the $p \times p$ matrix $\boldsymbol{C}_{dd}(t) = \boldsymbol{P}^{-1}(t)$ in the AAJD algorithm.

It is worth noting that $\boldsymbol{W}_i(t)$ is the estimation result of the ith eigenvector of $\boldsymbol{C}(t) = E(\boldsymbol{y}(t)\boldsymbol{y}^{\mathrm{H}}(t))$, $g_i(t)$ is the corresponding eigenvalue estimation result, which can be used to estimate the target number.

Proposition 1 In MIMO radar, $g_i(t)$ is a variable corresponding to the eigenvector estimated by the AAJDd algorithm, $\lambda_i(t)$ is the eigenvalue obtained by decomposing the received data covariance matrix $\boldsymbol{C}(t)$. When the algorithm converges, $g_i(t)$ is the same as the eigenvalue $\lambda_i(t)$, which is the estimated eigenvalue variable.

Proof Because $\lambda(t)$ and $\boldsymbol{W}(t)$ are the eigenvalue and eigenvector obtained by decomposition of the data covariance matrix $\boldsymbol{C}(t)$. According to eigenvalue decomposition theory,

$$\boldsymbol{C}(t)\boldsymbol{W}_i(t) = \lambda_i(t)\boldsymbol{W}_i(t) \tag{20}$$

$$\lambda_i(t) = \boldsymbol{W}_i^{-1}(t)\boldsymbol{C}(t)\boldsymbol{W}_i(t). \tag{21}$$

According to the above analysis, we have

$$g_i(t) = C_{d_i d_i}(t) = E(\boldsymbol{d}_i(t)\boldsymbol{d}_i^{\mathrm{H}}(t)). \tag{22}$$

Substituting (15) into (22), we can obtain the following relationship:

$$g_i(t) = E(\boldsymbol{d}_i(t)\boldsymbol{d}_i^{\mathrm{H}}(t)) = E(\boldsymbol{W}_i^{-1}(t)\boldsymbol{y}(t)\boldsymbol{y}^{\mathrm{H}}(t)(\boldsymbol{W}_i^{-1}(t))^{\mathrm{H}}) = \boldsymbol{W}_i^{-1}(t)E(\boldsymbol{y}(t)\boldsymbol{y}^{\mathrm{H}}(t))(\boldsymbol{W}_i^{-1}(t))^{\mathrm{H}}. \tag{23}$$

Substituting $\boldsymbol{C}(t) = E(\boldsymbol{y}(t)\boldsymbol{y}^{\mathrm{H}}(t))$ and $\boldsymbol{W}_i^{\mathrm{H}}(t)\boldsymbol{W}_i(t) = \boldsymbol{W}_i^{-1}(t)\boldsymbol{W}_i(t) = \boldsymbol{I}$ into (23), we obtain

$$g_i(t)\boldsymbol{W}_i^{-1}(t) = \boldsymbol{W}_i^{-1}(t)\boldsymbol{C}(t)(\boldsymbol{W}_i^{-1}(t))^{\mathrm{H}}\boldsymbol{W}_i^{-1}(t) = \boldsymbol{W}_i^{-1}(t)\boldsymbol{C}(t) \tag{24}$$

and further simplification

$$g_i(t) = \boldsymbol{W}_i^{-1}(t)\boldsymbol{C}(t)\boldsymbol{W}_i(t). \tag{25}$$

Comparing (21) with (25), we can conclude that $g_i(t)$ and $\lambda_i(t)$ have the same function, which is the eigenvalue corresponding to the eigenvector. At this point, proposition 1 is proved. □

Table 1 AAJDd algorithm eigenvector solution flow

Initialization: $\boldsymbol{P}(0) = \boldsymbol{I}_{P\times P}, 0 < \beta \leqslant 1$
Input: $\boldsymbol{y}(t)$
Output: $\varphi(t), \theta(t)$
For $t = 1, \cdots, T$
  Step 1
    for $i = 1, \ldots, P$
      $\boldsymbol{d}_i(t) = \boldsymbol{W}_i^{-1}(t-1)\boldsymbol{y}_i(t)$
      $g_i(t) = \beta g_i(t-1) + |\boldsymbol{d}_i(t)|^2$
      $Q_i(t) = \boldsymbol{d}_i^{\mathrm{H}}(t)/g_i(t)$
      $\eta_i(t) = |\boldsymbol{d}_i(t)|^2/g_i(t)$
      $\boldsymbol{e}_i(t) = \boldsymbol{y}_i(t) - \boldsymbol{W}_i(t-1)\boldsymbol{d}_i(t)\widehat{\boldsymbol{W}}_i(t) = \boldsymbol{W}_i(t-1) + 1/\beta + \eta_i(t)\boldsymbol{e}_i(t)Q_i(t)$
      $\boldsymbol{y}_{i+1}(t) = \boldsymbol{y}_i(t) - \widehat{\boldsymbol{W}}_i^{-1}(t)\boldsymbol{d}_i(t)$
    end
  Step 2
  If $t == 1$
    $\boldsymbol{\Psi}_r = \boldsymbol{W}_{r1}^{-1}\boldsymbol{W}_{r2}$
    eigenvalue decomposition of $\boldsymbol{\Psi}_r$: $\boldsymbol{\Psi}_r = \boldsymbol{T}\boldsymbol{\Phi}_r\boldsymbol{T}^{-1}$
    take the diagonal elements of $\boldsymbol{\Phi}_r$, comprise ω_r
    $\boldsymbol{\Phi}_t = \boldsymbol{T}^{-1}\boldsymbol{W}_{t1}^{-1}\boldsymbol{W}_{t2}\boldsymbol{T}$
    take the diagonal elements of $\boldsymbol{\Phi}_t$, comprise ω_t
  else
    $\boldsymbol{\Psi}_r = \boldsymbol{W}_{r1}^{-1}\boldsymbol{W}_{r2}$
    take the diagonal elements of $\boldsymbol{\Psi}_r$, comprise ω_r
    $\boldsymbol{\Psi}_t = \boldsymbol{W}_{t1}^{-1}\boldsymbol{W}_{t2}$
    take the diagonal elements of $\boldsymbol{\Psi}_t$, comprise ω_t
  end
  using $\left\{ \begin{array}{l} \theta_p = \arcsin\{\mathrm{angle}[\boldsymbol{\omega}_r(p)]/\pi \\ \varphi_p = \arcsin\{\mathrm{angle}[\boldsymbol{\omega}_t(p)]/\pi \end{array} \right\}$, get $\theta(t), \varphi(t)$
  according to $\boldsymbol{W}(t) = [\boldsymbol{a}_t(\varphi_1) \otimes \boldsymbol{a}_r(\theta_1), \ldots, \boldsymbol{a}_t(\varphi_P) \otimes \boldsymbol{a}_r(\theta_P)]$
  updated $\widehat{\boldsymbol{W}}(t)$
End

