

Research Article

Optimal Maneuver Strategy of Observer for Bearing-Only Tracking in Threat Environment

Renke He , Shuxin Chen, Hao Wu , Zhuowei Liu, and Jianhua Chen

Information and Navigation College, Air Force Engineering University, Xi'an 710077, China

Correspondence should be addressed to Hao Wu; wuhaostudy@163.com

Received 19 October 2017; Accepted 5 June 2018; Published 18 July 2018

Academic Editor: Mauro Pontani

Copyright © 2018 Renke He et al. 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 optimal maneuver of observer for bearing-only tracking (BOT) in a threat environment is a complex problem which involves nonlinear filtering, threat avoidance, and optimal maneuver strategy. Under comprehensive consideration, the reward function comprised of the lower bound on detFIM and threat cost was established; the finite-horizon MDP principle was applied to obtain the optimal strategy. The quantization method was used to discretize the BOT process and calculate the transition matrix of Markov chain; to achieve quantization in the beginning of each period, CKF was applied to provide the initial state estimate and the corresponding error covariance. The numerical simulations illustrated the applicability and superior performance for static and dynamic target tracking in several scenarios in the threat environment.

1. Introduction

Bearing-only tracking (BOT) techniques are applied in various scenarios of target location, and the related theoretical and practical problems have been studied for decades [1–3]. The accuracy and efficiency of the target location have witnessed significant improvement in recent years due to the improvement of advanced filtering techniques [4–6] and UAV (unmanned aerial vehicle) platform. Many researchers focus on location problems with static target or tracking problems under conditions of observer's specific trajectories, while the target could be located more accurately if the observer implements maneuverings based on certain rules [7]. In a military field, the observer's (aircraft, UAV, car, warship, submarine, etc.) maneuvering trajectories may be constrained by some threats (such as missile, torpedo, no fly zone, and air defense radar), and the optimal observer's trajectories for BOT could fail to satisfy those constraints; in result, the observer's maneuver need to not only ensure the achievement of bearing information but to also ensure the safety of the observer itself. This

problem concerns the balance between accuracy of target tracking and threat avoidance.

For bearing-only tracking problem, the Cramer-Rao lower bound (CRLB) and Fisher information matrix (FIM) are usually used to evaluate the performance of target tracking and observer's maneuvering. Fawcett [1] used the CRLB to evaluate the effect of course maneuvers on bearing-only range, but the optimization problem of the observer's maneuver was not considered. In [8], the FIM was used to achieve the optimal leg of the observer, and the EKF algorithm was updated by heuristic evolutionary optimization algorithms to enhance the accuracy of the BOT, while heuristic evolutionary optimization algorithms cannot ensure the efficiency of calculation. In [9], recursive Bayesian estimation methods are used to several angle-only applications in sea-to-sea, air-to-sea, and sea-to-sea scenarios; the particle filter (PF) and range parameterized EKF were used for comparison. In [10, 11], Kalman filter (KF), EKF, and PF have been applied in bearing and elevation measurements for real-time object tracking in underwater environment, while they did not consider the optimal trajectories of the observer. In

[12], the observability was analyzed when the observer maneuvers smoothly, and the necessary and sufficient conditions of observability were established. In [13], stochastic control was applied for underwater optimal trajectories in BOT, and dynamic programming was used to achieve the optimal control sequence. Zhang et al. [14, 15] applied Markov decision processes (MDP) and stochastic optimal control for optimization problem of the observer's trajectories. In [15], the lower bound of the determinant of the FIM matrix was used as reward function, while the parameters evaluating distance from the observer to target were ignored due to its inaccuracy. The observer's trajectories in all those researches do not consider the constraints of the threat environment, while obstacle/threat avoidance is the pure control or decision methodology applied for an autonomous flight of UAV [16, 17], which has little connection with BOT problem. Thus, it remains difficult to ensure that the observer's maneuvering trajectories satisfy the requirements of BOT and the threat avoidance at the same time.

In this paper, the model of threat environment was established, and the cost of threat avoidance was combined with the detFIM in reward function. Based on the initial state estimate and the corresponding error covariance provided by the cubature Kalman filter (CKF) [18], the quantization method [19] was applied to discretize the whole process. Finally, the optimal maneuvering strategy can be calculated in each step by the finite-horizon MDP approach based on the reward function. The paper is organized as follows. Section 2 defines the BOT problem, Section 3 introduces the quantization method and CKF, Section 4 introduces the establishment of reward function, Section 5 presents the finite-horizon MDP approach, Section 6 briefly outlines the algorithm, Section 7 presents the numerical simulations, and Section 7 summarizes the work.

2. Problem Definition

Suppose that the observer is an UAV platform, the target is a car moving with constant velocity, and the height from the observer to the target is known. Considering the static threats in the environment, the threats must be avoided by the UAV to ensure the safety of the flight. The positions are set in 2-dimensional Cartesian coordinate; the state of target is $\mathbf{x}_k^t = [x_k^t, y_k^t, \dot{x}_k^t, \dot{y}_k^t]^T$, where the position and velocity are $\mathbf{x}_k^p = [x_k^t, y_k^t]^T$ and $\mathbf{x}_k^v = [\dot{x}_k^t, \dot{y}_k^t]^T$, respectively. The state of the observer is $\mathbf{x}_k^s = [x_k^s, y_k^s, \dot{x}_k^s, \dot{y}_k^s]^T$; the related state $\mathbf{x}_k = \mathbf{x}_k^t - \mathbf{x}_k^o = [x_k, y_k, \dot{x}_k, \dot{y}_k]^T$, where $k = 1, 2, \dots, n$. The model of BOT can be described as follows:

$$\mathbf{x}_k = \mathbf{F}\mathbf{x}_{k-1} - \mathbf{u}_{k-1,k} + \mathbf{v}_{k-1}, \quad (1)$$

$$z_k = h(\mathbf{x}_k) = \arctan \frac{x_k}{y_k} + e_k. \quad (2)$$

Equations (1) and (2) are state function and measurement function, respectively, where $\mathbf{x}_k \in \mathbb{R}^{n_x}$, $z_k \in \mathbb{R}^{n_z}$, and n_x, n_z are the dimension. \mathbf{v}_{k-1} is a zero-mean Gaussian noise with covariance matrix \mathbf{Q} , \mathbf{F} is the state transition matrix, and $\mathbf{u}_{k-1,k}$ is the input.

$$\mathbf{Q} = \begin{bmatrix} \frac{\Delta t^3}{3} & 0 & \frac{\Delta t^2}{2} & 0 \\ 0 & \frac{\Delta t^3}{3} & 0 & \frac{\Delta t^2}{2} \\ \frac{\Delta t^2}{2} & 0 & \Delta t & 0 \\ 0 & \frac{\Delta t^2}{2} & 0 & \Delta t \end{bmatrix} \cdot \mathbf{q},$$

$$\mathbf{F} = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$\mathbf{u}_{k-1,k} = \begin{bmatrix} x_k^s - x_{k-1}^s - \Delta \dot{x}_{k-1}^s \\ y_k^s - y_{k-1}^s - \Delta \dot{y}_{k-1}^s \\ \dot{x}_k^s - \dot{x}_{k-1}^s \\ \dot{y}_k^s - \dot{y}_{k-1}^s \end{bmatrix}, \quad (3)$$

where Δt is the time interval of measurements, \mathbf{q} is the intensity of process noise, and e_k is a zero-mean Gaussian noise with covariance matrix \mathbf{R}^z .

