

Research Article

L-Shaped-Sensor-Array-Based Localization and Tracking Method for 3D Maneuvering Target

Xing Zhang and Xue Wang

State Key Laboratory of Precision Measurement Technology and Instrument, Department of Precision Instruments, Tsinghua University, Beijing 100084, China

Correspondence should be addressed to Xue Wang; wangxue@mail.tsinghua.edu.cn

Received 7 September 2012; Accepted 19 December 2012

Academic Editor: Ruchuan Wang

Copyright © 2013 X. Zhang and X. Wang. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The localization and tracking technology for a three-dimensional target, which is a kernel problem in the military area, has received more and more attention. This paper proposes a closed-loop system to detect 3D maneuvering targets, including data acquisition, the direction of arrival (DOA) estimation, the triangle localization, and a trajectory prediction. This system firstly uses several L-shaped sensor arrays to sample the signals of maneuvering targets. Then the 2D ESPRIT algorithm and a maximum likelihood algorithm are introduced to achieve the positions of the spatial targets. Thirdly an autoregressive (AR) particle filter (PF) algorithm is realized to predict the locations in the next moment. Finally the localization process is directed by using the predicted positions to form a positive feedback closed loop. Experiment results show that this system can enhance the robustness and accuracy of the localization and tracking for three-dimensional maneuvering targets.

1. Introduction

Localization and tracking play a critical role in military applications and such as radar warning, battlefield monitoring. The kernel problem of each application is the ability of accurate localization. The most popular location tool is the Global Positioning System (GPS) [1]. Sometimes the location accuracy of the GPS is less than 0.1 m, but this sensor is usually used for self-localization. The multimedia sensor is another big type of location tools, such as CCD sensor [2, 3], stereo camera [4], and infrared sensor. However, multimedia sensor needs an extremely high image resolution, a fast network flow rate, and distinguished feature which limits its extensive use. RFID [5], wireless sensor networks [6], hybrid sensor networks [7] are generally used for the short-range target localization because their wireless transmission is easily influenced by the circumstance and the distance.

The reasons why the sensor array [8, 9, 10, 11, 12] is chosen in this paper to localize targets are as follows: sensor array is able to localize multiple targets simultaneously; sensor array can certainly eliminate the same noise and interference which can increase the signal noise ratio of the

received data; and the localization and tracking system using sensor array is little impacted by the environment which can increase the accuracy reliability and robustness. Sensor array is widely used for localization. For example, Tidd used an 8-element uniform circular sensor array to precisely localize the far-field target by multiple signals classification algorithm [9]. Fallon and Godsill [10] and Talantzis [11] use a microphone array to locate the acoustic source. Lin compared the performance of DOA estimation with different types of sensor array, such as cross array, circular arrays, and L-shaped array. The Clay Metro lower boundary of uniform L-shaped array was almost lowest except circular array. The direction-finding resolution and orientation consistency of the L-shaped array were much better than circular array [12].

After well localization, tracking process is always carried out to enhance the performance of the localization and tracking system. Tidd use the Kalman filter to deal with the DOA and speed of the target to offer strategic, fast, and low-cost advantages [9]. Fallon and Godsill designed a track before detect framework to obtain the probability distribution of the particle filter which could guide the resample process to create better particles in a comparative

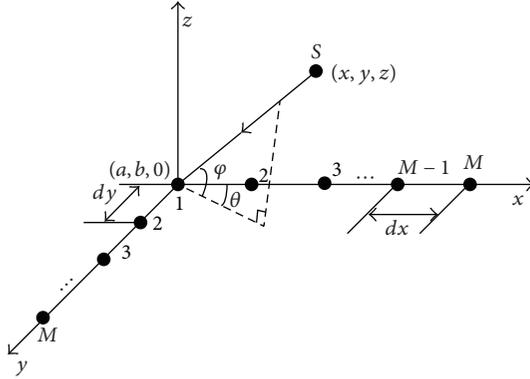


FIGURE 1: A uniform L-shaped sensor array receiving far field narrowband signal.

computational load [10]. This framework could increase the stability of the tracking system. Talantzis used the particle filter to predict the target trajectory [11]. The predicted position was used to estimate the accuracy of the localization by the microphone array and then replaced the inaccurate localization, sometimes as the true localization value when targets could not be detected.

This paper proposes a closed-loop localization and tracking system for three-dimensional targets. Firstly, choose several uniform L-shaped arrays to sample the spatial target signals and estimate the DOA of the 3D target with a two-dimensional estimating signal parameter via a rotational invariance techniques (2D ESPRIT) algorithm. Then the positions of the targets are calculated by a novel maximum likelihood (ML) algorithm according to the estimated DOA and the location of L-shaped sensor arrays. Thirdly the autoregressive particle filter algorithm is proposed to predict the target trajectory the in next time. Finally, the system uses the predicted position to guide the localization process. This system combines the localization and tracking algorithm together to form a positive feedback closed loop. This loop can increase the stability, robustness, and accuracy of the localization and tracking system.