To further illustrate the AAJDd algorithm, it is compared with the AAJD algorithm. $\boldsymbol{Q}(t)=\boldsymbol{P}(t-1)\boldsymbol{d}(t)$ and $\eta=\boldsymbol{d}^{\mathrm{H}}(t)\boldsymbol{Q}(t)$ in AAJD become $Q_i(t)=\boldsymbol{d}_i^{\mathrm{H}}(t)/g_i(t)$ and $\eta_i(t)=|\boldsymbol{d}_i(t)|^2/g_i(t)$ in the AAJDd algorithm. The last equation of the first step in Table 1 describes the deflation step. It subtracts the component of $\boldsymbol{y}_i(t)$ along the direction of the ith eigenvector $\boldsymbol{W}_i(t)$ from $\boldsymbol{y}_i(t)$. By comparison, we can find that the AAJDd algorithm turns the matrix operation into a constant operation, thereby reducing the amount of computation and speeding up the convergence of the algorithm. Compared with the AAJD algorithm, the AAJDd algorithm can express the feature decomposition process clearly.

The computational complexity of AAJDd and AAJD algorithms are $O(MNP+P)$ and $O(MNP+P^2)$. Comparing the computational complexity of two algorithms, we can see that computational complexity of the AAJDd algorithm is lower, the missing of the $O(P^2)$ term has the reason that no $P\times P$ matrices have to be computed. When the number of targets is large, the AAJDd algorithm can significantly reduce the computational complexity.

3.2.2 Angle estimation algorithm based on improved ESPRIT

Through the above analysis, we can see that the reason for the decrease of tracking performance of the AAJD algorithm is that the estimated angle information of the last time is used in the second step. When the target moves fast, the target angle difference between adjacent moments becomes larger, resulting in the decrease of the performance. Therefore, this paper improves the second step of the AAJD algorithm and proposes a new angle estimation algorithm based on the improved ESPRIT.

Let $\widehat{\boldsymbol{W}}(t)$ denote an eigenvector that is estimated through the AAJDd algorithm at moment t. $\boldsymbol{U}_s(t)$ is the real steering vector. They expand into the same signal subspace, that is

$$\mathrm{span}\{\widehat{\boldsymbol{W}}(t)\}=\mathrm{span}\{\boldsymbol{U}_s(t)\}. \tag{26}$$

In this case, there is a unique non-singular matrix $\boldsymbol{T}(t)$, and we can obtain the following relationship:

$$\widehat{\boldsymbol{W}}(t)=\boldsymbol{U}_s(t)\boldsymbol{T}(t). \tag{27}$$

Proposition 2 In MIMO radar, $\widehat{\boldsymbol{W}}(t)$ is the eigenvector estimated by the AAJDd algorithm, and $\boldsymbol{U}_s(t)$ is the corresponding real steering vector $(t>1)$. When the algorithm converges, there is a unique non-singular matrix $\boldsymbol{T}$, satisfying $\widehat{\boldsymbol{W}}(t)=\boldsymbol{U}_s(t)\boldsymbol{T}$, and $\boldsymbol{T}=\boldsymbol{Q}\boldsymbol{E}$, where $\boldsymbol{Q}$ is the scale factor matrix, $\boldsymbol{E}$ is the column exchange matrix.