3. Discretization of the Process

In order to meet the requirements of the discrete time MDP approach, a quantization method is applied to approximate the continuous process of BOT. Actually, the quantization is used to approximate the process \mathbf{x}_k by a finite Markov chain, which is defined in the quantization algorithm [19, 20]. For each time, $(\mathbf{x}_k)_{0 \leq k \leq N}$ is divided by grid Γ_k , which is common to all these approximation methods.

$$\Pi_{z,t,f} = E\{f(\mathbf{x}_t) | z_t\} \approx \sum_{i=1}^N \mathbf{P}_{t,i} f(\mathbf{x}_t^i). \quad (4)$$

In the marginal quantization filtering, the grid points $\{\mathbf{x}_t^i\}_{i=1}^N \subset \mathbb{R}^{n_x}$ and probabilities $\{\mathbf{P}_{t,i}\}_{i=1}^N$ are generated by the Monte Carlo approach. These grids can define a new state by

$$\hat{\mathbf{x}}_k^\Gamma := \text{Proj}_{\Gamma_k}(\mathbf{x}_k), \quad \text{for } k = 0, 1, \dots, N. \quad (5)$$

There is one-to-one match between the nearest-neighbor projection Proj_{Γ_k} and Voronoi tessellation of Γ , which means Borel partitions $C_1(\Gamma_k), \dots, C_N(\Gamma_k)$ of \mathbb{R}^{n_x} satisfy

$$C_i(\Gamma_k) \subset \left\{ \eta \in \mathbb{R}^{n_x} : |\eta - x^i| = \min_{x^j \in \Gamma} |\eta - x^j| \right\}, \quad i = 1, \dots, N. \quad (6)$$

The updating rule of grids Γ_k is the competitive learning vector quantization (CLVQ) method. For every time of $s \in M$, M is the number of Monte Carlo; select $\mathbf{x}_{k,\text{Neig}}$ as the closest neighbor of \mathbf{x}_k in Γ_k^s ; in the learning phase, set

$$\mathbf{x}_{k,\text{Neig}}' = \mathbf{x}_{k,\text{Neig}} - \gamma_k (\mathbf{x}_{k,\text{Neig}} - \mathbf{x}_k), \quad (7)$$

where γ_k is the updating rate. The CLVQ has been widely adopted for the neuron updating in the neural network. The details of the algorithm can be seen in [19, 20]. At each step from Γ_k^i to Γ_{k+1}^j , the transition matrix of the Markov chain is

$$\begin{aligned} \hat{\mathbf{P}}_{i,j}^k &= \mathbf{P}(\hat{\mathbf{x}}_{k+1} = j | \hat{\mathbf{x}}_k = i) = \frac{\mathbf{P}(\hat{\mathbf{x}}_k = i, \hat{\mathbf{x}}_{k+1} = j)}{\mathbf{P}(\hat{\mathbf{x}}_k = i)} \\ &= \frac{\mathbf{P}(\mathbf{x}_k \in C_i(\Gamma_k), \mathbf{x}_{k+1} \in C_j(\Gamma_{k+1}))}{\mathbf{P}(\mathbf{x}_k \in C_i(\Gamma_k))}. \end{aligned} \quad (8)$$

Thus, the process $(\mathbf{x}_k)_{0 \leq k \leq N}$ is replaced by the Markov chain $(\hat{\mathbf{x}}_k)_{0 \leq k \leq N}$ with the transition matrix $(\hat{\mathbf{P}}_{i,j}^k)_{0 \leq k \leq N-1}$ at each step. The whole process is divided in several periods; the initial distribution of each period is also quantized by $\mathbf{P}(\mathbf{x}_0 \in C_i(\Gamma_0))$, whose initial values are given randomly. So, the CKF is used to estimate the actual position of the target and provide the density $(\mathbf{x}_{k|k}, \mathbf{P}_{k|k})$ as the initial distribution in each period of quantization.

Compared with EKF, the accuracy of CKF is much higher when they are applied for nonlinear filtering problem. The brief steps of CKF are as follows:

(i) Time update

$$\begin{aligned} \hat{\mathbf{x}}_{k|k-1} &= \mathbf{F}\hat{\mathbf{x}}_{k-1} - \mathbf{u}_{k-1,k}, \\ \mathbf{P}_{k|k-1} &= \mathbf{F}\mathbf{P}_{k-1}\mathbf{F}^T + \mathbf{Q}. \end{aligned} \quad (9)$$

(ii) Measurement update

(iii) Factorize

$$\mathbf{P}_{k|k-1} = \mathbf{S}_{k|k-1}\mathbf{S}_{k|k-1}^T. \quad (10)$$

(iv) Evaluate the cubature points

$$\mathbf{X}_{i,k|k-1} = \mathbf{S}_{k|k-1}\xi_i + \hat{\mathbf{x}}_{k|k-1}. \quad (11)$$

(v) Evaluate the propagated cubature points

$$\mathbf{Z}_{i,k|k-1} = \mathbf{h}(\mathbf{X}_{i,k|k-1}, \mathbf{u}_k). \quad (12)$$

(vi) Estimate the predicted measurement

$$\hat{\mathbf{z}}_{i,k|k-1} = \frac{1}{m} \sum_{i=1}^m \mathbf{Z}_{i,k|k-1}. \quad (13)$$

(vii) Estimate the innovation covariance matrix

$$\mathbf{P}_{zz,k|k-1} = \frac{1}{m} \sum_{i=1}^m \mathbf{Z}_{i,k|k-1} \mathbf{Z}_{i,k|k-1}^T - \hat{\mathbf{z}}_{k|k-1} \hat{\mathbf{z}}_{k|k-1}^T + \mathbf{R}_k. \quad (14)$$

(viii) Estimate the cross-covariance matrix

$$\mathbf{P}_{xz,k|k-1} = \frac{1}{m} \sum_{i=1}^m \mathbf{X}_{i,k|k-1} \mathbf{Z}_{i,k|k-1}^T - \hat{\mathbf{x}}_{k|k-1} \hat{\mathbf{z}}_{k|k-1}^T. \quad (15)$$