The outline of this communication is as follows. In Section 2, the signal reception model based on the sensor array is developed. The proposed closed-loop localization and tracking system is described in Section 3. In Section 4, a series of experiment results are made to validate that the regenerative feedback system can work well to enhance the performance of the localization and tracking system. Finally the conclusion and perspectives are presented in Section 5.

2. Signal-Reception Model of Sensor Array

We consider a uniform L-shaped sensor array to localize and track the 3D maneuvering target. Make the intersected sensor as the origin and the arms of L type array as the axis to build a virtual coordinate system (shown in Figure 1). Each arm of the array have M elements. The sensors in the array are of the same nature. The distance between adjacent elements in x -axis is dx and in y -axis is dy .

L-shaped sensor array is used for collecting the narrow band signals emitted from the far-field target located at (x, y, z) . The wavelength of signal is λ . (ϕ, θ) is the DOA of the spatial target. The elevation angle $\phi \in [0, 90]^\circ$. Azimuth angle $\theta \in (-180, 180]^\circ$.

The array manifold vector, called the steering vector, of the uniform L-shaped array is given as

$$\begin{aligned} \mathbf{a}_x(\phi, \theta) &= [1, u(\phi, \theta), \dots, u^{M-1}(\phi, \theta)]^T, \\ \mathbf{a}_y(\phi, \theta) &= [1, v(\phi, \theta), \dots, v^{M-1}(\phi, \theta)]^T, \end{aligned} \quad (1)$$

$$\text{with } \begin{cases} u^k(\phi, \theta) = \exp\left(j2\pi d_x \times k \times \cos \phi \cos \frac{\theta}{\lambda}\right) \\ v^k(\phi, \theta) = \exp\left(j2\pi d_y \times k \times \cos \phi \sin \frac{\theta}{\lambda}\right), \end{cases}$$

where $(\cdot)^T$, j , x , y denote matrix transposition, imaginary unit, x -axis, and y -axis, respectively. The snapshot signal vector $\mathbf{X}(t) = [x_1(t), x_2(t), \dots, x_M(t)]^T$, $\mathbf{Y}(t) = [y_1(t), y_2(t), \dots, y_M(t)]^T$ on x - and y -axis sampled by the L-shaped array is written as

$$\begin{aligned} \mathbf{X}(t) &= \mathbf{A}_x \mathbf{S}(t) + \mathbf{N}_x(t), \\ \mathbf{Y}(t) &= \mathbf{A}_y \mathbf{S}(t) + \mathbf{N}_y(t), \end{aligned} \quad (2)$$

where $\mathbf{S}(t) = [\mathbf{s}_1(t), \mathbf{s}_2(t), \dots, \mathbf{s}_n(t)]^T$ are the complex envelope vector of the incoming signals. n is the number of targets which can be detected. $\mathbf{N}_x(t)$ and $\mathbf{N}_y(t)$ are the Gaussian white noise vectors whose mean is 0 and variance is σ^2 decided by signal noise ratio (SNR). \mathbf{A}_x and \mathbf{A}_y are array manifold matrixes as

$$\begin{aligned} \mathbf{A}_x &= [\mathbf{a}_x(\phi_1, \theta_1), \dots, \mathbf{a}_x(\phi_n, \theta_n)], \\ \mathbf{A}_y &= [\mathbf{a}_y(\phi_1, \theta_1), \dots, \mathbf{a}_y(\phi_n, \theta_n)]. \end{aligned} \quad (3)$$

3. Closed-Loop Localization and Tracking System for the 3D Target

With the data received by L-shaped sensor arrays, this paper uses 2D ESPRIT and ML algorithms to get the targets' positions. Then predict the trajectories by integrating AR model into PF algorithm. Finally guide the targets' localization process with the predicted positions. The whole system forms a closed-loop feedback and improves the localization and tracking accuracy.

3.1. 2D ESPRIT Algorithm. 2D ESPRIT algorithm [11] is the expansion of ESPRIT algorithm. According to the snapshot data $\mathbf{X}(t)$, $\mathbf{Y}(t)$ received by arrays, the special covariance matrix is calculated. Then construct the signal subspace using a singular value decomposition (SVD) of the covariance matrix. With the eigenvalue derived from SVD, get $u(\phi_i, \theta_i)$ and $v(\phi_i, \theta_i)$ mentioned in (5). The elevation and azimuth angle (ϕ_i, θ_i) can be achieved by $u(\phi_i, \theta_i)$ and $v(\phi_i, \theta_i)$.

The 2D ESPRIT algorithm can be described as follows.