Proof From [22], we can see that when the AAJDd algorithm converges, $\mathrm{span}\{\widehat{\boldsymbol{W}}(t)\}=\mathrm{span}\{\boldsymbol{U}_s(t)\}$, then $\widehat{\boldsymbol{W}}(t)=\boldsymbol{U}_s(t)\boldsymbol{T}(t)$. $\widehat{\boldsymbol{W}}(t)$ is obtained by solving $\nabla J(\widehat{\boldsymbol{W}}(t))=0$ at time t.

$$\nabla J(\widehat{\boldsymbol{W}}(t))=\sum_{i=1}^{t}\lambda^{t-i}[\widehat{\boldsymbol{W}}(t)\boldsymbol{d}(i)\boldsymbol{d}^{\mathrm{H}}(i)-\boldsymbol{y}(i)\boldsymbol{d}^{\mathrm{H}}(i)]=0. \tag{28}$$

The recursion formula is obtained from (28):

$$\widehat{\boldsymbol{W}}(t)=\boldsymbol{W}(t-1)+\frac{1}{\beta+\eta}\cdot$$
$$[\boldsymbol{y}(t)-\boldsymbol{W}(t-1)\boldsymbol{d}(t)]\boldsymbol{d}^{\mathrm{H}}(t)\boldsymbol{P}(t-1) \tag{29}$$

where $\eta=\boldsymbol{d}^{\mathrm{H}}(t)\boldsymbol{P}(t-1)\boldsymbol{d}(t)$ is constant, β is a forgotten factor, $0<\beta<1$.

The formula (22) is multiplied by $\boldsymbol{d}(t)$ on both sides, then

$$\widehat{\boldsymbol{W}}(t)\boldsymbol{d}(t)=\boldsymbol{W}(t-1)\boldsymbol{d}(t)+\frac{\eta}{\beta+\eta}[\boldsymbol{y}(t)-\boldsymbol{W}(t-1)\boldsymbol{d}(t)] \tag{30}$$

and further simplification

$$\widehat{\boldsymbol{W}}(t)\boldsymbol{d}(t)=\frac{\eta}{\beta+\eta}\boldsymbol{y}(t)+\frac{\beta}{\beta+\eta}\boldsymbol{W}(t-1)\boldsymbol{d}(t). \tag{31}$$

Taking $\boldsymbol{d}(t)=\widehat{\boldsymbol{W}}^{-1}(t)\boldsymbol{y}(t)$ into (31), we obtain

$$\widehat{\boldsymbol{W}}(t)=\boldsymbol{W}(t-1)=\boldsymbol{U}_s(t)\boldsymbol{T}(t)=\boldsymbol{U}_s(t-1)\boldsymbol{T}(t-1). \tag{32}$$

The change of the steering vector of the adjacent time in [22] is negligible, then

$$\boldsymbol{U}_s(t)\simeq\boldsymbol{U}_s(t-1). \tag{33}$$

Substituting (26) into (25), we obtain

$$\boldsymbol{T}(t)=\boldsymbol{T}(t-1)=\boldsymbol{T}. \tag{34}$$

$\boldsymbol{W}(t-1)$ is reconstructed by angles obtained in Step 2 of the AAJDd algorithm, so the updated $\boldsymbol{W}(t-1)$ and $\boldsymbol{U}_s(t-1)$ satisfy the column transformation. We can obtain the following relationship:

$$\boldsymbol{T}=\boldsymbol{Q}\boldsymbol{E} \tag{35}$$

where $\boldsymbol{Q}$ is the scale factor matrix, $\boldsymbol{E}$ is the column exchange matrix.

So far, the proof of Proposition 2 has been completed. The procedure of Step 2 of the AAJDd algorithm is given below. □

According to Proposition 2, $\widehat{\boldsymbol{W}}(t)=\boldsymbol{U}_s(t)\boldsymbol{T}$ and $\boldsymbol{T}=\boldsymbol{Q}\boldsymbol{E}$.

Using the ESPRIT algorithm, define $\boldsymbol{W}_{r1}$ and $\boldsymbol{W}_{r2}$ as the $M(N-1)\times P$ sub-matrices formed by the first and last $M(N-1)$ rows of $\widehat{\boldsymbol{W}}(t)$, respectively. According to

the principle of rotation available, we get $\boldsymbol{W}_{r1} = \boldsymbol{U}_{r1}\boldsymbol{T}$, $\boldsymbol{W}_{r2} = \boldsymbol{U}_{r2}\boldsymbol{T}$, $\boldsymbol{U}_{r2} = \boldsymbol{U}_{r1}\boldsymbol{\Phi}_r$, where $\boldsymbol{T}$ is a non-singular transformation matrix. Then $\boldsymbol{T}^{-1}\boldsymbol{\Phi}_r\boldsymbol{T} = \boldsymbol{W}_{r1}^{-1}\boldsymbol{W}_{r2} = \boldsymbol{\Psi}_r$, where $\boldsymbol{T}$ is just a column transformation and scaling operations, $\boldsymbol{T}^{-1}\boldsymbol{\Phi}_r\boldsymbol{T}$ just change the position of the diagonal elements of $\boldsymbol{\Phi}_r$. Therefore, the matrix which contains the DOA parameters information can be obtained by taking the elements on the diagonal of $\boldsymbol{\Psi}_r$. Similarly, define $\boldsymbol{W}_{t1}$ and $\boldsymbol{W}_{t2}$ as the $N(M-1)\times P$ sub-matrices formed by the first $M(k-1)+l$ and the last $M(k-1)+l+1$ rows of $\widehat{\boldsymbol{W}}(t)$ respectively $(k = 1, 2, \ldots, N, l = 1, 2, \ldots, M-1)$. Then $\boldsymbol{T}^{-1}\boldsymbol{\Phi}_t\boldsymbol{T} = \boldsymbol{W}_{t1}^{-1}\boldsymbol{W}_{t2} = \boldsymbol{\Psi}_t$. Therefore, the matrix which contains the DOD parameters information can be obtained by taking the elements on the diagonal of $\boldsymbol{\Psi}_t$.