(ix) Estimate the Kalman gain

$$\mathbf{W}_k = \mathbf{P}_{xz,k|k-1} \mathbf{P}_{zz,k|k-1}^{-1}. \quad (16)$$

(x) Estimate the updated state

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{W}_k (\mathbf{z}_k - \hat{\mathbf{z}}_{k|k-1}). \quad (17)$$

(xi) Estimate the corresponding error covariance

$$\mathbf{P}_{k|k} = \mathbf{P}_{k|k-1} - \mathbf{W}_k \mathbf{P}_{zz,k|k-1} \mathbf{W}_k^T. \quad (18)$$

4. Reward Function

It is important to note that the reward function is the key to solve the problem with BOT and threat avoidance, namely, it consists of two parts, one of which describes the profits of bearing information from target, and the other is the cost of threat avoidance. For the former part, the detFIM performance indexed by a lower bound on detFIM is commonly used:

$$J_{\text{FIM}} = \frac{1}{\sigma^2} \int_0^T \frac{\dot{\theta}}{r^2} d\tau, \quad (19)$$

where σ is the standard deviation of the bearing measurement error, T is the time horizon, $\dot{\theta}$ is the bearing rate, and r is the relative distance from the observer to a static threat. This function is convenient to evaluate.

The second part is the cost of threat avoidance. The intensity of threats is determined by the relative distance from the observer to different threats, which means the smaller the relative distance, the greater the intensity. A potential field can be used to establish the threat environment.

In the two-dimensional Cartesian coordinate, the coordinate of the observer is (x_k^s, y_k^s) , the coordinate of a threat is (x_k^q, y_k^q) , and the potential at the coordinate of the observer at time t is

$$V_p(t) = \sum_{i=1}^l \frac{1}{\sqrt{(x^s(t) - x^q_i)^2 + (y^s(t) - y^q_i)^2}}, \quad l = 1, 2, \dots, L. \quad (20)$$

In time horizon T , the cost of the threats is

$$J_{\text{threat}} = \int_0^T V(\tau) d\tau. \quad (21)$$

The intensity of threat can only exert influences within a limited distance d_{lim} , so we define the potential of threats as

$$V_p(t) = \begin{cases} \sum_{i=1}^l \frac{1}{\sqrt{(x^s(t) - x^{q_i})^2 + (y^s(t) - y^{q_i})^2}}, & \text{if } d \leq d_{\text{lim}}, \\ 0, & \text{otherwise,} \end{cases} \quad (22)$$

where d is the relative distance of the observer and threats. Thus, the reward function can be represented by

$$J = \int_0^T \left\{ \frac{1}{\sigma^2 r^2} \dot{\theta} - \varepsilon \cdot V(\tau) \right\} d\tau, \quad (23)$$

where ε is a constant coefficient, which ensures an agreement of order of magnitude in two parts.

5. Optimal Maneuvering Strategy

The core issue of optimal strategy is the maximization of the reward function established. To achieve this purpose, the finite-horizon MDP principle can be used, and it demands that the reward function J satisfies the dynamic programming property [7]:

$$\max_{\mathbf{u}_k} J(\mathbf{x}_k) \leq \max_{\mathbf{u}_k} J(\mathbf{x}_k + F(\mathbf{x}_k, \mathbf{u}_{k+1})). \quad (24)$$

So the sequence of controls $\{\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_k\}$ must satisfy

$$\mathbf{u}_k \leftarrow \arg \max [J]. \quad (25)$$

For the model of the finite-horizon MDP,

$$(\mathbf{x}, \mathbf{A}(\mathbf{x}), \mathbf{Q}, c, J_N), \quad (26)$$

where \mathbf{x} is the state space, a Borel space; $\mathbf{A}(\mathbf{x})$ is the control or action set, a Borel space; \mathbf{Q} is the transition probability function; c is the reward function at each step; and J_N is the terminal reward the each finite time horizon.

The maneuvering strategy is $\pi = \{a_t \mid t = 0, 1, \dots, N-1\} \in \mathbf{A}$; a_t is the angle set of maneuvering; the state of the observer with the maneuver strategy a_t is defined as

$$\begin{aligned} x_{t+1}^s &= x_t^s + v_t \Delta t \cdot \cos(a_t), \\ y_{t+1}^s &= y_t^s + v_t \Delta t \cdot \sin(a_t). \end{aligned} \quad (27)$$

For the whole process, the maneuver strategy is $\pi = \{a_t \mid t = 0, 1, \dots, N-1\} \in \mathbf{A}$; the expected reward function is

$$J(\mathbf{x}, \pi) = E \left[\sum_t^{N-1} c(x_t, a_t) + J_N(\mathbf{x}_N) \right]. \quad (28)$$

The optimal whole reward function is

$$J^* = \inf_{\pi \in \mathbf{A}} J(\mathbf{x}, \pi). \quad (29)$$

For $t \in \{N-1, \dots, 2, 1\}$,

$$J_t^* = \max_{a \in \mathbf{A}} \sum_{\mathbf{x}_{t+1} \in \mathbf{X}} P_{\mathbf{x}_t, \mathbf{x}_{t+1}}^t [c(\mathbf{x}_t, a, \mathbf{x}_{t+1}) + J_{t+1}^*(\mathbf{x}_{t+1})]. \quad (30)$$

The maneuver strategy can be calculated by

$$a_t^* = \arg \max_{a \in \mathbf{A}} \sum_{\mathbf{x}_{t+1} \in \mathbf{X}} P_{\mathbf{x}_t, \mathbf{x}_{t+1}}^t [c(\mathbf{x}_t, a, \mathbf{x}_{t+1}) + J_{t+1}^*(\mathbf{x}_{t+1})]. \quad (31)$$

Based on the quantization, $\forall (i, j) \in \Gamma_k \times \Gamma_{k+1}$, $P_{\mathbf{x}_t, \mathbf{x}_{t+1}}^t$ can be replaced by $P_{i,j}^t$:

$$\begin{aligned} P_{\mathbf{x}_t, \mathbf{x}_{t+1}}^t(a) &\approx P_{i,j}^t(\mathbf{x}_{t+1}^s = h, \widehat{\mathbf{x}}_{t+1} = j \mid \mathbf{x}_t^s = g, \widehat{\mathbf{x}}_t = i, a_t = a), \\ J_t^*(h, i) &= \max_{a \in \mathbf{A}} \sum_{j \in \Gamma_{t+1}} P_{i,j}^t [c((g, i), a, (h, j)) + J_{t+1}^*(h, i)]. \end{aligned} \quad (32)$$

6. Algorithm

The algorithm is composed of two processes; firstly, a quantization method is applied to provide the discretized density and transition matrix, which is used by the finite-horizon MDP to calculate the optimal maneuvering strategy. The reward function combines the detFIM and the cost of threats. The parameters $(\Gamma_k, \mathbf{P}_{ij}^k)$ can be achieved by quantization and based on the parameters and reward function; the optimal maneuvering strategy can be output by the finite-horizon MDP. At the same time, CKF provides the density for $(\mathbf{x}_{k|k}, \mathbf{P}_{k|k})$ as initial density for each period. The diagram of the algorithm is shown in Figure 1 as below.

