

## Research Article

# A Risk-Based Multisensor Optimization Scheduling Method for Target Threat Assessment

Yunpu Zhang  and Ganlin Shan 

*Department of Electronic and Optical Engineering, Army Engineering University, Shijiazhuang 050003, China*

Correspondence should be addressed to Ganlin Shan; [shanganlin@163.com](mailto:shanganlin@163.com)

Received 3 June 2019; Revised 1 July 2019; Accepted 11 July 2019; Published 4 August 2019

Academic Editor: Volodymyr Ponomaryov

Copyright © 2019 Yunpu Zhang and Ganlin Shan. 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 reasonable scheduling of multisensor systems to maximize combat benefits has become a research hotspot in the field of sensor management. To minimize the uncertainty in the threat level of targets and improve the survivability of sensors, a risk-based multisensor scheduling method is proposed in this paper. In this scheduling problem, the best sensors are systematically selected to observe targets for the trade-off between the threat assessment risk and the emission risk. First, the scheduling problem is modelled as a partially observable Markov decision process (POMDP) for target threat assessment. Second, the calculation methods of the threat assessment risk and the emission risk are proposed to quantify the potential loss caused by the uncertainty in the threat level of targets and the emission of sensors. Then, a nonmyopic sensor scheduling objective function is built to minimize the total risk which is the weighted sum of the threat assessment risk and the emission risk. Furthermore, to solve the high complexity computational problem in optimization, a decision tree search algorithm based on branch pruning is designed. Finally, simulations are conducted, and the results show that the proposed algorithm can significantly reduce the searching time and memory consumption in optimization compared with those of traditional algorithms, and the proposed method has a better risk control effect than the existing sensor scheduling methods.

## 1. Introduction

With the development of sensing technology, multisensor systems play an increasingly important role in various fields. In the military field, the multisensor systems often have constraints in operation, algorithmic complexity, deployment space, and other aspects, especially when the uncertainties of the target state increase, and the operational difficulty of the sensor system will increase greatly [1, 2]. Therefore, it is necessary to manage sensors in an effective way to maximize combat benefits. Researchers have begun to focus on Bayesian management optimization methods since Nash used the linear programming theory to establish a sensor management objective function in 1977. Until now, the management method based on Bayesian theory has evolved into three main methods, namely, task-based methods, information-based methods, and risk-based methods [3].

The above three methods all build objective functions related to the optimal scheme of sensor management. In the task-based method, the objective function is directly

related to the tasks that the sensors need to execute. Typical examples are the covariance matrix of target state [4–6], the posterior Cramér-Rao lower bound [7–9], the probability of target detection [10–12], and the posterior expected number of targets [13]. The information-based method is aimed at reducing the uncertainty in targets or the environment and is driven by information, which can be approximated as a general management model for various tasks. This kind of method usually establishes the objective function relative to the information gain of the sensors. Typical examples are Shannon entropy [14], Fisher information [15–17], Kullback-Leibler divergence [18, 19], and Rényi divergence [20–22]. The above two methods focus on optimizing one or more indicators through sensor management. However, due to the uncertainty in the target state and sensor measurement, there are uncertain risks in every decision. In some cases, it is better to control the risks for reducing the potential losses caused by decisions, rather than obtaining the optimal values of these indicators [23, 24]. For example, a better target tracking accuracy can be obtained by the first two methods

in the process of target tracking. However, if the targets need not to be attacked, it is enough to keep the target without loss instead of obtaining too high tracking accuracy. (If all the possible positions where the targets can occur are distributed in the radar beam range, even if the target tracking accuracy is poor, the target can not be lost) At this time, it is of less practical significance to consider the target tracking accuracy. In risk-based sensor management, the risk of losing the target caused by different tracking schemes is further considered in the model, which makes the model more practical. Furthermore, the risk-based method also focuses on decreasing the risk caused by the misjudgement of the target state, such as the classification error in target recognition [25, 26] and the false alarm error in target detection [27].

However, previous work has not focused on controlling the risk in the process of target threat assessment. Target threat assessment is the basis of many combat tasks, and commanders will determine what action to take with regard to the target by the assessment results. However, the assessment results will be affected by the measurement error of sensors and the uncertainty in the target state, which may generate corresponding risks. For example, when a high-threat target is misjudged as a low-threat target, we will assign fewer defensive resources to the target, which may result in a lethal attack on the defence target, and when a low-threat target is misjudged as a high-threat target, this misjudgement may only result in a waste of defensive resources, which represents a smaller loss than in the previous case. Therefore, it is necessary to control the risks in the process of target threat assessment to minimize the uncertainty in assessment results. Furthermore, the emitted signal of the active sensor may be intercepted by enemy detectors, thus exposing its position to the enemy. Therefore, it is necessary to manage the emission risk of active sensors to improve their battlefield survivability. A reasonable quantification of the emission state of the sensor is a prerequisite for controlling emission risk. In [28–30], the transmitting power, target echo power, target receiver sensitivity, and other parameters are used to calculate the intercepted probability of the sensor to represent the emission state at each time step. However, it is difficult to obtain prior knowledge, such as the target receiver sensitivity in practice. To solve this problem, the emission level impact (ELI) is used to replace the intercepted probability in [31], which represents the cumulative intercepted emission of the sensor intercepted by the enemy. It is not necessary to obtain related parameters of targets for the calculations, which has good practical application value.

In this paper, we aim to schedule sensors to control the total risk in the process of target threat assessment. The total risk is divided into the threat assessment risk and the emission risk to quantify the potential loss caused by the uncertainty in the threat level of targets and the electromagnetic emission of sensors. Moreover, the above risk-based management methods are all myopic management methods; that is to say, the methods take the minimum risk in the next sampling time step as the optimization objective. Although the method has a lower calculation cost, it makes the decision-making optimization of the whole sensor management process become a greedy search to a certain extent, without considering the influence of sensor

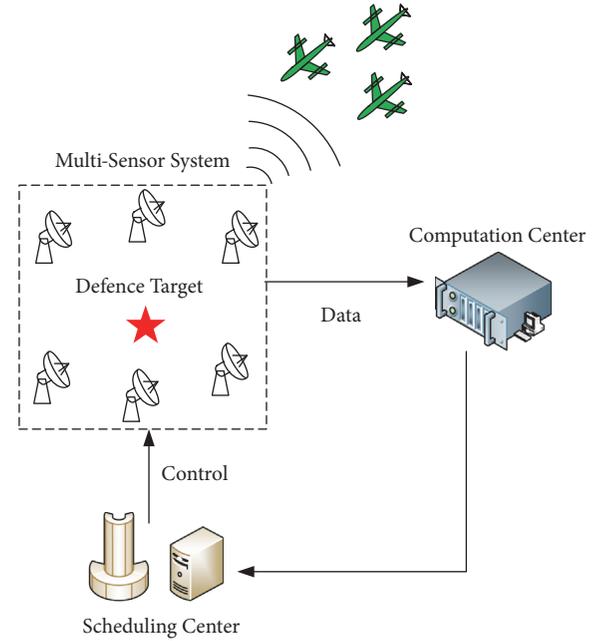


FIGURE 1: Schematic of multisensor scheduling.

actions on the future system state [32]. Therefore, this paper expands the risk-based management method from a myopic to a nonmyopic method, which takes the cumulative risk over a period of time as the basis for decision-making to obtain better combat benefits.

The main ideas in this paper are summarized as follows. In Section 2, sensor scheduling model is modelled as a partially observable Markov decision process (POMDP). In Section 3, the target threat and emission state are quantified, and calculation methods of the threat assessment risk and the emission risk are proposed. Based on the analysis of Sections 2 and 3, a nonmyopic multisensor scheduling objective function is established in Section 4. Then, a decision tree search algorithm is proposed to solve this scheduling problem in Section 5. In Section 6, the simulations are conducted and the paper is concluded in Section 7.

## 2. Sensor Scheduling Model

For the convenience of presentation and expression, we made the following assumptions.

*Assumption 1.* All sensors work independently and all of the sensors are active sensors.

*Assumption 2.* All targets move independently.

*Assumption 3.* There are  $M$  sensors to assess the threat level of  $N$  targets, which tend to attack our defence target.

*Assumption 4.* Let  $k = 0, 1, 2, 3, \dots$  denote discrete time step.

A schematic of multisensor scheduling is shown in Figure 1. Because of the uncertainty in the sensor measurements and the randomness of target motions in the whole scheduling process, the scheduling problem is a decision-making

problem under uncertain information conditions, and the POMDP is a theoretical method to study multistage decision-making in a stochastic environment [33]. Therefore, the sensor scheduling problem can be modelled as a continuous-state, discrete-time POMDP, which is described in detail as follows.

**2.1. Sensor Scheduling Action.** To denote the sensor assignment at time step  $k$ , sensor scheduling action  $\mathbf{A}_k = [a_k^{n,m}]_{n \times m}$  is an  $N \times M$  matrix where elements  $a_k^{n,m} = 1$  or  $a_k^{n,m} = 0$  indicate whether sensor  $n$  is to be scheduled for assessing the threat level of target  $m$  at the next time step  $k + 1$ .