Note that when $t = 1$, Proposition 2 is not valid. Therefore, the angle estimation algorithm in [21] is used. The procedure of angle estimation is shown in Step 2 of Table 1.

Through the conclusion of Proposition 2, we can see that the matrix $\boldsymbol{T}$ does not change with time. Each column of $\widehat{\boldsymbol{W}}(t)$ corresponds to a target's steering vector, and the order is invariant. Therefore, the target sequence does not change, completing the automatic association of angles. There is no feature decomposition operation in Step 2. Therefore, the DOD and DOA estimated by each column of $\boldsymbol{W}(t)$ correspond to one target, and the angle automatic matching of the same target is completed. At this point, the automatic pairing and association of DOD and DOA are realized.

3.3 Performance analysis of algorithms

In the AAJDd algorithm, eigenvalue variables are obtained, which can be used to estimate the number of targets and solve the target angle tracking problem when the number of targets is unknown. Each column of $\boldsymbol{W}_i(t)$ represents a steering vector of a target. When the target velocity is different, different forgetting factors may be set to improve the tracking speed and performance of the algorithm. The proposed method is general and suitable for a broad class of angle tracking algorithms based on subspace methods.

The computational complexity of AAJD and AAJDd algorithms are $O(MNP + P^2)$ and $O(MNP + P)$ every update respectively. When the number of targets to be estimated is large, the AAJDd algorithm can greatly reduce the computation. Fig. 2 shows that the computational complexity of the loop part of the proposed algorithm is lower than the AAJD tracking algorithm.

Step 2 in the AAJD algorithm estimates the target angle by using the last moment angle, reducing the performance of the algorithm. Proposition 2 proves that the eigenvectors obtained in Step 1 of AAJDd and the joint steering vectors are only column transformations. Each column of eigenvectors represents a target steering vector. Based on this, the AAJDd algorithm improves the AAJD algorithm and improves the tracking performance.

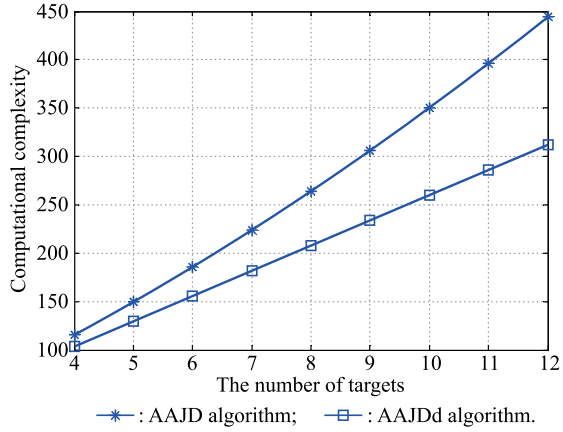

Fig. 2 Complexity comparison with $M = N = 5$ and different P

4. Simulation results

Assuming that the transmit and receive arrays of bistatic MIMO radars are both linearly configured, array spacing is all half-wavelength, the carrier frequency is 1 GHz, the transmit pulse width is 10 μs, the pulse repetition frequency is 10 kHz. The transmit waveform uses a Hadamard code pulse (HCP) signal. The target scattering coefficient d_p and additive noise are randomly generated.

Signal to noise ratio (SNR) is defined as $\mathrm{SNR} = 10\lg\dfrac{\sum_{p=1}^{P}\sigma_p^2}{P\sigma_e^2}$, where σ_p and σ_e represent the target scattering coefficient and noise power, respectively. The performance is measured in terms of averaged root mean square error (RMSE) over all targets: $\mathrm{RMSE} = \dfrac{1}{M}\sum_{m=1}^{M}\sqrt{\dfrac{1}{P}\sum_{p=1}^{P}\dfrac{1}{T}\sum_{t=1}^{\mathrm{T}}[(\widehat{\theta}_{k,m,t}-\theta_{k,m,t})^2]}$, where $\widehat{\theta}_{k,m,t}$ is the estimated value of target angle $\theta_{k,m,t}$ in the mth Monte Carlo experiment at time t, M is the number of Monte Carlo.

Experiment 1 Multiple target location

In order to verify the validity of the algorithm, we assume there are four far-field point targets in the air, located at $(\theta_1, \varphi_1) = (30°, -40°)$, $(\theta_2, \varphi_2) = (40°, 20°)$, $(\theta_3, \varphi_3) = (-50°, 20°)$ and $(\theta_4, \varphi_4) = (-50°, -20°)$, $M = N = 3$ transceivers are considered $\mathrm{SNR} = 10$ dB,

the number of transmitted pulses is $K = 100$, and the forgetting factor is $\beta = 0.95$.

To verify the effectiveness of the algorithm, the AAJDd algorithm is used to locate the target. The simulation results of the AAJDd algorithm are shown in Fig. 3.