Step 1. Decompose $\mathbf{X}(t)$ and $\mathbf{Y}(t)$ into 4 an same size of the submatrixes $\mathbf{X}_1(t)$, $\mathbf{X}_2(t)$, $\mathbf{Y}_1(t)$, $\mathbf{Y}_2(t)$.

$$\begin{aligned} \mathbf{X}_1(t) &= \mathbf{X}(1 : M - 1, :) (t), & \mathbf{X}_2(t) &= \mathbf{X}(2 : M, :) (t), \\ \mathbf{Y}_1(t) &= \mathbf{Y}(1 : M - 1, :) (t), & \mathbf{Y}_2(t) &= \mathbf{Y}(2 : M, :) (t). \end{aligned} \quad (4)$$

Define $\mathbf{A}_{x1} = \mathbf{A}_x(1, M - 1, :)$, $\mathbf{A}_{x2} = \mathbf{A}_x(2, M, :)$, $\mathbf{A}_{y1} = \mathbf{A}_y(1, M - 1, :)$, $\mathbf{A}_{y2} = \mathbf{A}_y(2, M, :)$, so

$$\begin{aligned} \mathbf{A}_{x2} &= \mathbf{A}_{x1} \mathbf{\Phi}_x, & \mathbf{A}_{y2} &= \mathbf{A}_{y1} \mathbf{\Phi}_y, \\ \mathbf{X}_1(t) &= \mathbf{A}_{x1} \mathbf{X}(t), & \mathbf{X}_2(t) &= \mathbf{A}_{x2} \mathbf{X}(t), \\ \mathbf{Y}_1(t) &= \mathbf{A}_{y1} \mathbf{Y}(t), & \mathbf{Y}_2(t) &= \mathbf{A}_{y2} \mathbf{Y}(t), \end{aligned} \quad (5)$$

where $\mathbf{\Phi}_x$ and $\mathbf{\Phi}_y$ imply the translation characterization of the sensor array. They are denoted as

$$\begin{aligned} \mathbf{\Phi}_x &= \text{diag}(u(\phi_1, \theta_1), \dots, u(\phi_n, \theta_n)), \\ \mathbf{\Phi}_y &= \text{diag}(v(\phi_1, \theta_1), \dots, v(\phi_n, \theta_n)). \end{aligned} \quad (6)$$

Step 2. Calculate the covariance matrixes of these four submatrixes

$$\begin{aligned} \mathbf{R}_{\mathbf{X}_1 \mathbf{Y}_1} &= E[\mathbf{X}_1(t) \mathbf{Y}_1(t)^H] = \mathbf{A}_{x1} \mathbf{R}_s \mathbf{A}_{y1}^H + \mathbf{F}_{\mathbf{N}_x, \mathbf{N}_y}, \\ \mathbf{R}_{\mathbf{X}_2 \mathbf{Y}_1} &= E[\mathbf{X}_2(t) \mathbf{Y}_1(t)^H] = \mathbf{A}_{x1} \mathbf{\Phi}_x \mathbf{R}_s \mathbf{A}_{y1}^H + \mathbf{F}_{\mathbf{N}_x, \mathbf{N}_y}, \\ \mathbf{R}_{\mathbf{X}_2 \mathbf{Y}_2} &= E[\mathbf{X}_2(t) \mathbf{Y}_2(t)^H] = \mathbf{A}_{x1} \mathbf{\Phi}_x \mathbf{R}_s \mathbf{\Phi}_y^H \mathbf{A}_{y1}^H + \mathbf{F}_{\mathbf{N}_x, \mathbf{N}_y}, \end{aligned} \quad (7)$$

where $\mathbf{R}_s = E[\mathbf{S}(t) \mathbf{S}^H(t)]$ is the signal self-related matrix. \mathbf{H} is the complex conjugate transpose matrix. $\mathbf{F}_{\mathbf{N}_x, \mathbf{N}_y}$ conclude the information of the noise. Usually the noise is too less than the received signals to ignore, so we have $\mathbf{F}_{\mathbf{N}_x, \mathbf{N}_y} = 0$.

Step 3. Construct a special matrix \mathbf{C} in formula (8). And then adopt SVD for the matrix \mathbf{C} to achieve the signal subspace \mathbf{E}_s

$$\begin{aligned} \mathbf{C} &= [\mathbf{R}_{\mathbf{X}_1 \mathbf{Y}_1} \quad \mathbf{R}_{\mathbf{X}_2 \mathbf{Y}_1} \quad \mathbf{R}_{\mathbf{X}_2 \mathbf{Y}_2}], \\ [\mathbf{U}, \mathbf{S}, \mathbf{V}] &= \text{svd}(\mathbf{C}); \end{aligned} \quad (8)$$

$$\mathbf{E}_s = \begin{bmatrix} \mathbf{E}_0 \\ \mathbf{E}_1 \\ \mathbf{E}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{A} \\ \mathbf{A} \mathbf{\Phi}_x \\ \mathbf{A} \mathbf{\Phi}_x \mathbf{\Phi}_y^H \end{bmatrix} \mathbf{T} = \mathbf{U}(:, 1 : n). \quad (9)$$

There are two unknown quantities \mathbf{A} and \mathbf{T} . \mathbf{A} is the transfer matrix produced by the \mathbf{A}_{x1} and \mathbf{A}_{y1} . \mathbf{T} is a middle matrix which is invertible. \mathbf{E}_0 , \mathbf{E}_1 , \mathbf{E}_2 and \mathbf{E}_s are the signal subspace.