**2.2. System State.** The system state is denoted by  $\mathbf{U}_k = [\mathbf{X}_k, \mathbf{E}_k]^T$ .

$\mathbf{X}_k = [\mathbf{X}_k^1, \dots, \mathbf{X}_k^M]^T$  indicates the state of all targets at time step  $k$ , where  $\mathbf{X}_k^m = [x_k^m, \dot{x}_k^m, y_k^m, \dot{y}_k^m, z_k^m, \dot{z}_k^m]^T$  ( $m = 1, \dots, M$ ) is the state of target  $m$  including the position coordinates and velocity information in a three-dimensional Cartesian coordinate system. The target state at the next time step can be determined according to the following state transition equation:

$$\mathbf{X}_{k+1}^m = \mathbf{F}^m \mathbf{X}_k^m + \mathbf{G}^m \boldsymbol{\varepsilon}_{k+1}^m, \quad (1)$$

where  $\mathbf{F}^m$ ,  $\mathbf{G}^m$ , and  $\boldsymbol{\varepsilon}_k^m$  represent the state transition matrix, process noise gain matrix, and Gaussian process noise, respectively. Generally, the process noise gain matrix and the process noise are used to describe the uncertainties of target state distribution in motions. The process noise gain matrix  $\mathbf{G}^m$  represents the magnitude of Gaussian process noise on each state variable. The Gaussian process noise  $\boldsymbol{\varepsilon}_{k+1}^m$  is zero-mean white Gaussian noise and the covariance matrix of the process noise is  $\mathbf{Q}_{k+1}^m = E[\mathbf{G}^m \boldsymbol{\varepsilon}_{k+1}^m (\boldsymbol{\varepsilon}_{k+1}^m)^T (\mathbf{G}^m)^T] = \mathbf{G}^m (\sigma_{\varepsilon,k+1}^m)^2 (\mathbf{G}^m)^T$ , where  $\sigma_{\varepsilon,k}^m$  is the process noise standard deviation of the position estimates [28, 34]. There are two common motion models: the nearly constant velocity (NCV) model and nearly constant turn (NCT) model, which can be described as

$$\mathbf{F}_{\text{NCV}}^m = \begin{bmatrix} 1 & \tau & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & \tau & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & \tau \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad (2)$$

$$\mathbf{Q}_{\text{NCV},k+1}^m = \begin{bmatrix} \frac{1}{4}\tau^4 & \frac{1}{2}\tau^3 & 0 & 0 & 0 & 0 \\ \frac{1}{2}\tau^3 & \tau^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{4}\tau^4 & \frac{1}{2}\tau^3 & 0 & 0 \\ 0 & 0 & \frac{1}{2}\tau^3 & \tau^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{4}\tau^4 & \frac{1}{2}\tau^3 \\ 0 & 0 & 0 & 0 & \frac{1}{2}\tau^3 & \tau^2 \end{bmatrix} (\sigma_{\varepsilon,k+1}^m)^2$$

$$\mathbf{F}_{\text{NCT}}^m = \begin{bmatrix} 1 & \frac{\sin \beta \tau}{\beta} & 0 & \frac{\cos \beta \tau - 1}{\beta} & 0 & 0 \\ 0 & \cos \beta \tau & 0 & -\sin \beta \tau & 0 & 0 \\ 0 & \frac{1 - \cos \beta \tau}{\beta} & 1 & \frac{\sin \beta \tau}{\beta} & 0 & 0 \\ 0 & \sin \beta \tau & 0 & \frac{\sin \beta \tau}{\beta} & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & \tau \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad (3)$$

$$\mathbf{Q}_{\text{NCT},k+1}^m = \mathbf{Q}_{\text{NCV},k+1}^m$$

where  $\tau$  is sampling interval and  $\beta$  is the turn rate of the target.

$\mathbf{E}_k = [E_k^1, E_k^2, \dots, E_k^N]^T$  indicates the emission state of all sensors at time step  $k$ , where  $E_k^n$  ( $n = 1, \dots, N$ ) is the ELI state of sensor  $n$ , representing the cumulative intercepted emission until time step  $k$ . The value of  $E_k^n$  can be quantized into a positive integer set  $\{1, 2, \dots, E_{\max}\}$ , and each value in this set represents a true emission level. The larger the value, the greater the probability of the sensor being attacked by the enemy [31]. The ELI state transition process can be approximated as a Markov process by introducing state transition matrix  $\mathbf{T}^n$  to describe the state transition. If sensor  $n$  is activated to the assess target threat at time step  $k$ , then

$$\mathbf{T}^n = (t_{i,j}^n)_{i,j \in \{1,2,3,\dots,E_{\max}\}} \quad (4)$$

where  $t_{i,j}^n = p(E_k^n = j | E_{k-1}^n = i)$  is the transition probability. Otherwise,  $\mathbf{T}^n$  is an  $E_{\max}$  dimension identity matrix.

**2.3. System Observation.** Similarly, the system observation is denoted by  $\mathbf{Z}_k = [\mathbf{Z}_{X_k}, \mathbf{Z}_{E_k}]^T$ .

$\mathbf{Z}_{X_k} = [\mathbf{Z}_{X_k}^1, \dots, \mathbf{Z}_{X_k}^M]$  represents the observations of all targets at time step  $k$ , where observation value  $\mathbf{Z}_{X_k}^m$  of target  $m$  is obtained by sensor measurements. If sensor  $n$  is activated to observe target  $m$ , the measurement equation is as follows:

$$\mathbf{Z}_{X_k}^m = H^n(\mathbf{X}_k^m) + \boldsymbol{\gamma}_k^n = \begin{bmatrix} r_k^{m,n} \\ \zeta_k^{m,n} \\ \phi_k^{m,n} \end{bmatrix} + \begin{bmatrix} \sigma_{r,k}^n \\ \sigma_{\zeta,k}^n \\ \sigma_{\phi,k}^n \end{bmatrix} \quad (5)$$

where  $H(\mathbf{X}_k^m)$  represents the measurement equation of sensor  $n$ ,  $\boldsymbol{\gamma}_k^m$  is the Gaussian measurement noise,  $\sigma_{r,k}^n$ ,  $\sigma_{\zeta,k}^n$ , and  $\sigma_{\phi,k}^n$  represent the measurement noise of range  $r_k^{m,n}$ , azimuth angle  $\zeta_k^{m,n}$ , and elevation angle  $\phi_k^{m,n}$ , respectively. The calculation methods of  $r_k^{m,n}$ ,  $\zeta_k^{m,n}$ , and  $\phi_k^{m,n}$  are shown in

$$r_k^{m,n} = \sqrt{(x_k^m - x^n)^2 + (y_k^m - y^n)^2 + (z_k^m - z^n)^2}$$

$$\zeta_k^{m,n} = \arctan \frac{y_k^m - y^n}{x_k^m - x^n} \quad (6)$$

$$\phi_k^{m,n} = \arctan \frac{(z_k^m - z^n)^2}{\sqrt{(x_k^m - x^n)^2 + (y_k^m - y^n)^2}}$$

where  $x^n$ ,  $y^n$ , and  $z^n$  are the position coordinates of sensor  $n$ .

$\mathbf{Z}_{E_k} = [Z_{E_k}^1, Z_{E_k}^2, \dots, Z_{E_k}^N]^T$  represents the instantaneous emission observations of all sensors at time step  $k$ , where  $Z_{E_k}^n$  ( $n = 1, \dots, N$ ) denotes the instantaneous emission of  $e_k^n = E_k^n - E_{k-1}^n$ . We quantize the instantaneous emission as a positive integer set  $\{1, 2, 3, \dots, O_{\max}\}$ , which can be called the instantaneous observed emission level [31]. If sensor  $n$  is activated at time step  $k$ , then its instantaneous observed emission level can be calculated by a set of observation matrices:

$$\mathbf{W}(Z_{E_k}^n = o) = (w_{o,i,j}^n)_{i,j \in \{1,2,3,\dots,E_{\max}\}}, \quad (7)$$

$$o \in \{1, 2, 3, \dots, O_{\max}\}$$

where  $w_{o,i,j}^n = p(Z_{E_k}^n = o \mid E_k^n = j, E_{k-1}^n = i)$  indicates the probability that the level is  $o$  when the ELI state changes from  $i$  to  $j$ . Otherwise, its observation matrix is an  $O_{\max}$  dimension identity matrix.

$$d_k^m = \sqrt{(x_k^m - x_p)^2 + (y_k^m - y_p)^2}$$

$$h_k^m = z_k^m - z_p$$

$$v_k^m = \sqrt{(\dot{x}_k^m)^2 + (\dot{y}_k^m)^2 + (\dot{z}_k^m)^2} \quad (8)$$

$$\alpha_k^m = \arccos \frac{\left( (x_k^m - x_p) \dot{x}_k^m + (y_k^m - y_p) \dot{y}_k^m + (z_k^m - z_p) \dot{z}_k^m \right) / \left( v_k^m \cdot \sqrt{(x_k^m - x_p)^2 + (y_k^m - y_p)^2 + (z_k^m - z_p)^2} \right)}{\pi}$$

where  $x_p, y_p$ , and  $z_p$  are the coordinates of our defence target position.

Then the threat function of each factor is constructed as follows:

$$\theta_{d,k}^m = \begin{cases} 1 & d_k^m \leq d_0 \\ e^{-c_d(d_k^m - d_0)^2} & d_k^m > d_0 \end{cases}$$

$$\theta_{h,k}^m = \begin{cases} 1 & h_k^m \leq h_0 \\ e^{-c_h(h_k^m - h_0)^2} & h_k^m > h_0 \end{cases} \quad (9)$$

$$\theta_{v,k}^m = \begin{cases} 0 & v_k^m \leq v_0 \\ (1 - e^{-c_v(v_k^m - v_0)}) & v_k^m > v_0 \end{cases}$$

$$\theta_{\alpha,k}^m = e^{-c_\alpha(\alpha_k^m)^2} \quad 0^\circ \leq \alpha_k^m \leq 180^\circ$$

where  $d_0, h_0$ , and  $v_0$  are the points where the threat degree is equal to the extremum. And  $c_d, c_h, c_v$ , and  $c_\alpha$  are the distance coefficient, the height coefficient, the speed coefficient, and the angle coefficient, respectively.