7. Numerical Example

To test the feasibility of the algorithm, we apply the algorithm to several scenarios, including static target tracking and moving target tracking with threat or no-threat zone, and different trajectories and filtering methods are compared. The simulation is conducted in MATLAB 7.0 using Windows XP, Intel Core i3, 3.3 GHz platform. The parameters are as follows: $q = 10^{-5} \text{ km}^2/\text{s}^3$, $\sigma = 0.5 \text{ deg}$, $\varepsilon = 0.001$, $T = 120 \text{ s}$, and $d_{\text{lim}} = 20 \text{ km}$; measurement interval $\Delta t = 10 \text{ s}$; the observer initial position (0 km, 0 km); and target initial position (100 km, 100 km). Monte Carlo simulation is implemented 100 times for each scenario.

7.1. Static Target Tracking. Two simple scenarios are given for static target tracking, one is in no-threat environment, and the other considers the problem in the threat environment. Considering the kinematic constraint of the observer, the change of relative velocity direction in adjacent time is limited in $\pm 60^\circ$. For static target tracking problem, the state transition matrix $\mathbf{F} = \mathbf{I}$.

In Figure 2, the optimal trajectory is "s" shaped; the shape can be explained by formula (19), where r and change of bearing $\Delta\theta$ are positive related to J_{FIM} . Therefore, the observer can move to the target; at the same time, the maneuvering must guarantee the change of bearing is in a limited range; the shape "s" represents the balance between the numerator and the denominator of the formula. Figure 3 shows the target tracking error in the x -axis and the y -axis of scenarios 1 and 2. Compared with Figure 2, Figure 4 shows the variation of the trajectory when the observer encounters the threats. The circle lines of various colors represent the contours of different threats' potential field. In the beginning,

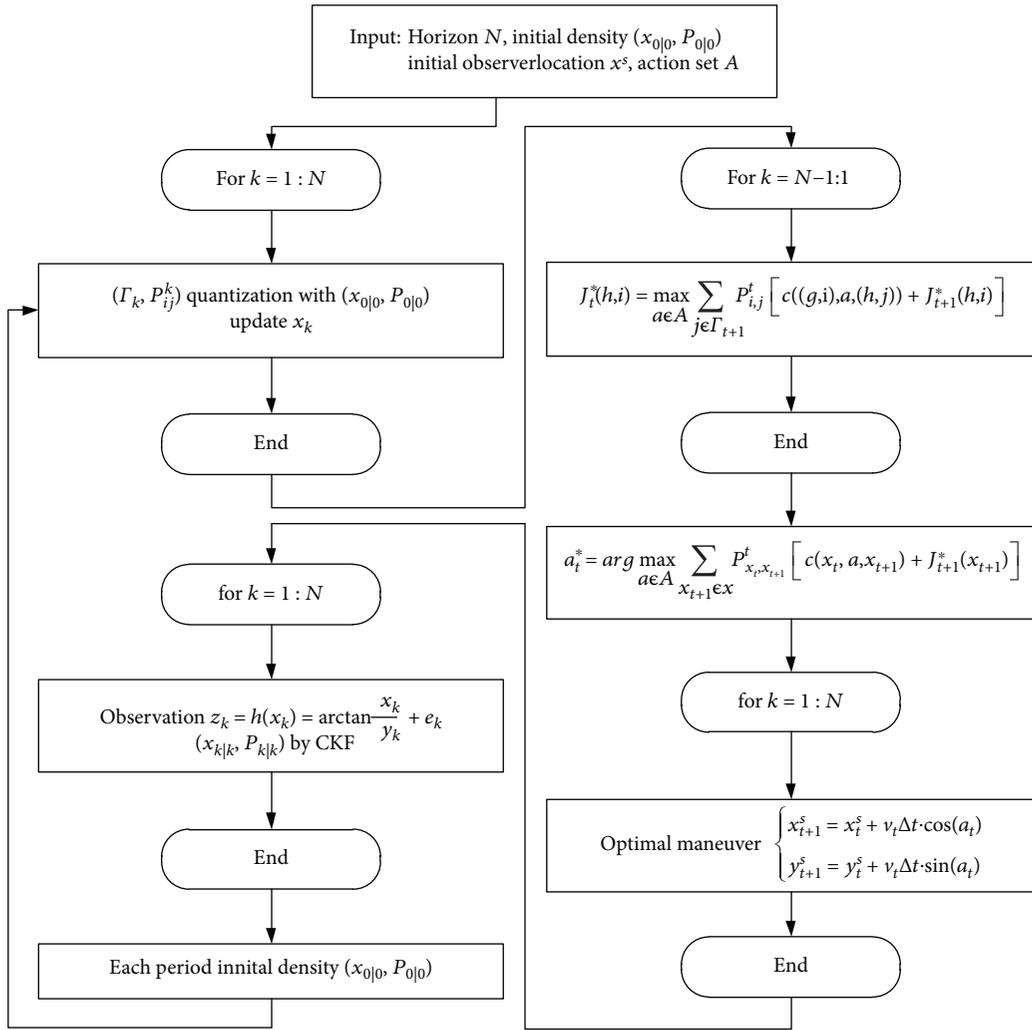
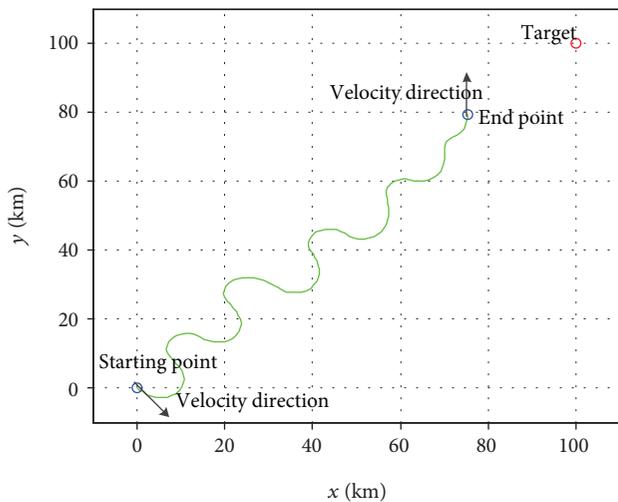
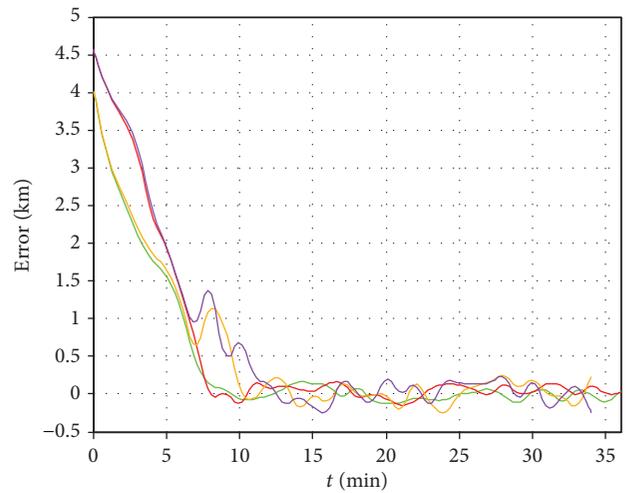