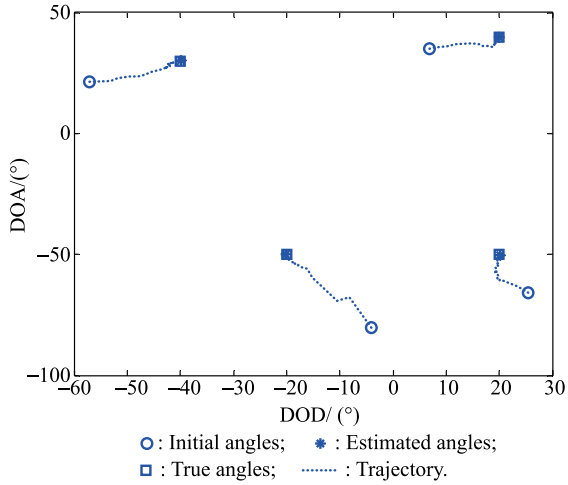

Fig. 3 Trajectory of the estimation procedure

It can be seen from Fig. 3 that the AAJDd algorithm can successfully track the multi-target angle. The final estimated angles in the graph coincide with the true angles, indicating that the algorithm gets the true angle. And the estimated trajectory of each target angle is approximately straight line, which shows the convergence speed is fast. Through experimental observation, it can be found that the target angle can be accurately estimated by only 40 pulses.

Comparing the DOA and DOD of the four targets, it can be seen that the DOD and DOA of the third target are the same as DOD of the target two and DOA of the target four respectively, indicating that the algorithm can estimate the target positions of the same DOD or DOA, making up the lack of the PASTd algorithm.

Experiment 2 Multiple target tracking

We assume there are five moving point targets in the air, $M = N = 4$ transceivers are considered and SNR $=$ 10 dB, the number of transmitted pulses is $K = 500$, the forgetting factor is $\beta = 0.85$. The simulation results of the AAJDd algorithm are shown in Fig. 4.

Fig. 4 shows the estimated trajectories coincide with true trajectories after approximately 40 pulses, which shows that the proposed algorithm can track the target moving trajectory successfully. The initial angle is randomly generated, and it can be seen that the trajectory from the initial value to the starting point of the target motion is approximately linear, indicating that the proposed algorithm can track the target quickly and has good convergence.

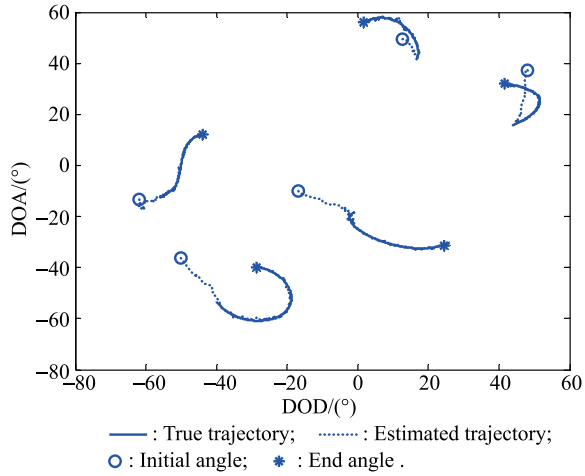

Fig. 4 Target tracking results of AAJDd algorithm

The AAJD algorithm, AAJDd algorithm and PASTd algorithm tracking error are compared to further illustrate the effectiveness of the algorithm in this paper, the simulation conditions are the same as the previous experiment, the simulation results are shown in Fig. 5. The tracking error is defined as $\text{Error} = \frac{1}{P}\sqrt{\sum_{p=1}^{P} E(\widehat{\theta}_p - \theta_p)^2 + E(\widehat{\varphi}_p - \varphi_p)^2}$.

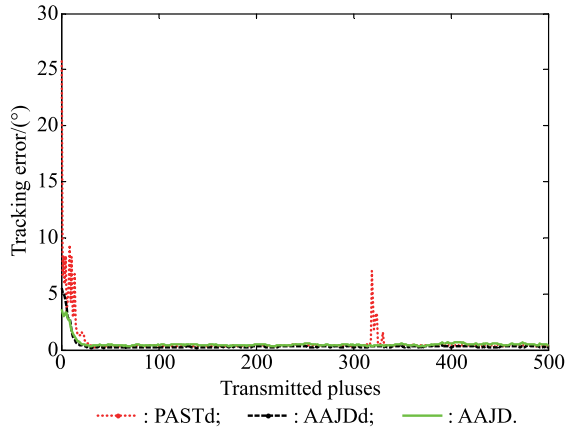

Fig. 5 Three algorithms tracking performance with SNR = 10 dB

It can be seen from Fig. 5 that the AAJD algorithm, AAJDd algorithm and PASTd algorithm can all successfully track the target. However, the PASTd algorithm fails when the number of pulses is 319. The DOD of the first and second target is the same, resulting in the failure of the PASTd algorithm. The AAJD and AAJDd algorithms successfully solve the lack of the PASTd algorithm.

In order to further verify the performance advantages of the AAJDd algorithm, the tracking performance of the AAJDd algorithm is compared with the AAJD algorithm.

The AAJDd algorithm has a greater advantage when the target is highly maneuvering. Assuming that the speed of the five goals is accelerating, the basic motion model does not change, the forgetting factor is $\beta = 0.85$. The rest of the simulation conditions remains unchanged. Fig. 6 shows the tracking performance of the AAJD algorithm and the AAJDd algorithm.