After obtaining the threat degree of each factor, the total threat degree is calculated as

$$\theta_k^m = s_d \theta_{d,k}^m + s_h \theta_{h,k}^m + s_v \theta_{v,k}^m + s_\alpha \theta_{\alpha,k}^m \quad (10)$$

### 3. Risk Calculation Method

Predicting the total risk in future action cycles is the premise of formulating scheduling plans. In this paper, we divide the total risk into the threat assessment risk and the emission risk.

**3.1. Threat Assessment Risk.** In the process of threat assessment, since the threat degree is a variable that changes with the target state, the uncertainty in the target state will be expanded when the threat degree is calculated, making the result of the threat assessment difficult to accurately assess. For example, when we consider that threat degree  $\theta$  is a variable related to target velocity  $v$  and horizontal distance  $d$ , its uncertainty distribution is shown in Figure 2.

**3.1.1. Threat Model.** In this paper, we consider the horizontal distance  $d_k^m$ , the height  $h_k^m$ , the velocity  $v_k^m$ , and the course angle  $\alpha_k^m$  as the threat degree factors of target  $m$  at time step  $k$ , which can be calculated as

where  $s_d, s_h, s_v$ , and  $s_\alpha$  represent the weights of the horizontal distance, height, speed, and course angle to the total threat degree, respectively. Generally, the weights are set according to the operational experience and actual situation [3, 35, 36].

Then, threat level  $l_k^m$  of the target can be determined by setting level rule. In this paper, we define the threat level as 1 (low-threat level), 2 (medium-threat level), and 3 (high-threat level), and the level rule is as follows:

$$l_k^m = \begin{cases} 1 & 0 \leq \theta_k^m < \theta_{\text{medium}} \\ 2 & \theta_{\text{medium}} \leq \theta_k^m < \theta_{\text{high}} \\ 3 & \theta_{\text{high}} \leq \theta_k^m \leq 1 \end{cases} \quad (11)$$

where  $\theta_{\text{medium}}$  and  $\theta_{\text{high}}$  represent the boundary points of the threat level.

In addition, more threat degree factors and different threat levels can be considered in the threat model according to different actual situations.

**3.1.2. Threat Assessment Risk Calculation.** Because the threat degree is only related to the target state, the risk is stated by estimating the target state. Since the target state cannot be fully observed, the target belief state  $\mathbf{b}_{X_k} =$

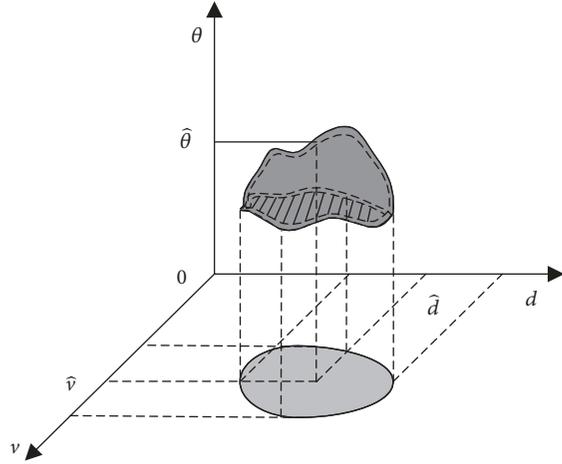


FIGURE 2: Uncertainty distribution of the target threat degree.

$[\mathbf{b}_{X_k}^1, \mathbf{b}_{X_k}^2, \dots, \mathbf{b}_{X_k}^N]^T$  is introduced, which is a sufficient statistic of historical information, indicating the probability distribution of the target state [37].

According to the Bayesian estimation method, the transition of the target belief state can be divided into the prediction stage and update stage. The prediction stage is described as

$$p(\mathbf{X}_{k+1}^m | \mathbf{b}_{X_k}^m) = \int p(\mathbf{X}_{k+1}^m | \mathbf{X}_k^m) \mathbf{b}_{X_k}^m d\mathbf{X}_k^m \quad (12)$$

The updating stage is described as

$$\begin{aligned} \mathbf{b}_{X_{k+1}}^m &= p(\mathbf{X}_{k+1} | \mathbf{b}_{X_k}^m, \mathbf{Z}_{X_{k+1}}^m) \\ &= \frac{p(\mathbf{X}_{k+1}^m | \mathbf{b}_{X_k}^m) p(\mathbf{Z}_{X_{k+1}}^m | \mathbf{X}_{k+1}^m)}{\int p(\mathbf{X}_{k+1}^m | \mathbf{b}_{X_k}^m) p(\mathbf{Z}_{X_{k+1}}^m | \mathbf{X}_{k+1}^m) d\mathbf{X}_{k+1}^m} \end{aligned} \quad (13)$$

If both the process noise and the measurement noise are Gaussian noise, the target belief state is also a Gaussian distribution. The target belief state can be updated by cubature Kalman filter (CKF) [38]. When the target has multiple motion models, its belief state can be updated by combining the interacting multiple model (IMM) algorithm and the CKF [39].

Then, threat assessment risk prediction can be conducted by predicting the target belief state, and the process is as follows.

*Step 1.* Obtain the belief state of target  $m$  at time step  $k$ , and predict the belief state at time step  $k + 1$  by the CKF.

*Step 2.* Obtain  $L$  sample points that can be implemented in the Monte Carlo method by sampling from the belief state at time step  $k + 1$ .

*Step 3.* Calculate the corresponding threat level by using the  $L$  sample points according to (8)-(11) and obtain the number of sample points with low-, medium-, and high-threat levels of  $L_1$ ,  $L_2$ , and  $L_3$ , respectively.

*Step 4.* Get the probability distribution of threat level  $\hat{q}_k^m = (\hat{q}_{1,k+1}^m, \hat{q}_{2,k+1}^m, \hat{q}_{3,k+1}^m) = (L_1, L_2, L_3)/L$ .

*Step 5.* Set loss matrix  $\boldsymbol{\lambda} = \begin{matrix} & 1 & 2 & 3 \\ \begin{matrix} 1 \\ 2 \\ 3 \end{matrix} & \begin{bmatrix} 0 & \lambda_{12} & \lambda_{13} \\ \lambda_{21} & 0 & \lambda_{23} \\ \lambda_{31} & \lambda_{32} & 0 \end{bmatrix} \end{matrix}$ , where  $\lambda_{ij}$

indicates the potential loss of assessment error in which the real threat level is  $i$  but the estimated level is  $j$ .

If the threat level of target  $m$  is estimated as  $\hat{l}_{k+1}^m = j$ , the risk is  $u_{k+1}^{m,j} = \sum_{i=1}^3 \lambda_{ij} \hat{q}_{i,k+1}^m$ . Then, the threat assessment risk is defined as  $u_{k+1}^m = \min\{u_{k+1}^{m,1}, u_{k+1}^{m,2}, u_{k+1}^{m,3}\}$ , and the threat level is  $\hat{l}_{k+1}^m = \arg \min\{u_{k+1}^{m,1}, u_{k+1}^{m,2}, u_{k+1}^{m,3}\}$ .

*Step 6.* Set  $k = k + 1$ , go to Step 1, and perform  $H - 1$  calculation cycles ( $H$  is the horizon length). Then, the cumulative threat assessment risk of target  $m$  is given by

$$U_{k+1:k+H}^m = \sum_{h=1}^H u_{k+h}^m = \sum_{h=1}^H \min\{u_{k+h}^{m,1}, u_{k+h}^{m,2}, u_{k+h}^{m,3}\} \quad (14)$$

Take the sensor scheduling actions into consideration;  $u_{k+1}^{m,n}$  is defined as the threat assessment risk of target  $m$  by sensor  $n$  at time step  $k + 1$ . Then, the cumulative threat assessment risk of all targets is given by

$$U_{k+1:k+H} = \sum_{h=1}^H \sum_{n=1}^N \sum_{m=1}^M a_{k+h-1}^{n,m} u_{k+h}^{m,n} \quad (15)$$

*3.2. Sensor Emission Risk.* Similarly, the ELI state of the sensors cannot be fully observed, so the ELI belief state  $\mathbf{b}_{E_k} = [(\mathbf{b}_{E_k}^1), (\mathbf{b}_{E_k}^2), \dots, (\mathbf{b}_{E_k}^N)]^T$  is introduced to indicate the probability distribution of the ELI state at time step  $k$ . If the instantaneous emission observation is  $Z_{E_{k+1}}^n = o_l$ , the ELI belief state can be updated by the hidden Markov model filter:

$$\mathbf{b}_{E_{k+1}}^n = \frac{[\mathbf{W}(Z_{E_{k+1}}^n = o_l) \odot \mathbf{T}^n]^T \mathbf{b}_{E_k}^n}{\mathbf{1}^T [\mathbf{W}(Z_{E_{k+1}}^n = o_l) \odot \mathbf{T}^n]^T \mathbf{b}_{E_k}^n} \quad (16)$$

where the symbol  $\odot$  represents the Hadamard product, and  $\mathbf{1}$  is the identity vector.