FIGURE 1: Algorithm flow diagram.



— Observer
○ Target

FIGURE 2: Static target tracking in scenario 1.



— X error in scenario 1 — X error in scenario 2
— Y error in scenario 1 — Y error in scenario 2

FIGURE 3: Target tracking results in scenario 1, 2.

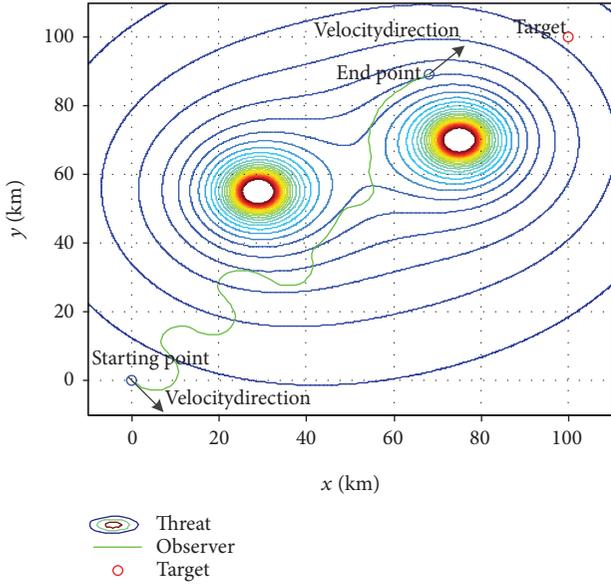


FIGURE 4: Static target tracking in scenario 2.

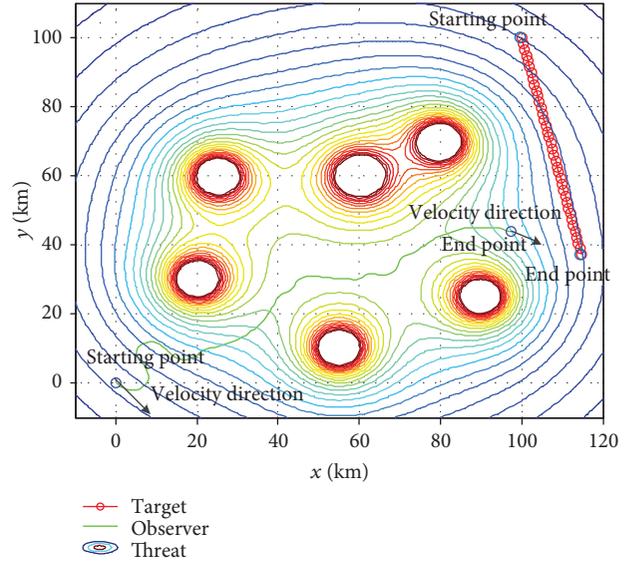


FIGURE 6: moving target tracking in scenario 4.

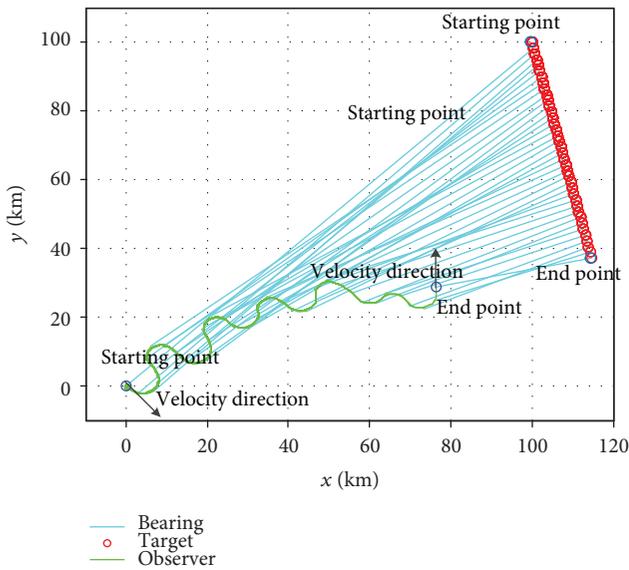


FIGURE 5: Moving target tracking in scenario 3.

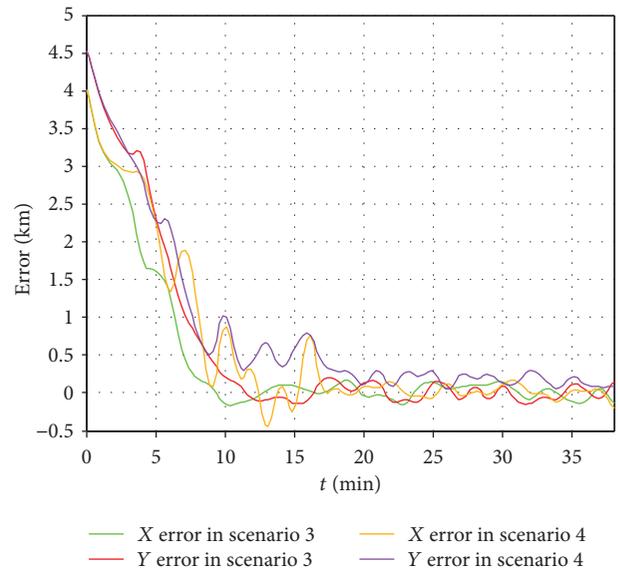


FIGURE 7: Target tracking results in scenario 3, 4.

the maneuver is “s” shaped; as the observer approach the threats, the range of feasible maneuvers markedly reduced, and the direction of velocity changes as well. The difference between Figures 2 and 4 shows the process from formulas (19), (20), (21), (22), and (23), namely, the cost of threat avoidance influences the reward of detFIM. Although the threats affect the maneuver of the observer, it still provides effective measurements.