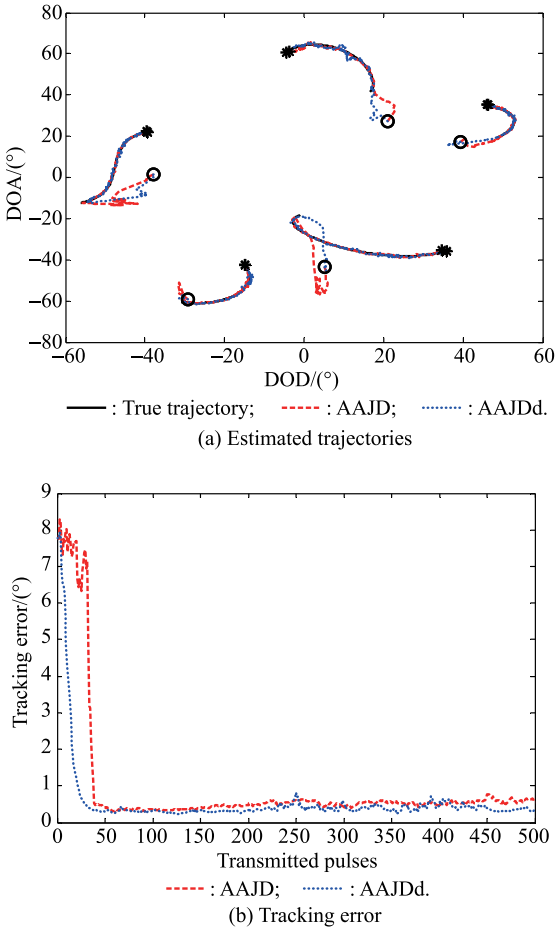

Fig. 6 Tracking DOD and DOA of five targets for $K = 500$

Fig. 6(a) shows the trajectories of the two algorithms coincide with the real trajectories of the target, indicating that the AAJD algorithm and AAJDd algorithm can still successfully track the target when the speed is high, but the tracking effect of the AAJD algorithm is poor. Fig. 6(b) shows the tracking error results of the two algorithms. It can be seen that the tracking error of the AAJDd algorithm is lower than the AAJD algorithm, and the AAJDd algorithm has better tracking performance and is suitable for high maneuvering target tracking, which is the same as the theoretical analysis.

Experiment 3 Target number estimation and angle tracking

To illustrate the performance of the proposed algorithm, we assume that the number of targets is changing. The number of transmitted pulses is $K = 500$, and the number of transceivers is $M = N = 5$. During $1 \leqslant K \leqslant 500$, there are always three moving points in the air. During $150 < K < 300$, two new targets appear in the observational airspace. During $400 < K < 450$, a new target appears in the observational airspace. The simulation results are shown in Fig. 7.

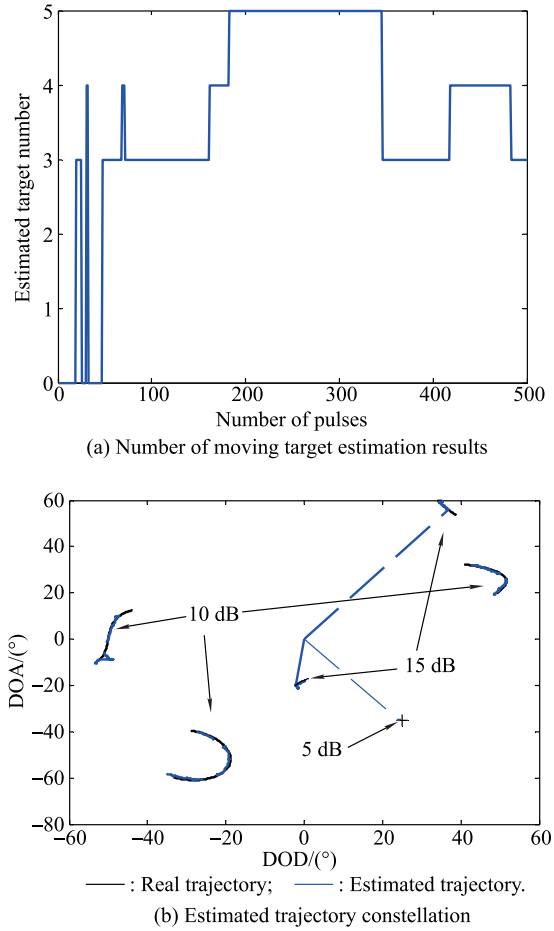

Fig. 7 Tracking results of the number and angle of the target

Fig. 7 shows that the number and angle of the target estimated by the AAJDd algorithm are correct, indicating that the algorithm can track the angles of the moving targets without knowing the number of targets.

Experiment 4 RMSE of the algorithms

We assume that there are five large maneuvering point targets, $M = N = 5$ transceivers are considered and $\mathrm{SNR} = -10 - 10$ dB, the number of transmitted pulses is $K = 400$, the number of Monte Carlo is $M = 100$. In order to make the performance comparison meaningful, the data in the tracking steady-state is used to calculate

the RMSE. After observing experimentally, the algorithm converges after the pulse number 100, $100 \leqslant K \leqslant 400$ is used to solve RMSE, The simulation results are shown in Fig. 8.