However, in actual scheduling, we cannot obtain  $Z_{E_{k+1}}^n$  at time step  $k$ , but its probability distribution is obtained as

$$\begin{aligned} p(Z_{E_{k+1}}^n | \mathbf{b}_{E_k}^n) &= \sum_{E_{k+1}^n=1}^{E_{\max}^n} \sum_{E_k^n=1}^{E_{\max}^n} p(Z_{E_{k+1}}^n, E_{k+1}^n, E_k^n | \mathbf{b}_{E_k}^n) \\ &= \sum_{E_{k+1}^n=1}^{E_{\max}^n} \sum_{E_k^n=1}^{E_{\max}^n} p(Z_{E_{k+1}}^n | E_{k+1}^n, E_k^n) p(E_{k+1}^n | E_k^n, \mathbf{b}_{E_k}^n) \\ &\cdot p(E_k^n | \mathbf{b}_{E_k}^n) = \sum_{E_{k+1}^n=1}^{E_{\max}^n} \sum_{E_k^n=1}^{E_{\max}^n} p(Z_{E_{k+1}}^n | E_{k+1}^n, E_k^n) \\ &\cdot p(E_{k+1}^n | E_k^n) \mathbf{b}_{E_k}^n \end{aligned} \quad (17)$$

The distribution is represented by the following matrix:

$$p(Z_{E_{k+1}}^n | \mathbf{b}_{E_k}^n) = \mathbf{1}^T [\mathbf{W}(Z_{E_{k+1}}^n) \odot \mathbf{T}^n]^T \mathbf{b}_{E_k}^n \quad (18)$$

Then, the predicted ELI state is updated as

$$\begin{aligned} \hat{\mathbf{b}}_{E_{k+1}}^n &= \sum_{Z_{E_{k+1}}^n=1}^{O_{\max}} \frac{[\mathbf{W}(Z_{E_{k+1}}^n) \odot \mathbf{T}^n]^T \mathbf{b}_{E_k}^n}{\mathbf{1}^T [\mathbf{W}(Z_{E_{k+1}}^n) \odot \mathbf{T}^n]^T \mathbf{b}_{E_k}^n} \\ &\quad \cdot p(Z_{E_{k+1}}^n | \mathbf{b}_{E_k}^n) \\ &= \sum_{Z_{E_{k+1}}^n=1}^{O_{\max}} [\mathbf{W}(Z_{E_{k+1}}^n) \odot \mathbf{T}^n]^T \mathbf{b}_{E_k}^n \\ &= \left\{ \left[ \sum_{Z_{E_{k+1}}^n=1}^{O_{\max}} \mathbf{W}(Z_{E_{k+1}}^n) \right]^T \odot (\mathbf{T}^n)^T \right\} \mathbf{b}_{E_k}^n \\ &= (\mathbf{T}^n)^T \mathbf{b}_{E_k}^n \end{aligned} \quad (19)$$

Furthermore, the emission cost of sensor  $n$  at time step  $k$  is calculated as

$$Y_{k+1}^n = \mathbf{V}^T \hat{\mathbf{b}}_{E_{k+1}}^n = \mathbf{V}^T (\mathbf{T}^n)^T \mathbf{b}_{E_k}^n \quad (20)$$

where  $\mathbf{V} = [1, 2, \dots, E_{\max}]^T$  represents the ELI value corresponding to the belief state.

According to the relationship between ELI and intercepted probability, the intercepted probability of the sensor is calculated as

$$\eta_{k+1}^n = \frac{Y_{k+1}^n}{E_{\max}} \times 100\% \quad (21)$$

Taking the loss into consideration when the sensor is destroyed, and the sensor emission risk is calculated as

$$Y_{k+1}^n = \eta_{k+1}^n c_n \quad (22)$$

where  $c_n \in \mathbf{c}_{1 \times N} = [c_1, \dots, c_n, \dots, c_N]$  represents the tactical value of sensor  $n$ .

Taking the sensor scheduling actions over  $H$  time steps into consideration, the cumulative emission risk is calculated as

$$\begin{aligned} R_{k+1:k+H}^n &= \sum_{h=1}^H Y_{k+h}^n = \sum_{h=1}^H \frac{\mathbf{V}^T \hat{\mathbf{b}}_{E_{k+h}}^n c_n \times 100\%}{E_{\max}} \\ &= \sum_{h=1}^H \left[ \frac{\left( \sum_{m=1}^M a_{k+h-1}^{n,m} \mathbf{V}^T (\mathbf{T}^n)^{\sum_{i=1}^h \sum_{m=1}^M a_{k+i-1}^{n,m}} \mathbf{b}_{E_k}^n \right) c_n \times 100\%}{E_{\max}} \right] \end{aligned} \quad (23)$$

Furthermore, the cumulative emission risk of all sensors is given by

$$R_{k+1:k+H} = \sum_{n=1}^N R_{k+1:k+H}^n \quad (24)$$

## 4. Objective Function

Considering the threat assessment risk and the sensor emission risk, combined with (15) and (24), the objective function over the future horizon of  $H$  time steps is given by

$$J_{k+1:k+H} = \min \{wU_{k+1:k+H} + (1-w)R_{k+1:k+H}\} \quad (25)$$

where  $J_{k+1:k+H}$  represents the total risk and  $w$  is the equilibrium coefficient.

Specifically, the sensor scheduling objective function proposed in this paper is a nonmyopic function that considers the cumulative total risk in the time step domain  $[k+1, k+H]$ . That is, the sensor scheduling scheme that can gain the lowest cumulative total risk will be selected. The optimal solution  $\mathbf{A}_{k:k+H-1}^{\text{opt}} = [\mathbf{A}_k, \mathbf{A}_{k+1}, \dots, \mathbf{A}_{k+H-1}]^{\text{opt}}$  of the objective function is the optimal scheduling scheme in the time step domain of  $[k+1, k+H]$ . At this time, the scheduling problem is converted to an optimization problem. Since the problem is a POMDP problem with continuous scheduling actions, its computational complexity will increase exponentially with increasing time step, which makes it difficult to meet the real-time requirements of scheduling. Therefore, to obtain the optimal solution in a short time, we transform the scheduling problem into a decision tree optimization problem and propose a uniform cost search (UCS) algorithm based on branch pruning in the next section.

## 5. UCS Algorithm Based on Branch Pruning

Figure 3 shows a decision tree with  $H = 3$ ,  $M = 2$ , and  $N = 4$ . As can be seen from Figure 3, each available scheduling action at each time step is a node, and the lower node contains the scheduling scheme of all the upper nodes. We denote the node cost as the cumulative total risk corresponding to the scheduling scheme contained in the node. Then, the optimal solution is converted to the scheduling scheme contained in the lowest node with the smallest node cost.

There are three commonly used decision tree search algorithms, namely, the breadth-first search (BFS), depth-first search (DFS), and UCS. Compared with BFS and DFS, in which all nodes are traversed, UCS preferentially opens the node with the lowest cost, which has the highest search speed [40]. However, as the number of nodes increases exponentially, UCS still needs a lot of time. Therefore, the branch pruning method is introduced in this paper. By estimating the lower bound of the node, the branch with a lower node bound greater than the current minimum node cost can be deleted in time, and the number of opened nodes can be reduced. For the node containing scheduling scheme  $\mathbf{A}_{k:k+h-1}$ , its lower bound is given by

$$\begin{aligned} \psi(\mathbf{A}_{k:k+h-1}) &= \omega U_{k+1:k+h} + (1-w)R_{k+1:k+h} \\ &\quad + \omega \widehat{U}_{k+h+1:k+H} + (1-w)\widehat{R}_{k+h+1:k+H} \end{aligned} \quad (26)$$

Here, node cost  $\omega U_{k+1:k+h} + (1-w)R_{k+1:k+h}$  is known, but  $\omega \widehat{U}_{k+h+1:k+H} + (1-w)\widehat{R}_{k+h+1:k+H}$  is unknown, which indicates the minimum cumulative total risk in the future time step domain of  $[k+h+1, k+H]$ . The minimum risk in future

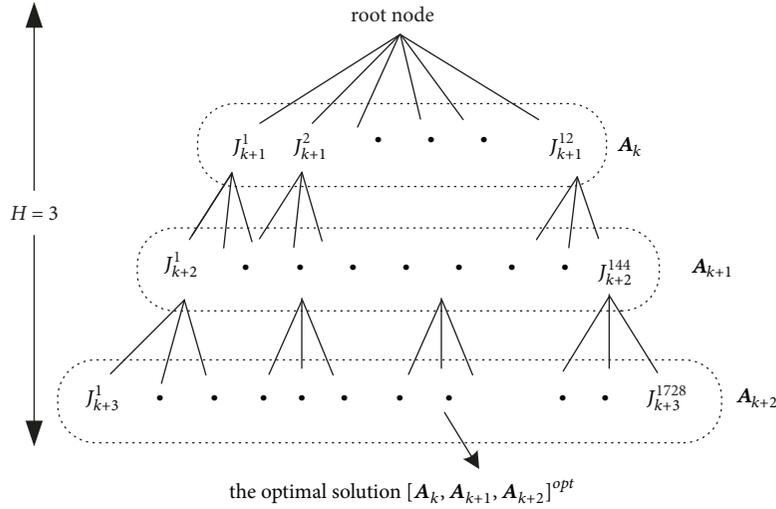


FIGURE 3: Schematic of the decision tree.

time steps must be calculated by opening all child nodes, so we propose an approximate estimation method for the lower bound value to improve the efficiency of the algorithm.