7.2. Moving Target Tracking. The moving target tracking scenarios are also set in environments with or without threats. The motion process is shown in 2D Cartesian coordinate.

As shown in Figure 5, the green line represents the trajectory of the observer, the blue line represents the partial

direction vectors of bearings, and the red line represents the trajectory of target. The trajectory in Figure 5 is similar to the situation in Figure 2, while the trajectory in Figure 5 is tracking the moving target, so the contour of the entire trajectory is more complicated compared with the one in Figure 2 due to the change of location of the target. Without the influence of threats, the trajectory keeps the “s” shape. Scenario 4 considering multithreat environment is shown above. The trajectory in Figure 6 varies greatly due to the influence of multithreats compared with the one shown in Figure 5; the multithreats limit the range of the observer’s maneuvers. Figure 7 shows the target tracking error in the x -axis and y -axis in scenarios 3 and 4.

Scenario conditions are modified in scenarios 4b, 5, 6. In contrast to Figure 6, the trajectory in Figure 8 is quite

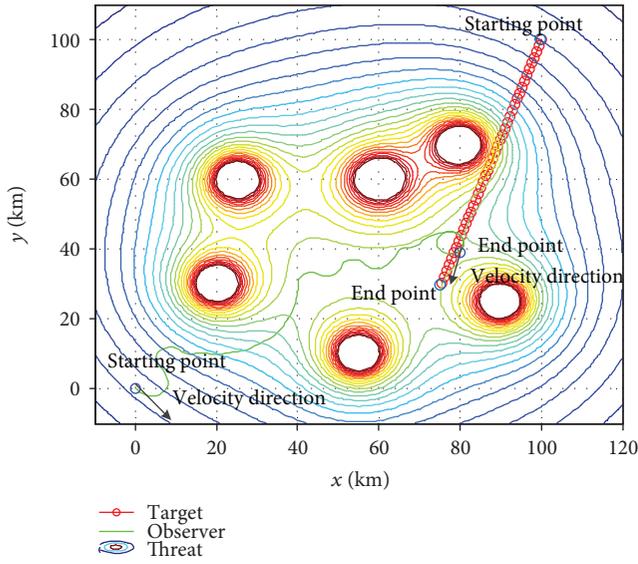


FIGURE 8: Moving target tracking in scenario 4(b).

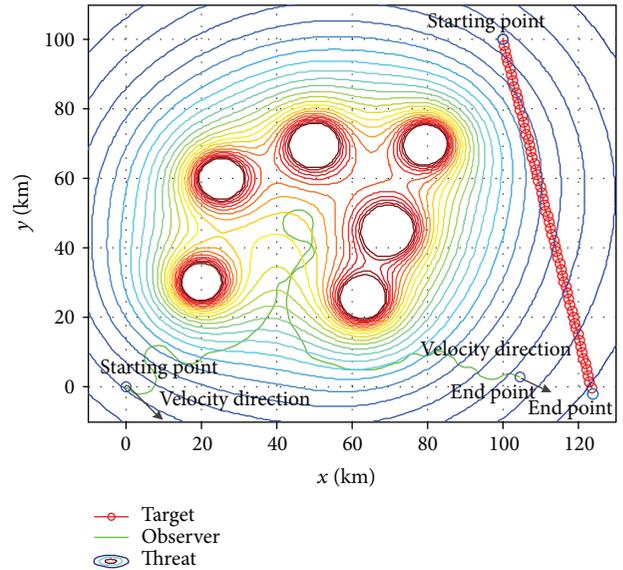


FIGURE 10: "Trap" problem in threats environment in scenario 6.

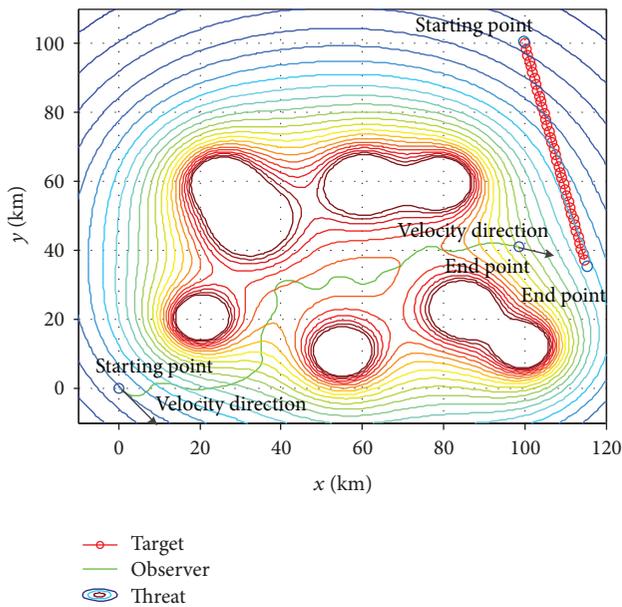


FIGURE 9: Moving target tracking in scenario 5.

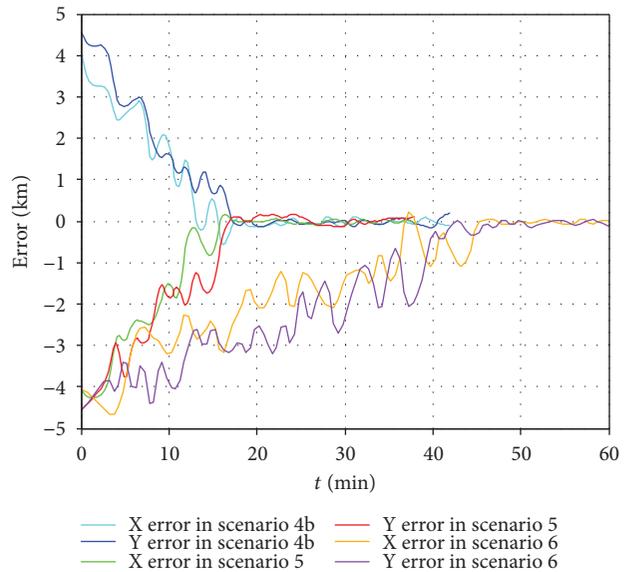


FIGURE 11: Target tracking results in scenarios 4b, 5, 6.

different because of the change of motion direction of the target, and at the end of tracking, the observer is very close to the target; it moves circularly to obtain more bearing information. In Figure 9, the number of threats increases; the observer changes the trajectory in this complex threat environment compared with Figure 6. Because the target is moving and the trajectory of the observer is not linked to the target all the time, so the changes of angles enable the reward of BOT.