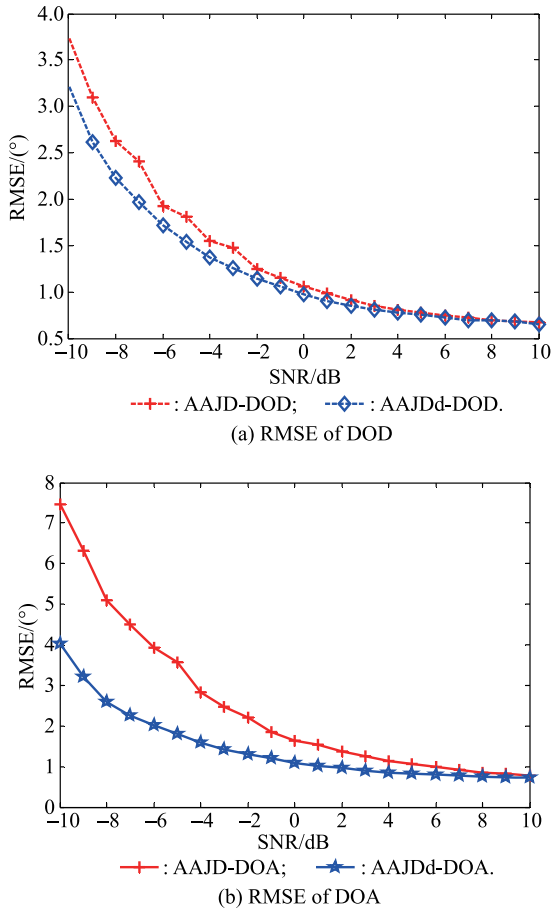

Fig. 8 RMSE versus SNR for two competitors

It can be seen from Fig. 8 that the RMSE of the AAJDd algorithm is lower than the AAJD algorithm, indicating that the tracking performance of the AAJDd is better. With the decrease of SNR, the performance gap between the two algorithms becomes larger, because the performance of the AAJD algorithm depends on the estimation accuracy of the previous time angle. When the SNR is low, the estimation error of the last time angle is larger, which affects the tracking performance and has the error accumulation.

5. Conclusions

In order to solve the problem of low performance of the AAJD tracking algorithm for bistatic MIMO radar, this paper proposes the AAJDd algorithm to avoid reusing the previous angles in the AAJD algorithm. On this basis, sequential estimation of the principal component is introduced to improve the tracking speed. Meanwhile our algorithm obtains the eigenvalues, which can be used to estimate the number of targets. Simulation results show that the AAJDd algorithm can effectively improve the target tracking performance. As shown in Fig. 7, the proposed algorithm can estimate the number of targets, but it cannot track the number of targets. In the future, we will be committed to studying the technique for joint tracking the number and angle of the target.

References

[1] FISHER E, HAIMOVICH A, BLUM R S, et al. Spatial diversity in radar-models and detection performance. IEEE Trans. on Signal Processing, 2006, 54(3): 823 – 838.
[2] YU X X, CUI G L, ZHANG T X, et al. Constrained transmit beampattern design for colocated MIMO radar. Signal Processing, 2018, 144: 145 – 154.
[3] LIU J, ZHOU S H, LIU W J, et al. Tunable adaptive detection in colocated MIMO radar. IEEE Trans. on Signal Processing, 2018, 66(4): 1080 – 1092.
[4] ABTAHI A, GAZOR S, MARVASTI F. Off-grid localization in MIMO radars using sparsity. IEEE Signal Processing Letters, 2018, 25(2): 313 – 317.
[5] THAMERI M, ABED-MERAIM K, FOROOZAN F, et al. On the statistical resolution limit (SRL) for time-reversal based MIMO radar. Signal Processing, 2018, 144: 373 – 383.
[6] HU J F, CHEN H W, LI Y, et al. Moving target parameter estimation for MIMO radar based on the improved particle filter. Multidimensional Systems and Signal Processing, 2018, 29(1): 1 – 17.
[7] GAO C C, TEH K C, LIU A F. Piecewise nonlinear frequency modulation waveform for MIMO radar. IEEE Journal of Selected Topics in Signal Processing, 2017, 11(2): 379 – 390.
[8] HERBERT S, HOPGOOD J R, MULGREW B. MMSE adaptive waveform design for active sensing with applications to MIMO radar. IEEE Trans. on Signal Processing, 2018, 66(5): 1361 – 1373.
[9] WEN F Q, ZHANG Z J, WANG K, et al. Angle estimation and mutual coupling self-calibration for ULA-based bistatic MIMO radar. Signal Processing, 2018, 144: 61 – 67.
[10] TANG B, ZHANG Y, TANG J. An Efficient minorization maximization approach for MIMO radar waveform optimization via relative entropy. IEEE Trans. on Signal Processing, 2018, 66(2): 400 – 411.
[11] WEN F Q, XIONG X D, SU J, et al. Angle estimation for bistatic MIMO radar in the presence of spatial colored noise. Signal Processing, 2017, 134(C): 261 – 267.
[12] CHENG Z Y, HE Z S, LI R Y, et al. Robust transmit beampattern matching synthesis for MIMO radar. Electronics Letters, 2017, 53(9): 620 – 622.
[13] WANG X H, ZHANG G, WEN F Q, et al. Angle estimation for bistatic MIMO radar with unknown mutual coupling based on three-way compressive sensing. Journal of Systems Engineering and Electronics, 2017, 28(2): 257 – 266.
[14] SHI J P, HU G P, ZHOU H. Entropy-based multipath detection model for MIMO radar. Journal of Systems Engineering and Electronics, 2017, 28(1): 51 – 57.
[15] ZHANG Y, ZHANG G, WANG X H. Computationally efficient DOA estimation for monostatic MIMO radar based on covariance matrix reconstruction. Electronics Letters, 2017, 53(2): 111 – 113.
[16] ZHANG Q, ZHANG L R, ZHENG G M, et al. Joint DOD and DOA estimation for bistatic MIMO radar with arbitrary ar-