Because of the stability of the sensor measurements and the persistence of the target motions, the estimation error of the target state will not change much from the previous time step. Therefore, the threat assessment risk will not be mutated most of the time. Furthermore, according to simulations, the threat assessment risk in the next time step is usually 0.6-1.6 times that in the previous next time step. Then, the suboptimal lower bound of the threat assessment risk is given by

$$\begin{aligned} \psi_U(\mathbf{A}_{k:k+h-1}) \\ = \omega \left[ U_{k+1:k+h} + \sum_{t=1}^{H-h} (0.6)^t U_{k+h}(\mathbf{A}_{k:k+h-1}) \right] \end{aligned} \quad (27)$$

where  $U_{k+h}(\mathbf{A}_{k:k+h-1})$  represents the threat assessment risk at time step  $k+h$  after performing scheduling scheme  $\mathbf{A}_{k:k+h-1}$ .

This is because,  $\forall \tau \in \{1, 2, \dots, H-h\}$ , the following applies:

$$\begin{aligned} \widehat{R}_{k+h+1:k+H} \\ \geq (H-h) \min_{\substack{\tau=1,2,\dots,H-h \\ \forall \mathbf{A}_{k+h:k+h+\tau-1}}} \{R_{k+h+\tau}(\mathbf{A}_{k:k+h+\tau-1})\} \end{aligned} \quad (28)$$

where  $R_{k+h+\tau}(\mathbf{A}_{k:k+h+\tau-1})$  represents the emission risk at time step  $k+h$  after performing scheduling scheme  $\mathbf{A}_{k:k+h-1}$ .

Therefore, the suboptimal lower bound of the emission risk can be given:

$$\begin{aligned} \psi_R(\mathbf{A}_{k:k+h-1}) = (1-\omega) \left( R_{k+1:k+h} \right. \\ \left. + (H-h) \min_{\substack{\tau=1,2,\dots,H-h \\ \forall \mathbf{A}_{k+h:k+h+\tau-1}}} \{R_{k+h+\tau}(\mathbf{A}_{k:k+h+\tau-1})\} \right) \end{aligned} \quad (29)$$

Then, combined with (27) and (29), the suboptimal lower bound of the node is as follows:

$$\begin{aligned} \psi(\mathbf{A}_{k:k+h-1}) &= \psi_U(\mathbf{A}_{k:k+h-1}) + \psi_R(\mathbf{A}_{k:k+h-1}) \\ &= \omega \left[ U_{k+1:k+h} + \sum_{t=1}^{H-h} (0.6)^t U_{k+h}(\mathbf{A}_{k:k+h-1}) \right] + (1 \\ &\quad - \omega) \left[ R_{k+1:k+h} \right. \\ &\quad \left. + (H-h) \min_{\substack{\tau=1,2,\dots,H-h \\ \forall \mathbf{A}_{k+h:k+h+\tau-1}}} \{R_{k+h+\tau}(\mathbf{A}_{k:k+h+\tau-1})\} \right] \end{aligned} \quad (30)$$

When UCS is carried out, the lower bound of the node needs to be compared with the current optimal cumulative total risk to determine whether the pruning condition is met. In summary, the UCS algorithm based on branch pruning is as follows.

*Step 1.* Add the root node to the list, set the initial optimal risk  $\tilde{J}_{k:k+H} = +\infty$ , and specify that, once the node is opened, the node will be deleted from the list.

*Step 2.* If the list is not empty, open the first node in the list.

If the depth of the child node  $< H$ , calculate the lower bounds of all the child nodes and compare the lower bounds with  $\tilde{J}_{k:k+H}$  in turn. Delete the node with a lower bound greater than  $\tilde{J}_{k:k+H}$  and order the remaining nodes in the list from small to large by the values of the lower bounds, and go to Step 2.

If the depth of the child node  $= H$ , calculate the node cost. If the node cost is less than  $\tilde{J}_{k:k+H}$ , record the current cost as  $\tilde{J}_{k:k+H}$  and the corresponding scheduling scheme of the node as the optimal scheme, and go to Step 2.

TABLE 1: The parameters of the sensors.

| Sensor number | Coordinate     | Tactical Value | The standard deviation of the range noise | The standard deviation of the azimuth noise | The standard deviations of the elevation noise |
|---------------|----------------|----------------|---|---|--|
| 1             | (5, 5, 0) km   | 4              | 200 m                                     | 8 mrad                                      | 8 mrad   |
| 2             | (5, -5, 0) km  | 8              | 150 m                                     | 5 mrad                                      | 5 mrad   |
| 3             | (-5, 5, 0) km  | 12             | 100 m                                     | 5 mrad                                      | 5 mrad   |
| 4             | (-5, -5, 0) km | 10             | 50 m                                      | 2 mrad                                      | 2 mrad   |

*Step 3.* If the list is empty, end the search. Record the current optimal scheduling scheme as the optimal solution.

## 6. Simulations

In our simulations, 4 sensors are used to assess the threat levels of 2 enemy targets. The sensor sampling interval is 1 s and the simulation duration is 60 s. Target 1 moves along a uniform straight line, whose initial position and velocity are (18, 15, 8) km and (-250, -260, -80)m/s, respectively. Target 2 turns left at an angle of  $10^\circ$  from 20 to 30s, turns right at an angle of  $5^\circ$  from 30 to 45s, and moves along a uniform straight line during other times, and its initial position and velocity are (-6, 10, 3) km and (320, 200, 60) m/s, respectively. The sensor parameters are shown in Table 1. The values of the parameters in (9) are set as  $d_0 = 10$  km,  $h_0 = 5$  km,  $v_0 = 50$  m/s,  $c_d = 0.005$ ,  $c_h = 0.008$ ,  $c_v = 0.005$ , and  $c_\alpha = 0.0002$ . Furthermore, the weights of threat factors  $s_r$ ,  $s_d$ ,  $s_v$ , and  $s_\alpha$  are 0.4, 0.2, 0.1, and 0.3, respectively, the boundary points of the threat level are set as  $\theta_{\text{medium}} = 0.4$  and  $\theta_{\text{high}} = 0.7$ , and the assessment loss matrix

$$\lambda = \begin{matrix} & 1 & 2 & 3 \\ \begin{matrix} 1 \\ 2 \\ 3 \end{matrix} & \begin{bmatrix} 0 & 5 & 10 \\ 20 & 0 & 8 \\ 35 & 25 & 0 \end{bmatrix} \end{matrix}.$$

Moreover, we quantify the ELI state as  $\{1, 2, 3, 4\}$  and the instantaneous observed emission as  $\{1, 2, 3\}$ . For generality, the sensor with a high measurement accuracy more easily obtains a high emission state. Then, the ELI state transition matrix of each sensor is as follows:

$$T_1 = \begin{bmatrix} 0.6 & 0.2 & 0.1 & 0.1 \\ 0.5 & 0.2 & 0.2 & 0.1 \\ 0.3 & 0.3 & 0.2 & 0.2 \\ 0.3 & 0.2 & 0.3 & 0.3 \end{bmatrix}$$

$$T_2 = \begin{bmatrix} 0.5 & 0.3 & 0.1 & 0.1 \\ 0.4 & 0.3 & 0.2 & 0.1 \\ 0.2 & 0.2 & 0.3 & 0.3 \\ 0.1 & 0.2 & 0.2 & 0.5 \end{bmatrix}$$

$$T_3 = \begin{bmatrix} 0.4 & 0.3 & 0.2 & 0.1 \\ 0.3 & 0.3 & 0.3 & 0.1 \\ 0.1 & 0.2 & 0.2 & 0.5 \\ 0.1 & 0.1 & 0.2 & 0.6 \end{bmatrix}$$

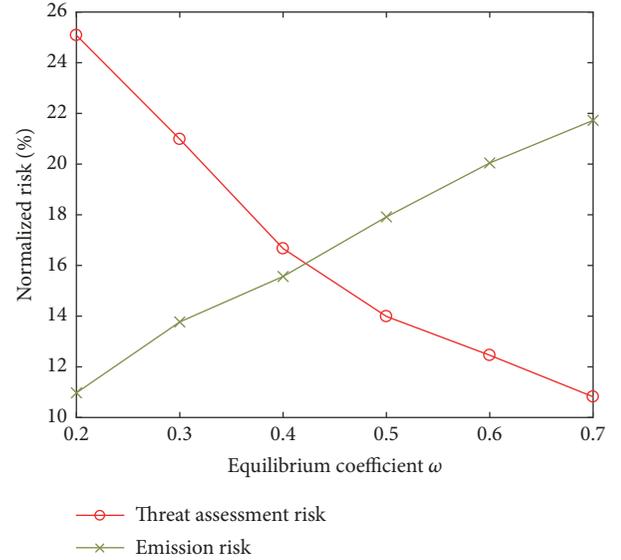


FIGURE 4: Normalized risk for different equilibrium coefficients.

$$T_4 = \begin{bmatrix} 0.2 & 0.3 & 0.2 & 0.3 \\ 0.2 & 0.2 & 0.4 & 0.2 \\ 0.1 & 0.1 & 0.2 & 0.6 \\ 0.05 & 0.1 & 0.15 & 0.7 \end{bmatrix}$$

(31)

*6.1. Determination of the Equilibrium Coefficient.* Equilibrium coefficient  $\omega$  can adjust the impacts of the threat assessment risk and emission risk on the total risk, thus affecting the decision-making process. We study the impacts of different equilibrium coefficients on the two kinds of risks for a horizon length  $H = 1$ , which are shown in Figure 4. It can be seen in Figure 4 that the greater  $\omega$ , the lower the threat assessment risk and the higher the emission risk. This phenomenon is because, with the increasing  $\omega$ , the impact of threat assessment risk on the total risk will gradually increase, and the system will pay more attention to control the threat assessment risk, which makes it fall. Similarly, the system will pay less attention to control the emission risk. Only when  $\omega = 0.4$ , are the two kinds of risks very similar, indicating that the scheduling scheme at this time can balance the impacts of the two risks well. Therefore, we choose  $\omega = 0.4$  in the next simulations.