Step 4. Let $\mathbf{\Psi}_x = \mathbf{T}^{-1} \mathbf{\Phi}_x \mathbf{T}$, $\mathbf{\Psi}_y = \mathbf{T}^{-1} \mathbf{\Phi}_y^H \mathbf{T}$, we have $\mathbf{E}_1 = \mathbf{E}_0 \mathbf{\Psi}_x$, $\mathbf{E}_2 = \mathbf{E}_1 \mathbf{\Psi}_y$; therefore,

$$\begin{aligned} \mathbf{\Psi}_x &= (\mathbf{E}_0^H \mathbf{E}_0)^{-1} \mathbf{E}_0^H \mathbf{E}_1, \\ \mathbf{\Psi}_y &= (\mathbf{E}_1^H \mathbf{E}_1)^{-1} \mathbf{E}_1^H \mathbf{E}_2. \end{aligned} \quad (10)$$

Step 5. Use eig-function to deal with the matrixes $\mathbf{\Psi}_x$ and $\mathbf{\Psi}_y$ and gain eigenvalues which are equal to the diagonal element of the matrix $\mathbf{\Phi}_x$ and $\mathbf{\Phi}_y^H$

$$\begin{aligned} [\mathbf{V}_x, \mathbf{\Phi}_x] &= \text{eig}(\mathbf{\Psi}_x), \\ [\mathbf{V}_y, \mathbf{\Phi}_y^H] &= \text{eig}(\mathbf{\Psi}_y). \end{aligned} \quad (11)$$

Step 6. Both $\mathbf{\Phi}_x$ and $\mathbf{\Phi}_y^H$ have the information of the DOA. Due to the respective calculation, the eigenvalues from the same targets are not pairing. It needs post-process. Define a new matrix $\mathbf{G} = \mathbf{V}_y^H \mathbf{V}_x$. Adjust the correspondence of the $\mathbf{\Phi}_x$, $\mathbf{\Phi}_y^H$ according to the maximum value in each line in \mathbf{G} .

Step 7. Compute the DOA of targets

$$\begin{aligned} \theta_k &= \arctan\left(\frac{\text{angel}(v(\phi_k, \theta_k))}{\text{angel}(u(\phi_k, \theta_k))}\right), \\ \phi_k &= \arccos\left(\frac{\lambda \sqrt{v^2(\phi_k, \theta_k) + u^2(\phi_k, \theta_k)}}{2\pi d}\right). \end{aligned} \quad (12)$$

3.2. Localization through the Maximum Likelihood Algorithm.

Using the estimated DOA and the state of the sensor arrays, a maximum likelihood algorithm is proposed to realize the triangulation location. As shown in Figure 2(a), there are three uniform L-shaped arrays which have the same character. The positions of sensor arrays are $(0, 0, 0)$, $(a, b, 0)$, and $(c, d, 0)$ and meet the conditions $a \neq c \neq 0$ and $b \neq d \neq 0$.

The problem of triangulation location can be transformed into the one of seeking the intersection S for the three known radials. It has one unique answer theoretically. But due to the error of DOA estimation, the three radials will not intersect (seen in Figure 2(b)). So the paper uses a novel ML algorithm to search for partial areas and get the nearest point far from the three radials.

Definition 1. The angle between the point and radial is the one no more than 180° which is intersected by the radial and the line of the point and radial's jumping-off point.

Theorem 2. In 2D plane, the point nearest to the three radials lies in the intersected triangle by the radials.

For example, the angle between the point S and the radial L_2P_3 is $\angle SL_2P_3$. The nearest point S to three radials (L_2P_3 , L_1P_3 and L_3P_1) lies in the triangle $P_1P_2P_3$.

This theorem can be deduced to a 3D space: in 3D space, the point nearest to three radials lies in the intersection space of projection triangle formed by the three radials on the Oxy , Oxz , and Oyz planes.

The maximum likelihood algorithm is described as follows
Firstly, get the projection of the three radials on the Oxy , Oxz , and Oyz planes. Then obtain the intersection points of the three radials on each plane. After that achieve the value ranges on each axis. Take the Oyz plane as an example, the points of intersected triangle are $P_1(y_1, z_1)$, $P_2(y_2, z_2)$, and $P_3(y_3, z_3)$,