ray using semi-real-valued MUSIC. Systems Engineering and Electronics, 2016, 38(3): 532 – 538. (in Chinese)
[17] HAIMOVICH A M, BLUM R S, CIMINI L J. MIMO radar with widely separated antenna. IEEE Signal Processing Magazine, 2008, 25(1): 116 – 129.
[18] LI J, STOICA P. MIMO radar with colocated antennas. IEEE Signal Processing Magazine, 2007, 25(5): 106 – 114.
[19] ZHENG G M, TANG J, YANG X. ESPRIT and unitary ESPRIT algorithms for coexistence of circular and noncircular signals in bistatic MIMO radar. IEEE Access, 2016, 4(99): 7232 – 7240.
[20] LIANG H, CUI C, YU J. Reduced-dimensional DOA estimation based on ESPRIT algorithm in monostatic MIMO radar with cross array. Journal of Electronics & Information Technology, 2016, 38(1): 80 – 89. (in Chinese)
[21] FAYAD Y, WANG C Y, CAO Q S. Temporal-spatial subspaces modern combination method for 2D-DOA estimation in MIMO radar. Journal of Systems Engineering and Electronics, 2017, 28(4): 697 – 702.
[22] LI J F, JIANG D F, ZHANG X F. DOA estimation based on combined unitary ESPRIT for coprime MIMO radar. IEEE Communications Letters, 2017, 21(1): 96 – 99.
[23] ZHANG X, HUANG Y, CHEN C, et al. Reduced complexity Capon for direction of arrival estimation in a monostatic multiple-input multiple-output radar. IET Radar, Sonar and Navigation, 2012, 6(8): 796 – 801.
[24] ZHANG Q, ZHANG L R, ZHENG G M, et al. Joint DOD and DOA estimation for bistatic MIMO radar with arbitrary array using semi-real-valued MUSIC. Systems Engineering and Electronics, 2016, 38(3): 532 – 538. (in Chinese)
[25] TAN J, NIE Z P, WEN D B. Low complexity MUSIC-based direction-of-arrival for monostatic MIMO radar. Electronics Letters, 2017, 53(4): 275 – 277.
[26] ZAHERNIA A, DEHAHANI M J, JAVIDAN R. MUSIC algorithm for DOA estimation using MIMO arrays. Proc. of the 6th International Conference on Telecommunication Systems, Services, and Application, 2011: 149 – 153.
[27] WU H L, ZHANG X F. DOA tracking in monostatic MIMO radar using PARAFAC-RLST algorithm. Proc. of the 3rd International Conference on Information Science and Engineering, 2011: 958 – 961.
[28] ZHANG X F, LI J F, FENG G P, et al. Kalman-PASTd based DOA tracking algorithm for monostatic MIMO radar. Proc. of the International Conference on Information, Services and Management Engineering, 2011: 220 – 224.
[29] YU H X, ZHANG X F, CHEN X Q, et al. Computationally efficient DOA tracking algorithm in monostatic MIMO radar with automatic association. International Journal of Antennas and Propagation, 2014, 2014(12): 1 – 10.
[30] LI L, QIU T S. A novel algorithm for target parameter estimation and dynamic tracking in bistatic MIMO radar system. Journal of Signal Processing, 2014, 30(2): 155 – 162. (in Chinese)
[31] ZHANG Z Y, LI X B, XU X Y, et al. Target angle rapid tracking algorithm for bistatic MIMO radar. Journal of Signal Processing, 2016, 32(6): 701 – 706. (in Chinese)
[32] WU H L, ZHANG X F. DOD and DOA tracking for bistatic MIMO radar using PASTd without additional angles pairing. Proc. of the 5th IEEE International Conference on Advanced Computational Intelligence, 2012: 1132 – 1136.
[33] ZHANG W T, LOU S T, LI X J, et al. Tracking multiple target in MIMO radar via adaptive asymmetric joint diagonalization. IEEE Trans. on Signal Processing, 2016, 64(11): 2880 – 2893.
[34] YANG J F, KAVEH M. Adaptive eigensubspace algorithms for direction or frequency estimation and tracking. IEEE Trans. on Acoustics Speech & Signal Processing, 2002, 36(2): 241 – 251.
[35] BANNOUR S, AZIMISADJADJ M R. An adaptive approach for optimal data reduction using recursive least squares learning method. Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing, 1992: 297 – 300.

Biographies

ZHANG Zhengyan was born in 1991. He received his M.S. degree in signal and information processing from Electronic Engineering Institute in 2012. Now he is a Ph.D. candidate in National University of Defense Technology. His main research interests are MIMO radar signal processing and array signal processing.
E-mail: zzyaisj@163.com

ZHANG Jianyun was born in 1963. He is a professor and doctor advisor of National University of Defense Technology and a senior member of Chinese Institute of Electronics. His main research interest is radar signal processing.
E-mail: zjy921@sina.com