TABLE 2: Comparisons of the algorithm performance.

| $H$ | Algorithm | Average number of opened nodes | Maximum number of stored nodes | Cumulative total risk. |
|-----|-----------|--------------------------------|--------------------------------|------------------------|
| 2   | UCS-BP    | 87 (56%)                       | 32                             | 744                    |
|     | UCS       | 126 (81%)                      | 36                             | 734                    |
| 3   | UCS-BP    | 772 (41%)                      | 107                            | 713                    |
|     | UCS       | 1168 (62%)                     | 481                            | 700                    |
| 4   | UCS-BP    | 6659 (29%)                     | 687                            | 702                    |
|     | UCS       | 9952 (44%)                     | 2036                           | 666                    |
| 5   | UCS-BP    | 42432 (16%)                    | 1254                           | 711                    |
|     | UCS       | 67683 (25%)                    | 11136                          | 668                    |

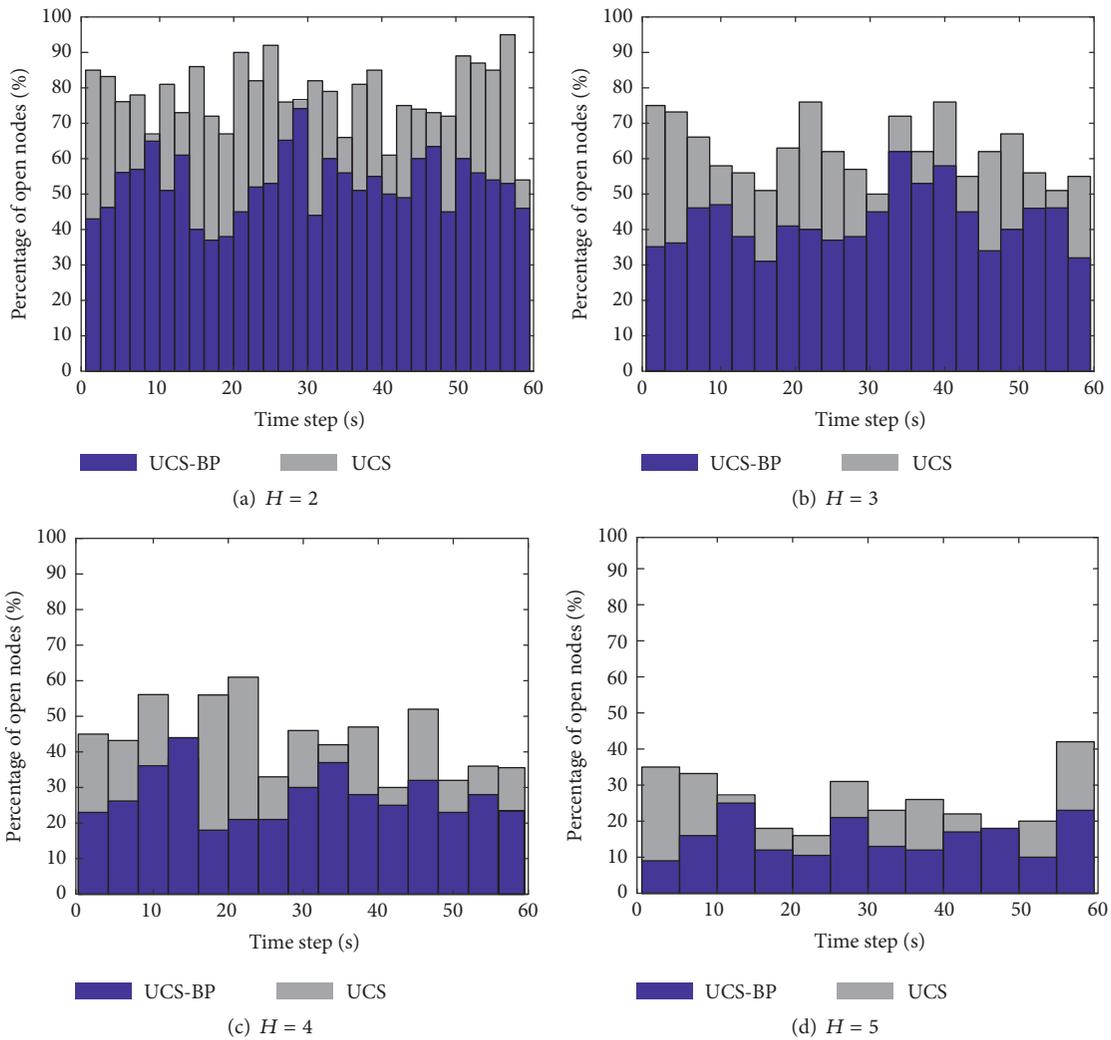


FIGURE 5: Comparison of the percentage of opened nodes for each algorithm.

6.2. Algorithm Performance Comparisons. To verify the advantages of UCS based on branch pruning (UCS-BP), we introduce traditional UCS to compare with. Figure 5 reveals the comparison of the percentage of opened nodes for each algorithm under different lengths of  $H$ . Figure 6 shows the comparison of the cumulative total risk. Table 2

summarizes the statistics, which include the average number of opened nodes, the maximum number of stored nodes, and the cumulative total risk. The search time and memory consumption of the algorithm are proportional to the average number of opened nodes and the maximum number of stored nodes, respectively [18]. The cumulative

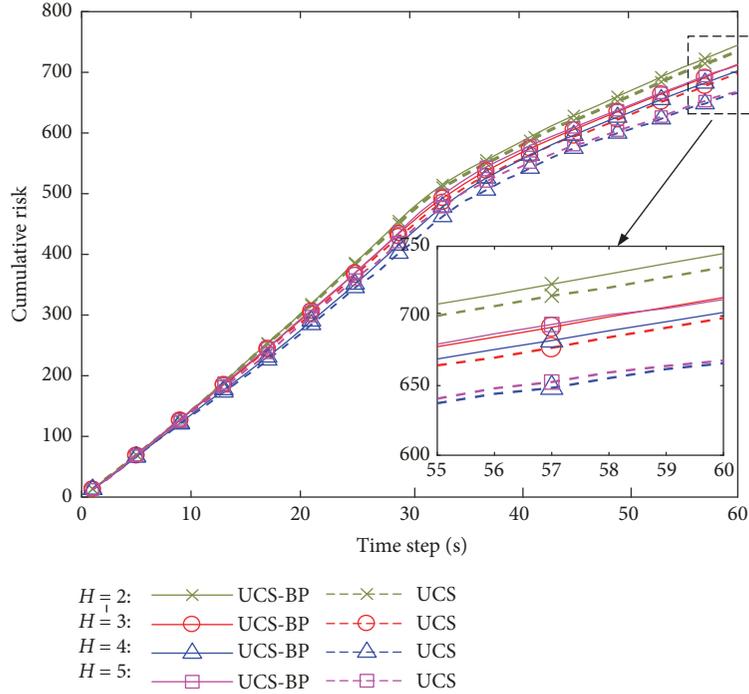


FIGURE 6: Comparisons of the cumulated risks for each algorithm.

total risk can reflect the solution quality of each algorithm.

It can be seen that UCS is able to search for a better solution, so its cumulative total risk is lower than that of UCS-BP. However, compared with UCS, UCS-BP can significantly reduce the number of nodes opened and the maximum number of nodes stored, thus reducing the search time and memory consumption. In addition, it can be seen from Table 2 and Figure 6 that when  $H = 2, 3$ , or  $4$ , the risk decreases with increasing  $H$ . However, the risk rises instead of falling when  $H = 5$ . This result is because the prediction error of the system state will increase with increasing of  $H$ , resulting in an increase in the risk.

Furthermore, there is little difference between the optimal solution and the solution under UCS-BP when  $H = 3$ . Considering the computational complexity and solution quality of the algorithm, we choose  $H = 3$  in the next simulations.

**6.3. Analysis of the Scheduling Method.** Figure 7 shows the time-varying total risk curve of the predicted value and the estimated value. The predicted value is the risk predicted by the belief state before performing a scheduling action, and the estimated value is the risk estimated by the actual measurement after executing a scheduling action. It can be seen that the predicted value is approximately equal to the estimated value the whole time. Consequently, the belief state prediction method can effectively predict the state of the target and the ELI state in future time steps even if the system state is not observable. In addition, the results indicate that it is reasonable to use the predicted risk as the scheduling basis.