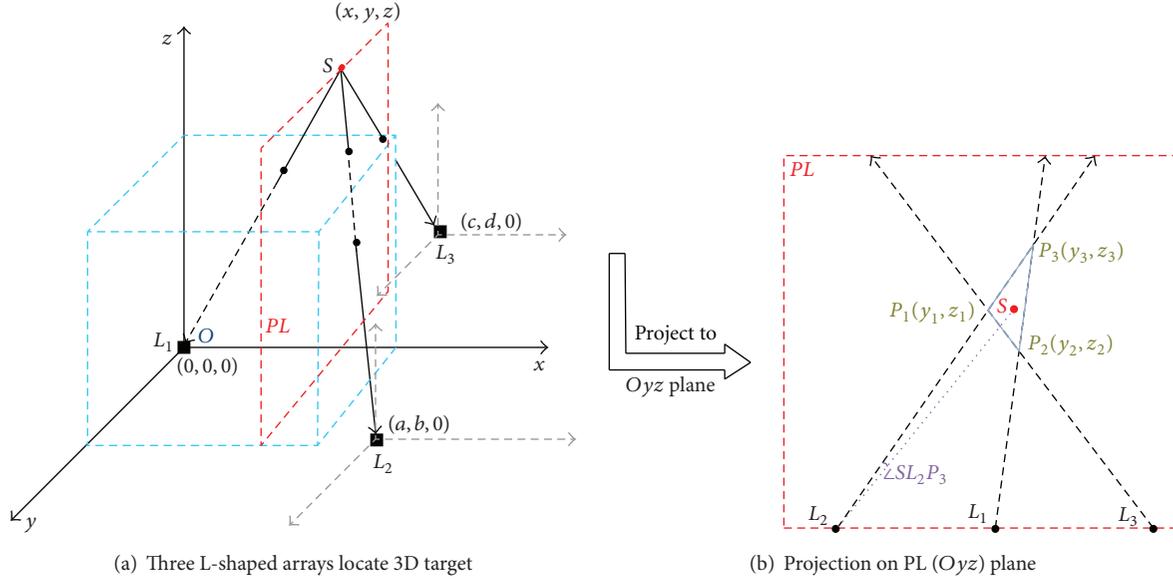


FIGURE 2: Localization by three uniform L-shaped sensor arrays.

and the range value of y is $Y_{Oyz} \in [\min(y_1, y_2, y_3), \max(y_1, y_2, y_3)]$. We can also get the range of z on the Oyz plane.

Secondly, intersect the value range of xyz axis on each plane to create a spatial space for the maximum likelihood points. Take y -axis for example, the intersection of y is $[\max(\min(Y_{Oxy}), \min(Y_{Oyz})), \min(\max(Y_{Oxy}), \max(Y_{Oyz}))]$. Then divide this 3D space into $N_X \times N_Y \times N_Z$ pieces by a self-adaptive interval. Suppose that the probability of the angle of point and radial accords to the normal distribution. For all the divided points, then calculate the total probability of the angle to the three radials

$$P(x, y, z) = \frac{1}{2\pi} \exp\left(-\frac{\theta_1^2}{\pi^2}\right) + \frac{1}{2\pi} \exp\left(-\frac{\theta_2^2}{\pi^2}\right) + \frac{1}{2\pi} \exp\left(-\frac{\theta_3^2}{\pi^2}\right), \quad (13)$$

where θ_i is the angle between the point (x, y, z) and the i th radial.

Finally, search for the maximum of $P(x, y, z)$ and find the target position (x, y, z) .

3.3. Autoregressive Particle Filter Algorithm. After localization, this paper introduces an autoregressive particle filter (ARPF) which combines AR models [13] with the particle filter [14] to predict the state of the targets in the next moment. This algorithm conquers the difficulty of designing some unknown state equations, noise matrixes, and initial state in particle filter and also provides a method to improve the accuracy of AR model.

The p -rank AR model (AR(p) in short) used for prediction can be denoted as

$$y_t = \epsilon_t - \phi_1 y_{t-1} - \dots - \phi_p y_{t-p}, \quad t = p + 1, p + 2, \dots, T, \quad (14)$$

where y_t is the predicted position, y_{t-1}, \dots, y_{t-p} is the historical state of the same target. $\phi_1, \phi_2, \dots, \phi_p$ is the autoregressive model coefficient and ϵ_t is the Gaussian white noise with the mean 0 and the variance ξ^2 .

The ARPF algorithm is carried out. (1) Form the initial particles. (2) Use the AR model to calculate the positions of the particles at the next moment instead of the motion equation. (3) Get the prediction by the weighted sum of each particle. (4) Update the probability of each particle and do importance resampling process to form the new particle cluster. (5) Skip to (2) to recur until target is out of the detecting area.

3.4. Localization and Tracking System. The localization and tracking system presented here contains the whole process of signal collection (model building), DOA estimation, and tracking and prediction. The pseudocode of the system is described in Algorithm 1.

This closed-loop system makes the localization and tracking algorithms described into an integral whole to form a positive feedback. The loop increases the robustness and the accuracy of the target localization and tracking.

4. Simulation

The paper chooses the American F22 battle plane as the emulation target. The parameters of F22 can be achieved in <http://www.aerospaceweb.org/aircraft/fighter/f22/>. Create

TABLE 1: DOA estimation with 2D ESPRIT algorithm.