Local minimum may occur in trajectory planning; in this paper, it represents the "trap" which makes up of threats. Next step, we will test the ability of the algorithm to resolve the "trap" problem. The "trap" consists of 6 threats in

Figure 10; when the observer comes into the "trap", the path ahead is blocked by threats, it can select the reasonable maneuvers based on the reward function, as we can see in Figure 10, and the observer turns around and moves out of the "trap." The result illustrates the good performance of the algorithm in resolving the "trap" problem. Figure 11 shows the target tracking error in the x -axis and y -axis in scenarios 4b, 5, and 6.

7.3. Performance Comparison. Firstly, the bearing reward of the optimal maneuver is shown in Figure 12, as we can see, the Fisher information increases significantly after 20 min; due to the influence of the distance parameter, its order of

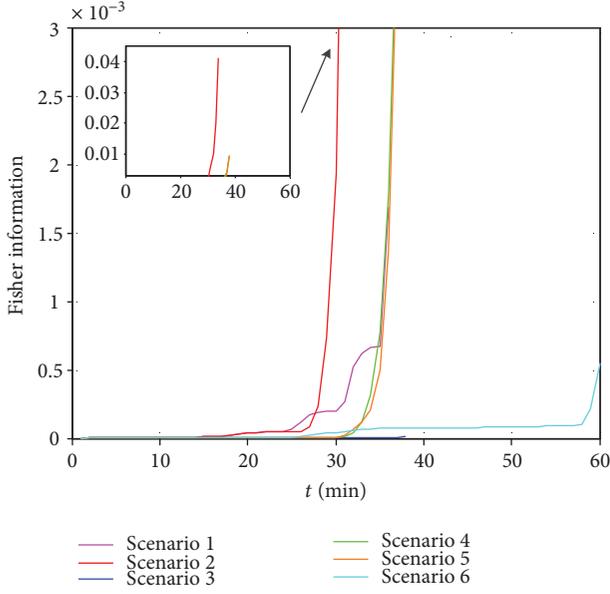


FIGURE 12: Fisher information of the scenarios.

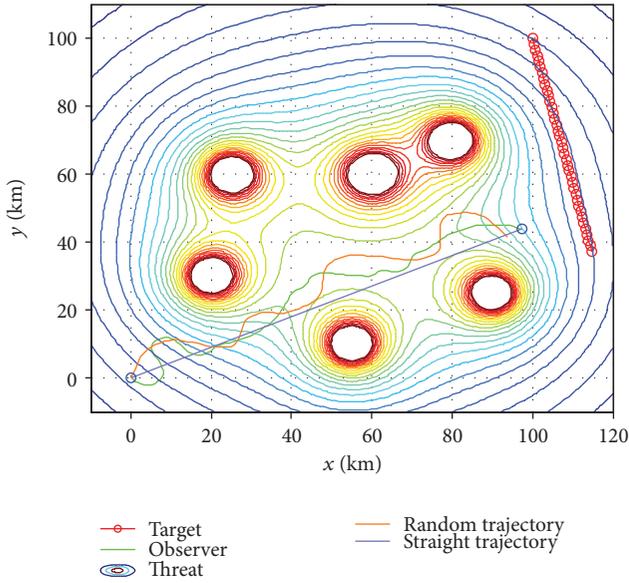


FIGURE 13: Different trajectories in scenario 4.

magnitude is very small, so the variation of the lines in scenario 3 and scenario 6 is not obvious. In scenarios 1, 2, 4, and 5, the observer approaches the target quickly, which results in the decrease of the value of the distance parameter and the increase of the Fisher information. Due to the influence of threats, the variation of the Fisher information does not apply to a certain principle. The time horizons of motions of the observer are different in these scenarios. The moving directions of the target in scenario 4b are different, so it is inapplicable to compare its Fisher information with others.

Then, the bearing reward of optimal maneuvers, random maneuvers, and fixed maneuvers in scenario 4 is compared to illustrate the performance of the algorithm proposed. The

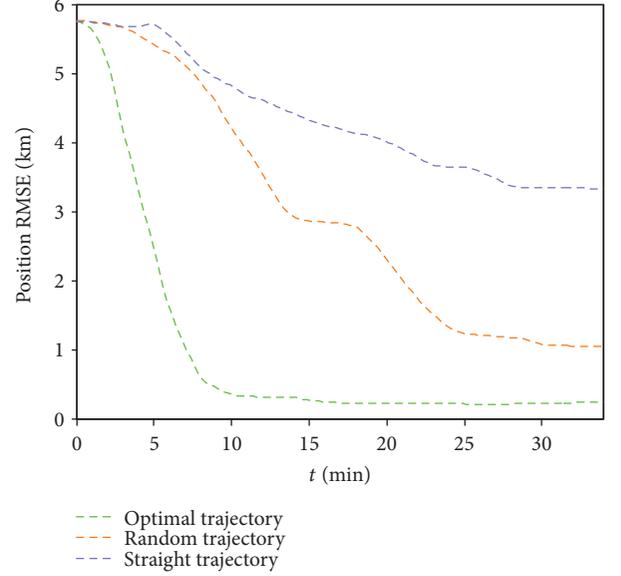


FIGURE 14: Comparison of position RMSE with different trajectories.

root mean square error (RMSE) of positions is applied to evaluate the accuracy of BOT.

$$\text{RMSE}(k) = \left(\frac{1}{N_{\text{MC}}} \sum_{j=1}^{N_{\text{MC}}} \left\| \mathbf{x}_k - \hat{\mathbf{x}}_k^j \right\|_2^2 \right)^{1/2}, \quad (33)$$

where N_{MC} is the number of iterations for Monte Carlo simulation, $\hat{\mathbf{x}}_k^j$ is the estimated state at time t for Monte Carlo simulation j , and \mathbf{x}_k is the real state of the target.

Figure 13 shows the trajectories of optimal maneuvers, random maneuvers, and fixed maneuvers; the green line shows the optimal trajectory, the orange line shows the random trajectory, and the purple line shows the trajectory of fixed maneuvers. Figure 14 demonstrates the difference of accuracy of three maneuvering strategies. As we can see, after 10 minutes, there is a slight fluctuation in every RMSE line, which is caused by the costs of threats when the observer is approaching the threats. The optimal maneuvering strategy is characterized with the minimum position RMSE, and the optimal maneuvering enables the position RMSE to decrease at the fastest speed. The straight trajectory has the worst accuracy.

Then, three filtering methods EKF [15], UKF [14], and CKF are compared to test their performances in scenario 4. The result is shown below.

Figure 15 shows the results of three filtering methods applied in scenario 4. The position RMSE of CKF and UKF decreases faster compared with the one with EKF, which illustrates the superiority of CKF and UKF in nonlinear filtering with the moving target.