Figure 8 shows the projections of the trajectories on the X-Y plane and the optimal scheduling scheme. Figure 9

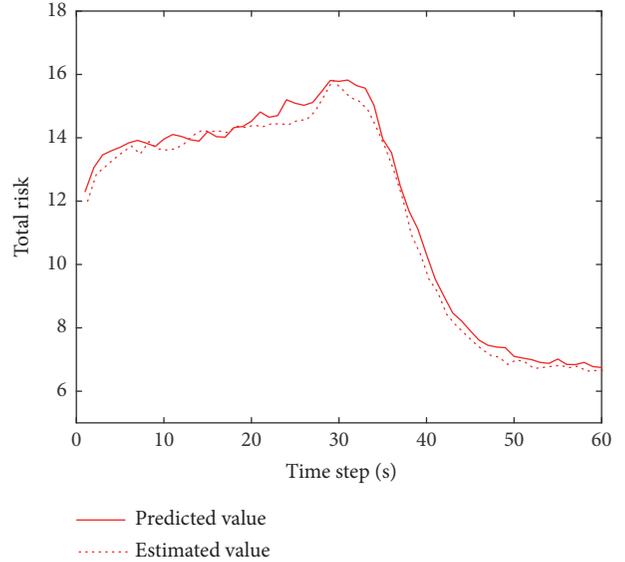


FIGURE 7: Total risk curve.

shows the ratio of the threat assessment risk to the total risk. It can be seen that as target 1 moves close to defence target and target 2 moves away from defence target, the uncertainty in the threat level becomes increasingly small, so the threat assessment risk also becomes low, and the system will pay more attention to controlling the sensor emission. According to Figure 8, we can see that, at the end of simulation time (52-60 s), the system only schedules sensors 1 and 2 with a low emission performance, and the threat assessment risk ratio is less than 10%, which shows that the main factor affecting

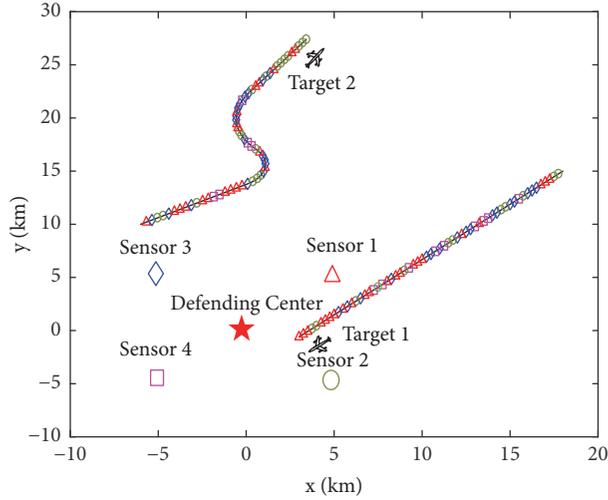


FIGURE 8: Trajectories and sensor scheduling sequences.

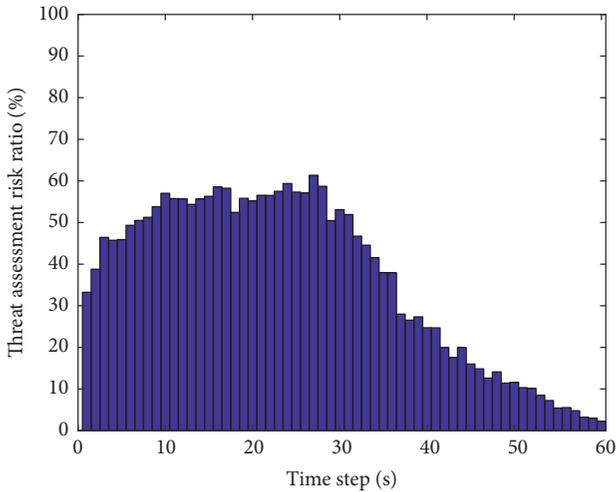


FIGURE 9: Ratio of threat assessment risk to total risk.

decision-making is the emission risk during these time steps. It also reveals the reason why the total risk in Figure 7 tends to be stable after decreasing.

Figure 10 is the threat level error sampling diagram of target 1 to show the uncertainty in the threat assessment. It can be seen that, in the process of threat assessment, the uncertainty in the target state will be transmitted to the threat model, which will generate the corresponding risk in the threat level assessment. From 20 to 35 seconds, the sampling points are widely distributed between both high- and medium-threat levels, and the risk of threat assessment is the highest. As the target gradually approaches the defence target, the distribution of sampling points moves to the high-threat level range, and the uncertainty in the sampling points decreases. When all sampling points are at a high level, the probability of a high-threat level of the target is 1 and the threat assessment risk is 0. At this time, only the emission risk of sensors need to be controlled.

To fully illustrate the effectiveness of the proposed scheduling method (PSM), we choose four existing methods to compare with, namely, (1) the random scheduling method (RSM) where the random sensor combinations are scheduled at each time step, (2) the closest scheduling method (CSM) where the sensors closest to each target are scheduled [28], (3) the fixed scheduling method (FSM) where the fixed sensor combinations are scheduled in the simulation duration (in this paper, we choose to assign sensor 1 and sensor 2 to assess target 1 and 2, respectively) [28], and (4) the myopic scheduling method (MSM) where scheduling plans are decided based on the total risk of the next time step [29].

Figure 11 shows the total risk curve of each method, and Figure 12 compares the normalized risk for the different methods. Since RSM, CSM, and FSM do not predict risk of scheduling, the corresponding total risk is very high. We find that PSM minimizes the three kinds of risks among all methods, indicating that the proposed method in this paper can control the total risk well to improve the security of sensor systems and defence target. The total risk of MSM is lower than those of RSM, CSM, and FSM but higher than that of PSM, which reveals that myopic scheduling can control the risk to some extent by predicting the risk in one step, but cannot obtain the best risk control effect compared with nonmyopic scheduling.

## 7. Conclusions

In this paper, we propose a risk-based multisensor nonmyopic scheduling method that schedules sensors to minimize the uncertainty in the threat level of targets and improve the survivability of sensors. First, a sensor scheduling model based on the POMDP is established, and the state transition law and observation law of target and sensor emissions in the scheduling process are presented. Next, the threat assessment model and the emission model are established, and corresponding risk calculation methods are proposed. Then, a nonmyopic multisensor scheduling objective function is

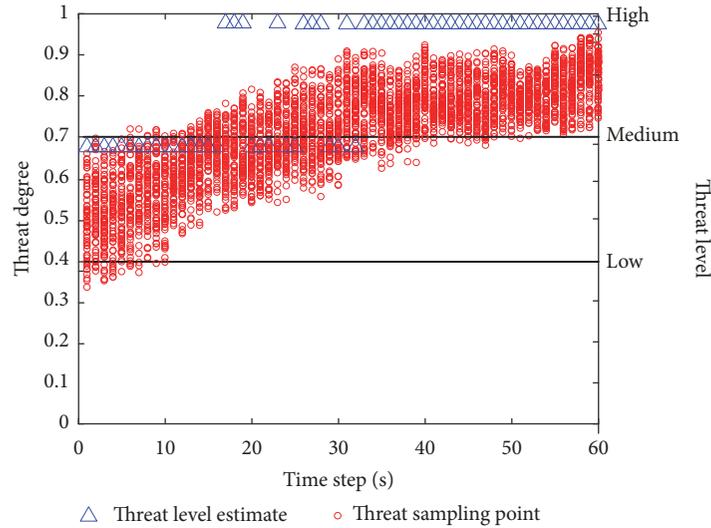


FIGURE 10: Target 1 threat level error sampling.

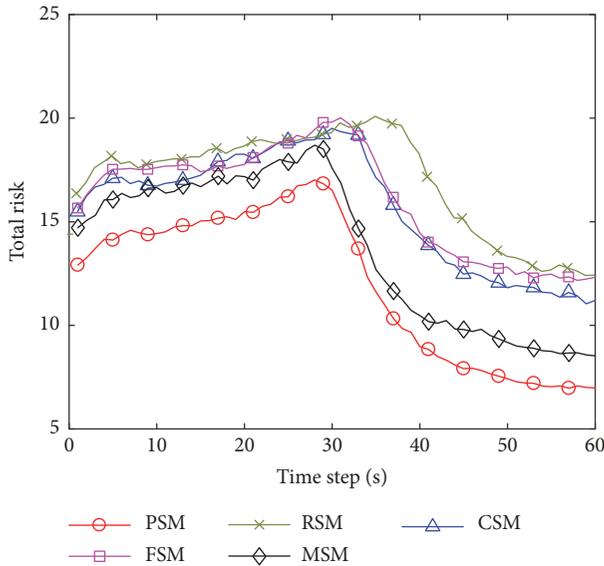


FIGURE 11: Comparison of the total risk at each time step.

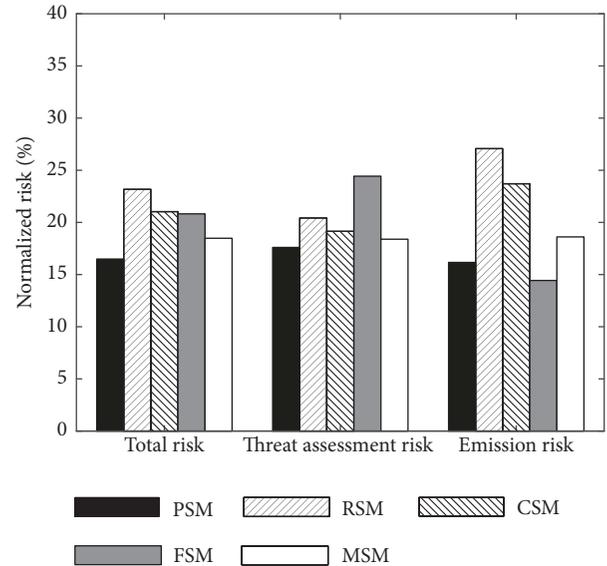


FIGURE 12: Comparison of the normalized risk.

built. Furthermore, to determine the optimal scheduling scheme quickly, an improved decision tree search algorithm, namely, the UCS algorithm based on branch pruning is designed. The simulation results show that the proposed algorithm can obtain high quality solutions quickly and the proposed scheduling method can reasonably predict and control the total risk. In addition, compared with common scheduling methods, the proposed scheduling method can reduce both the threat assessment risk and radiation risk.