		Array L_1	Array L_2	Array L_3
Real angle/°	F22-1	(9.25, -5.84)	(39.99, 139.28)	(15.07, -67.31)
	F22-2	(5.41, 2.46)	(12.85, 23.66)	(8.67, -19.22)
Estimated angle/°		(5.14, 2.49)	(40.02, 139.28)	(14.99, 112.65)
		(9.69, 174.12)	(12.63, 23.69)	(8.58, 160.79)

```

While (the target is in the detecting area of L-shaped sensor arrays)
  If (it's the first moment the arrays detect)
    (1) Use all the sensor arrays to sample the signal of the target.
    (2) Calculate the position of the target by maximum likelihood algorithm.
    (3) Select three sensor arrays which are nearest to the targets.
  Else
    (1) Use the selected three sensor arrays to sample the signal.
    (2) Get the location by ML algorithm.
  End
  If (the number of the measured position for the target <4)
    The value achieved by ML algorithm is regarded as the final position.
  Else
    (1) Use ARPF algorithm to predict the position of the target at the next moment.
    (2) If (the number >4)
      Use the predicted position to guide targets' localization process which is discussed in Section 4.
    End
  End
  Time++;
End

```

ALGORITHM 1

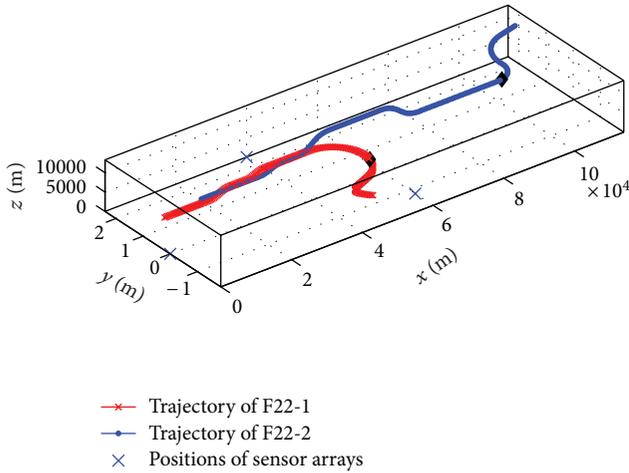


FIGURE 3: Three L-shaped arrays sensing 3D maneuvering targets.

the random trajectory of F22 by the navigation and positioning toolbox of GPSof in MATLAB, as shown in Figure 3. There are 2 pieces of moving orbits. The time range when F22-1 is flying is [195, 399] s, and the range of F22-2 is [171, 397] s. These planes emit sine complex envelope signals omnidirectionally when it moves along the trail.

We choose three uniform L-shaped sensor arrays of the same nature to localize and track 3D targets. The

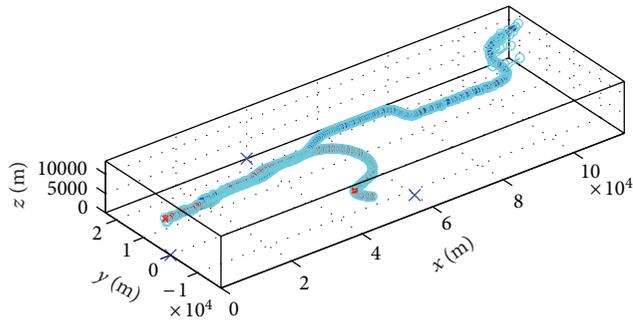
fixtures of the arrays are (0,0,0) m, (60000, -12000, 0) m, and (40000, 24000, 0) m. Set the parameters of the sensor array mentioned in Section 2, $M = 8$, $dx = dy = 0.5$ m, $\lambda = 1$ m, and $SNR = 20$ dB. The number of the snapshot is 64. The frequency of the localization is 1 Hz.

The experiment adopts Windows XP SP3 as the software platform, and AMD Athlon(tm) II X2 250 3.01 GHz and 2 GB memory as the hardware platform. The programming environment is MATLAB R2010b.

4.1. DOA Estimation. At a certain time, F22-1 is located in (52266, -5342, and 8561) m, and target F22-2 is located in (96898, 4164, and 9191) m, see the rhombuses shown in Figure 3. With the signal vector $\mathbf{X}(t)$ and $\mathbf{Y}(t)$ sampled by the sensor arrays, DOA of the two targets is estimated, using the 2D ESPRIT algorithm. The results are shown in Table 1. The first item in brackets is elevation, and the second item is azimuth.

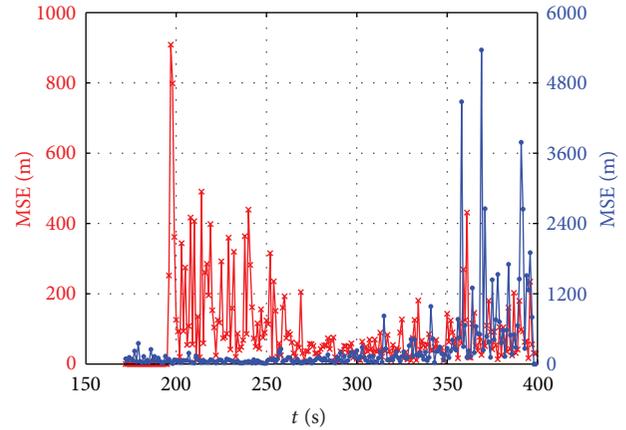
By doing tens Monte Carlo simulation and calculate the mean of DOA, it can be found that the average error of angle estimation is less than 0.2° . Most of the time the 2D ESPRIT algorithm works well, but when the incident DOA is very close to each other, the errors is a little big. The 2D ESPRIT algorithm needs to be improved in such situation.