8. Conclusion

The optimal maneuvering problem for BOT considering the influence of threats is solved by the finite-horizon MDP; the

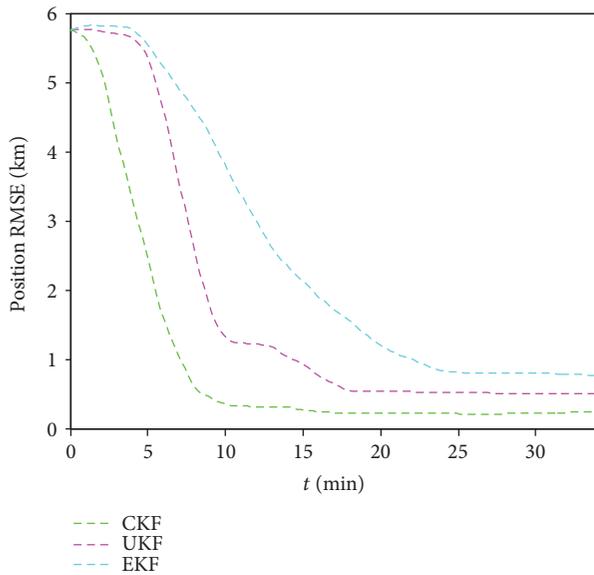


FIGURE 15: Comparison of position RMSE with EKF, UKF, CKF.

key is to connect the filtering and maneuvering. The quantization method was used to discretize the BOT process to satisfy the working condition of MDP, the initial state estimate and the corresponding error covariance for quantization are provided by CKF for each period, and reward function balances the reward of BOT and the cost of threat avoidance. The comparison results in several scenarios show the superior performance of the proposed algorithm. However, only one target is considered in this paper, and the scenario with multiple targets still needs to be solved. In addition, optimal navigation in the threat environment (which involves steering a mobile observer from an initial position to a final position [21]) and filtering method will be considered in future research.

Conflicts of Interest

The authors declare no conflict of interest.

Acknowledgments

This study was supported by the National Natural Science Foundation of China (Grant nos. 61673392, 61703420).

References

- [1] J. A. Fawcett, "Effect of course maneuvers on bearings-only range estimation," *IEEE Transactions On Acoustics, Speech and Signal Processing*, vol. 36, no. 8, pp. 1193–1199, 1988.
- [2] Y. Oshman and P. Davidson, "Optimization of observer trajectories for bearings-only target localization," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 35, no. 3, pp. 892–902, 1999.
- [3] M. Gavish and A. J. Weiss, "Performance analysis of bearing-only target location algorithms," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 28, no. 3, pp. 817–828, 1992.
- [4] M. F. Huber, "Chebyshev polynomial Kalman filter," *Digital Signal Processing*, vol. 23, no. 5, pp. 1620–1629, 2013.
- [5] T. Yang, P. G. Mehta, and S. P. Meyn, "Feedback particle filter," *IEEE Transactions on Automatic Control*, vol. 58, no. 10, pp. 2465–2480, 2013.
- [6] B. Jia, M. Xin, and Y. Cheng, "High-degree cubature Kalman filter," *Automatica*, vol. 49, no. 2, pp. 510–518, 2013.
- [7] J. P. le Cadre and S. Laurent-Michel, "Optimizing the receiver maneuvers for bearings-only tracking," *Automatica*, vol. 35, no. 4, pp. 591–606, 1999.
- [8] M. T. Sabet, A. R. Fathi, and H. R. Mohammadi Daniali, "Optimal design of the own ship maneuver in the bearing-only target motion analysis problem using a heuristically supervised extended Kalman filter," *Ocean Engineering*, vol. 123, pp. 146–153, 2016.
- [9] R. Karlsson and F. Gustafsson, "Recursive Bayesian estimation: bearings-only applications," *IEE Proceedings - Radar, Sonar and Navigation*, vol. 152, no. 5, pp. 305–313, 2005.
- [10] N. Modalavalasa, G. Sasi Bhushana Rao, K. Satya Prasad, L. Ganesh, and M. N. V. S. S. Kumar, "A new method of target tracking by EKF using bearing and elevation measurements for underwater environment," *Robotics and Autonomous Systems*, vol. 74, pp. 221–228, 2015.
- [11] M. Mallick, A. Sinha, and J. Liu, "Enhancements to bearing-only filtering," in *2017 20th International Conference on Information Fusion (Fusion)*, pp. 1442–1449, Xi'an, China, July 2017.
- [12] A. Pérez, C. Jauffret, and D. Pillon, "Bearings-only target motion analysis: observability when the observer maneuvers smoothly," in *20th International conference on Information Fusion*, pp. 1450–1457, Xi'an, China, July 2017.
- [13] A. Nègre, O. Marceau, and D. Laneuvre, "Stochastic control for underwater optimal trajectories," in *2012 IEEE Aerospace Conference*, pp. 1–7, Big Sky, MT, USA, March 2012.
- [14] H. Zhang, B. de Saporta, and F. Dufour, "Optimal trajectories for underwater vehicles by quantization and stochastic control," in *2014 17th International Conference on Information Fusion (FUSION)*, pp. 1–8, Salamanca, Spain, July 2014.
- [15] H. Zhang, F. Dufour, and J. Anselmi, "Piecewise optimal trajectories of observer for bearings-only tracking by quantization," in *2017 20th International Conference on Information Fusion (Fusion)*, pp. 1458–1464, Xi'an, China, July 2017.
- [16] H. Choi, Y. Kim, and I. Hwang, "Reactive collision avoidance of unmanned aerial vehicles using a single vision sensor," *Journal of Guidance, Control, and Dynamics*, vol. 36, no. 4, pp. 1234–1240, 2013.
- [17] C. Luo, S. I. McClean, G. Parr, L. Teacy, and R. de Nardi, "UAV position estimation and collision avoidance using the extended Kalman filter," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 6, pp. 2749–2762, 2013.
- [18] I. Arasaratnam and S. Haykin, "Cubature Kalman filters," *IEEE Transactions on Automatic Control*, vol. 54, no. 6, pp. 1254–1269, 2009.
- [19] A. Feuer and G. C. Goodwin, "On-line quantization in nonlinear filtering," *Journal of Statistical Computation and Simulation*, vol. 83, no. 7, pp. 1210–1222, 2013.
- [20] G. Pagès and H. Pham, "Optimal quantization methods for nonlinear filtering with discrete-time observations," *Bernoulli*, vol. 11, no. 5, pp. 893–932, 2005.
- [21] A. N. Bishop and P. N. Pathirana, "Optimal trajectories for homing navigation with bearing measurements," in *Proceedings of the 17th World Congress the International Federation of Automatic Control*, pp. 12117–12123, Seoul, Korea, July 2008.



Hindawi

Submit your manuscripts at
www.hindawi.com