A sensor management method that has continuity and relevance for the combination of multiple combat tasks, such as target detection, target recognition, and target tracking, needs to be proposed in future studies.

**Data Availability**

The data used to support the findings of this study are available from the corresponding author upon request.

**Conflicts of Interest**

The authors declare that there are no conflicts of interest regarding the publication of this paper.

**Acknowledgments**

This research was funded by DPRFC (Defense Pre-Research Fund Project of China), grant number 012015012600A2203.

## References

- [1] H. G. Hoang and B. T. Vo, "Sensor management for multi-target tracking via multi-Bernoulli filtering," *Automatica*, vol. 50, no. 4, pp. 1135–1142, 2014.
- [2] A. B. Asghar, S. T. Jawaid, and S. L. Smith, "A complete greedy algorithm for infinite-horizon sensor scheduling," *Automatica*, vol. 81, pp. 335–341, 2017.
- [3] F. Katsilieris, H. Driessen, and A. Yarovoy, "Threat-based sensor management for target tracking," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 51, no. 4, pp. 2772–2785, 2015.
- [4] C. Yang, L. Kaplan, and E. Blasch, "Performance Measures of Covariance and Information Matrices in Resource Management for Target State Estimation," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 48, no. 3, pp. 2594–2613, 2012.
- [5] M. Kalandros and L. Y. Pao, "Controlling target estimate covariance in centralized multi-sensor systems," in *Proceedings of the 1998 American Control Conference (ACC)*, pp. 2749–2753, Philadelphia, PA, USA, 1998.
- [6] M. Kalandros and L. Pao, "Multisensor covariance control strategies for reducing bias effects in interacting target scenarios," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 41, no. 1, pp. 153–173, 2005.
- [7] S. Subedi, Y. D. Zhang, M. G. Amin, and B. Himed, "Cramer–Rao type bounds for sparsity-aware multi-sensor multi-target tracking," *Signal Processing*, vol. 145, pp. 68–77, 2018.
- [8] R. Tharmarasa, T. Kirubarajan, M. Hernandez, and A. Sinha, "PCRLB-based multisensor array management for multitarget tracking," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 43, no. 2, pp. 539–555, 2007.
- [9] Y. Tao and C. Z. Han, "Sensor management for multi-target detection and tracking based on PCRLB," in *Proceedings of the 2017 20th International Conference on Information Fusion (Fusion)*, Xi'an, China, July 2017.
- [10] H. Zhou, M. Taj, and A. Cavallaro, "Target detection and tracking with heterogeneous sensors," *IEEE Journal of Selected Topics in Signal Processing*, vol. 2, pp. 503–513, 2008.
- [11] J. Dias and P. Marques, "Multiple moving target detection and trajectory estimation using a single SAR sensor," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 2, pp. 604–624, 2003.
- [12] J. Lu, Z. Xi, X. Yuan, and M. M. Zhang, "Sensor management for target search with unknown detection performance," in *Multimedia and Signal Processing*, vol. 346, pp. 499–505, 2012.
- [13] A. El-Fallah, A. Zatezalo, R. Mahler, R. K. Mehra, and D. Donatella, "Dynamic sensor management of dispersed and disparate sensors for tracking resident space objects," in *Proceedings of the International Society for Optical Engineering (SPIE)*, vol. 23, pp. 401–402, USA, 2008.
- [14] K. J. Hintz, "A measure of the information gain attributable to cueing," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 21, no. 2, pp. 434–442, 1991.
- [15] M. L. Fowler, M. Chen, and S. Binghamton, "Fisher-information-based data compression for estimation using two sensors," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 41, no. 3, pp. 1131–1137, 2005.
- [16] X. Shen and P. K. Varshney, "Sensor Selection Based on Generalized Information Gain for Target Tracking in Large Sensor Networks," *IEEE Transactions on Signal Processing*, vol. 62, no. 2, pp. 363–375, 2014.
- [17] T. Kangsheng and Z. Guangxi, "Sensor management based on fisher information gain," *Journal of Systems Engineering and Electronics*, vol. 17, no. 3, pp. 531–534, 2006.
- [18] A. S. Chhetri, D. Morrell, and A. Papandreou-Suppappola, "Nonmyopic sensor scheduling and its efficient implementation for target tracking applications," *EURASIP Journal on Applied Signal Processing*, vol. 1, pp. 1–18, 2006.
- [19] M. Xie, J. Hu, S. Guo, and A. Y. Zomaya, "Distributed segment-based anomaly detection with kullback–leibler divergence in wireless sensor networks," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 1, pp. 101–110, 2017.
- [20] X. Liu, C. Zhao, G. Shan, and Y. Wang, "Multi-sensor management based on Rényi information gain for maneuvering target-tracking," *Information and Control*, vol. 44, no. 2, pp. 184–189, 2015.
- [21] A. O. Hero III, B. Ma, O. J. J. Michel, and J. Gorman, "Applications of entropic spanning graphs," *IEEE Signal Processing Magazine*, vol. 19, no. 5, pp. 85–95, 2002.
- [22] C. M. Kreucher, A. O. Hero, and K. Kastella, "A comparison of task driven and information driven sensor management for target tracking," in *Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference, CDC-ECC '05*, pp. 4004–4009, Spain, December 2005.
- [23] E. G. B. Marcos, M. Dominique, and V. Philippe, "A risk-based sensor management using random finite sets and POMDP," in *Proceedings of the 20th International Conference on Information Fusion*, pp. 1588–1596, Xian, China, 2017.
- [24] M. Sean, "Risk-based sensor resource management for field of view constraint sensor," in *Proceedings of the 18th International Conference on Information Fusion*, pp. 2041–2048, Washington, Wash, USA, 2015.
- [25] M. G. Borges, D. Maltese, and P. Vanheeghe, "Sensor management using expected risk reduction approach," in *Proceedings of the 19th International Conference on Information Fusion*, pp. 2050–2058, Heidelberg, Germany, 2016.
- [26] D. Papageorgiou and M. Raykin, "A Risk-Based approach to sensor resource management," in *Advances in Cooperative Control and Optimization*, Springer, Berlin, Germany, 2nd edition, 2007.
- [27] Y. Wang, Hussein II, and R. S. Erwin, "Risk-based sensor management for integrated detection and estimation," in *Proceedings of the 2011 American Control Conference*, pp. 168–189, San Francisco, CA, USA, 2011.
- [28] Z. Zhang and G. Shan, "UTS-based foresight optimization of sensor scheduling for low interception risk tracking," *International Journal of Adaptive Control and Signal Processing*, vol. 28, no. 10, pp. 921–931, 2014.
- [29] G. Xu, C. Pang, X. Duan, and G. Shan, "Multi-sensor optimization scheduling for target tracking based on PCRLB and a novel intercept probability factor," *Electronics*, vol. 8, no. 2, pp. 140–140, 2019.
- [30] C. Shi, F. Wang, S. Salous, and J. Zhou, "Low probability of intercept-based radar waveform design for spectral coexistence of distributed multiple-radar and wireless communication systems in clutter," *Entropy*, vol. 20, no. 3, pp. 197–197, 2018.
- [31] V. Krishnamurthy, "Emission management for low probability intercept sensors in network centric warfare," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 41, no. 1, pp. 133–152, 2005.
- [32] S. Arai, Y. Iwatani, and K. Hashimoto, "Fast Sensor Scheduling for Spatially Distributed Sensors," *IEEE Transactions on Automatic Control*, vol. 56, no. 8, pp. 1900–1905, 2011.

- [33] R. Evans, V. Krishnamurthy, G. Nair, and L. Sciacca, "Networked sensor management and data rate control for tracking maneuvering targets," *IEEE Transactions on Signal Processing*, vol. 53, no. 6, pp. 1979–1991, 2005.
- [34] J. Z. Hare, S. Gupta, and T. A. Wettergren, "POSE: prediction-based opportunistic sensing for energy efficiency in sensor networks using distributed supervisors," *IEEE Transactions on Cybernetics*, vol. 48, no. 7, pp. 2114–2127, 2018.
- [35] G. Xu, X. Duan, and H. Lü, "Target priority determination methods by interval-valued intuitionistic fuzzy sets with unknown attribute weights," *Journal of Shanghai Jiaotong University (Science)*, vol. 22, no. 5, pp. 624–632, 2017.
- [36] X.-X. Liu, L. Zhou, and X.-Y. Du, "Method of sensor management based on target priority and information gain," *Tien Tzu Hsueh Pao/Acta Electronica Sinica*, vol. 33, no. 9, pp. 1683–1687, 2005.
- [37] B. Wu and M. Wu, "Real-time POMDP algorithm based on belief state space compression," *Kongzhi yu Juece/Control and Decision*, vol. 22, no. 12, pp. 1417–1420, 2007.
- [38] I. Arasaratnam and S. Haykin, "Cubature kalman filters," *IEEE Transactions on Automatic Control*, vol. 54, no. 6, pp. 1254–1269, 2009.
- [39] H. Liu and W. Wu, "Interacting multiple model (IMM) fifth-degree spherical simplex-radial cubature kalman filter for maneuvering target tracking," *Sensors*, vol. 17, no. 6, pp. 1374–1374, 2017.
- [40] Z. Zhang and G. Shan, "Non-myopic sensor scheduling to track multiple reactive targets," *IET Signal Processing*, vol. 9, no. 1, pp. 37–47, 2015.