Besides, there are other two problems: first, the DOA estimations of the same target are not in the same line.



—×— Trajectory of F22-1 × Positions of sensor arrays
 —●— Trajectory of F22-2 —○ Localization by ML algorithm

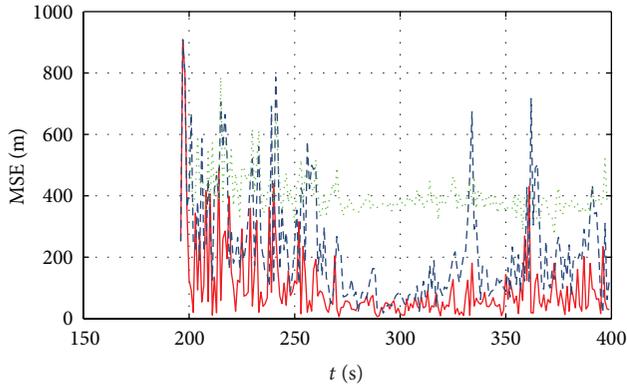
(a) Triangulation localization with the maximum likelihood algorithm



—×— Localization MSE of F22-1
 —●— Localization MSE of F22-2

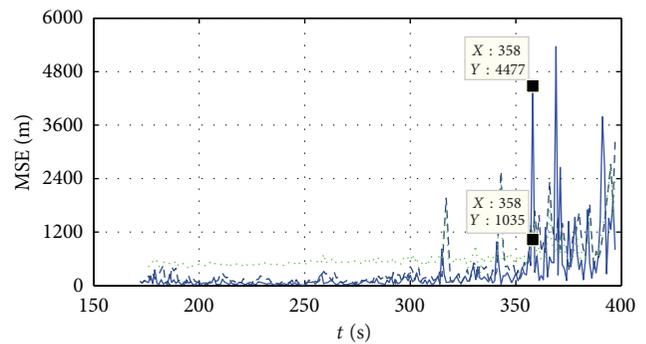
(b) MSE of localization with the ML algorithm

FIGURE 4: Performance of the maximum likelihood algorithm.



— Localization MSE of F22-1
 - - - Prediction MSE of F22-1 by ARPF
 ····· Distance between locations of F22-1
 at the current and preview moment

(a)



— Localization MSE of F22-2
 - - - Prediction MSE of F22-2 by ARPF
 ····· Distance between locations of F22-2
 at the current and preview moment

(b)

FIGURE 5: MSE of the ARPF algorithm.

Second, the estimated azimuth is all in $[0, 180]^\circ$. When the incident azimuth is minus, the result is always equal to this angle plus 180° .

So before localization by maximum likelihood algorithm, the DOA from the same target must be paired in the same group, and the azimuth must be well decided. When there is no other prior information, the positions of targets at the preview moment are used to confirm azimuth and group the DOA estimations which are out of order.

4.2. Triangulation Location. With the estimated DOA and the location of sensor arrays, the maximum likelihood algorithm is employed to calculate the positions of the targets. The localization results and errors are displayed in Figure 4.

We can find in Figure 4 that most localization mean square errors (MSEs) of F22-1 are less than 300 m, and F22-2 localization deviation is less than 1500 m by the maximum likelihood algorithm. Because of the larger distance between F22-2 and the sensor arrays, MSE of F22-2 is bigger than MSE of F22-1.

For far-field targets' localization in this application, the relative errors are less than 3% which is acceptable. So the ML method meets the localization needs. There are also some outliers that have exceptional large localization MSE. These outliers always appear when the target is far away from sensor arrays. So when there are redundant sensor arrays, select three sensor arrays which are nearest to the target for localization.

TABLE 2: Performance of closed positive feedback loop.

	F22-1 MSE/m	F22-2 MSE/m	Average consuming/s
Localization by 2D ESPRIT and ML algorithm	98.55	290.38	0.033
Using a closed-loop system	96.97	276.14	0.047

4.3. *Trajectory Prediction.* After localization by the 2D ESPRIT and ML algorithms, autoregressive particle filter is implemented to form a positive feedback closed loop which enhances the accuracy and stability of the localization. The forecast MSE is seen in Figure 5.

Compare prediction MSE by ARPF and the distance between the location at current and preview moment. It can be found that most of the time, the prediction MSE by ARPF is smaller. This illustrates that the prediction is more appropriate to direct the process mentioned above which confirms azimuth and groups the DOA estimations. There are also some moments that the prediction MSE is less than the localization MSE, so when the localization result is far away from the trajectory, in other words, when the outlier appears, we would better use the prediction as the true location instead of the measurement. For example, the location MSE of F22 by ML algorithm at 358 s is 4477 m, and the forecast MSE is 1035 m; this time uses the predicted position as the real measurement.

The performance of the closed loop is shown in Table 2.

Table 2 illustrates that 2D ESPRIT and ML algorithms are available to locate the target because the localization MSE is acceptable. The close loop system can slightly improve the accuracy of the localization. The time consumption is a little increased, but still fast enough to locate and track the target.

5. Conclusion

This paper proposes a localizing and tracking system for a three-dimensional target which combines the DOA estimation, the triangulation location, and the trajectory prediction. Firstly we establish a signal-reception model for a far-field target. Then the 2D ESPRIT algorithm is carried out to estimate the DOA of the 3D target. After that, a novel maximum likelihood algorithm is introduced to solve the triangulation location problem. Next the autoregressive particle filter is implemented to predict the target trajectory. Finally the localization process is guided according to the predictions. The experiment shows that the localization by 2D ESPRIT algorithm and the ML algorithm are useful for the 3D target. ARPF algorithm can well predict position at the next moment and guide the localization process. The localization and prediction processes form a passive feedback loop which can raise the accuracy and robustness of the localization and tracking system.

Acknowledgments

The authors would like to acknowledge the support from the National Grant Fundamental Research 973 Program

of China (Grant no. 2006CB303000), the National Natural Science Foundation of China (Grants no. 60673176, 60373014, 50175056, and 60970103), and the Ph.D. Programs Foundation of the Ministry of Education of China (Grant no. 20090002110016).

References

- [1] N. Abdelkrim, N. Aouf, A. Tsourdos, and B. White, "Robust nonlinear filtering for INS/GPS UAV localization," in *Proceedings of the 16th Mediterranean Conference on Control and Automation (MED '08)*, pp. 695–702, Ajaccio, France, June 2008.
- [2] S. Sohn, B. Lee, J. Kim, and C. Kee, "Vision-based real-time target localization for single-antenna GPS-guided UAV," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 44, no. 4, pp. 1391–1401, 2008.
- [3] C. Liu, X. Wang, and Z. Xing, "Fast face detection method based on real-time prediction and learning classification," *Chinese Journal of Scientific Instrument*, vol. 33, no. 1, pp. 36–41, 2012.
- [4] A. Nemra and N. Aouf, "Adaptive decentralised cooperative vision based simultaneous localization and mapping for multiple UAV," in *Proceedings of the 19th Mediterranean Conference on Control and Automation*, pp. 546–551, Corfu, Greece, 2011.
- [5] A. S. Paul and E. A. Wan, "RSSI-based indoor localization and tracking using sigma-point Kalman smoothers," *IEEE Journal of Selected Topics in Signal Processing*, vol. 3, no. 5, pp. 860–873, 2009.
- [6] S. Wang, X. Wang, Y. Wang, and X. Sun, "Distributed lightweight target tracking for wireless sensor networks," in *Proceedings of the 6th IEEE International Conference on Mobile Adhoc and Sensor Systems (MASS '09)*, pp. 870–875, Macau, China, October 2009.
- [7] A. Hurst, A. Wickenheiser, and E. Garcia, "Localization and perching maneuver tracking for a morphing UAV," in *Proceedings of the IEEE/ION Position, Location and Navigation Symposium (PLANS '08)*, pp. 1238–1245, Monterey, Calif, USA, May 2008.
- [8] X. Zhang, X. Wang et al., "Sensor arrays based predicted spatial multi-signals classification method for target localization and tracking," *Chinese Journal of Scientific Instrument*, vol. 33, no. 5, pp. 970–975, 2012.
- [9] W. Tidd, R. J. Weber, Y. Huang, and Y. Zhao, "A RF source localization and tracking system," in *Proceedings of the IEEE Military Communications Conference (MILCOM '10)*, pp. 858–863, San Jose, Calif, USA, November 2010.
- [10] M. F. Fallon and S. Godsill, "Acoustic source localization and tracking using track before detect," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 18, no. 6, pp. 1228–1242, 2010.
- [11] F. Talantzis, "An acoustic source localization and tracking framework using particle filtering and information theory,"

IEEE Transactions on Audio, Speech and Language Processing, vol. 18, no. 7, pp. 1806–1817, 2010.

- [12] L. M. Lin, *Study on Estimation of Two Dimensional Direction of Arrival Using a DBF Receiver*, Harbin Institute of Technology, Harbin, China, 2008.
- [13] W. Liu and M. Farooq, “ARMA model based scheme for maneuvering target tracking,” in *Proceedings of the 37th Midwest Symposium on Circuits and Systems*, pp. 1408–1411, August 1994.
- [14] R. van der Merwe and A. Doucet, “The unscented particle filter,” Tech. Rep. CUED/F-INFENG/TR 380, 2000.

