

Research Article

A Threat Assessment Method for Unmanned Aerial Vehicle Based on Bayesian Networks under the Condition of Small Data Sets

Ruohai Di , Xiaoguang Gao , Zhigao Guo, and Kaifang Wan

School of Electronic and Information, Northwestern Polytechnical University, Xi'an 710129, China

Correspondence should be addressed to Xiaoguang Gao; cxg2012@nwpu.edu.cn

Received 8 September 2017; Revised 17 April 2018; Accepted 9 May 2018; Published 13 June 2018

Academic Editor: Francisco Chicano

Copyright © 2018 Ruohai Di 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 autonomous decision-making of a UAV is based on rapid and accurate threat assessment of the target. Accordingly, modeling of threat assessment under the condition of a small data set is studied in this paper. First, the operational scenario of a manned/unmanned aerial vehicle is constructed, and feature selection and data preprocessing are performed. Second, to obtain the structure, a modeling method for threat assessment is proposed based on an improved BIC score. Finally, the obtained model is applied to compute the threat probability using the junction tree algorithm. The experimental results show that the method proposed in this paper is an available method for threat assessment under the condition of small data sets.

1. Introduction

Threat assessment is an important component of operational decision-making. The complex and varying battlefield environment creates an enormous burden on the decision-maker. To lighten this burden, selected methods for threat assessment were proposed based on machine learning, such as the analytic hierarchy process (AHP) [1], multiple attribute decision-making [2], fuzzy inference [3], rough set [4], neural network [5], support vector machine (SVM), and Bayesian network (BN) methods. These methods display advantages and disadvantages for different problems. Bayesian networks are used in threat assessment in this paper. The Bayesian network is suitable for addressing uncertain information depending on probability theory, is a “white-box” model compared with the neural network and support vector machine approaches, and has the ability to handle small data sets. The Bayesian network was used to construct an air combat threat assessment model in [6], and the junction tree algorithm was applied to obtain the results of inference. Based on the external environmental information, the threat assessment model was built to obtain the threat level of the target in [7]. A dynamic Bayesian network [5] was used to model the threat situation of a battlefield environment, and

the model was applied to a path planning for a UAV. The threat assessment framework [8] relying on the Bayesian networks was proposed, which exploits the expressiveness of first-order logic for semantic relations and the strength of the Bayesian networks in handling uncertainty. The Bayesian network model to assess the public health risk associated with wet weather sewer overflows is presented in [9]. The Bayesian networks [10] are employed to model the worst-case accident scenario and to assess the risks of natural gas station. Reference [11] presents a method of how to integrate and quantify these uncertainties in the warning process by evaluating a tsunami warning level probability distribution with a Bayesian network. The approach [12] for evaluating land-use adaptations was proposed for biofuel production, using Bayesian networks and integrating research on the food, water, and energy sectors. The Bayesian networks mentioned above are built using expertise only and without considering the data. Therefore, the obtained model has strong subjectivity. With the wide application of statistics in machine learning and artificial intelligence, obtaining a Bayesian network from data has become a mainstream method. For structured learning of the Bayesian network, the main research areas are divided into three categories, namely, methods based on the conditional independence test,

methods based on scoring, and a mixed approach using the previously mentioned methods. The typical algorithms of the methods based on the conditional independence test contain the Peter Clark (PC) [13] algorithm and the three-phase dependency analysis (TPDA) [14] algorithm. Certain representative algorithms, such as K2 [15], ant colony optimization (ACO) [16], particle swarm optimization (PSO) [17], ant colony optimization (ACO) [18, 19], and bacterial foraging optimization (BFO) [20], belong to the methods based on score-and-search. The max-min hill climbing (MMHC) [21] and max-min parents and children (MMPC) approaches are algorithms typical of the mixed approach based on the two types of methods described above. Analysis shows that the method based on score-and-search is the most popular method. Maximum likelihood estimation (MLE) can obtain high-quality parameters for parameter learning in the Bayesian network. Unfortunately, the previously mentioned research studies were performed under the conditions of sufficient data and are not applicable for a small data set. The reason for this observation is that both the score of the structure and the maximum likelihood function correlate closely with the statistical information of the data, and small data sets lead to inaccurate statistical information.

Selected work has been conducted on the modeling method using the Bayesian network with a small data set. The primary objective is to introduce the constraint of the structure and the constraint of the parameter into the learning process. For structure learning, the existing structural constraints contain the node order [22], causal link [23, 24], edge existence [25–28], and ancestral constraints [29]. The problem of structure learning is translated into the problem of objective optimization, and the various types of constraints [22, 26, 27] are directly used to limit the search space. Although this method can improve the results of structure learning under the condition of small data sets, the small data set makes the score inaccurate, which causes the score to lose the ability to evaluate the structure of the Bayesian network, which is why the proposed algorithms could not produce satisfactory results. To improve the precision of the score, several research studies have used the structural constraints to obtain the prior distribution of the structure. G. Borboudakis [23, 24] used the edge existence to represent the expert knowledge, which is integrated into the score by means of a nonuniform prior of the structure. The edge existence [30] is used to represent the expert knowledge and the confidence level is used to represent the uncertainty of expert knowledge, and then, the prior distribution of the structure is obtained based on the edge existence and the confidence level. However, there are two fatal shortcomings. One is that the expert knowledge plays determinative roles and is always regarded as completely correct. The other one is that the number of nodes in the structural constraint is no more than 4. If the number is more than 4, the prior distribution is not computed precisely. Therefore, this paper proposes a new method for obtaining the prior distribution for a number of nodes greater than 4. Combined with the new method, the threat assessment model is constructed using the K2 algorithm, and the parameters of the Bayesian network are obtained by the method in [28]. Finally, the threat assessment

is measured using the model learned by the proposed method and the junction tree algorithm.

2. Preliminary

2.1. Cooperative Combat with Manned/Unmanned Aerial Vehicles. Cooperative combat with manned/unmanned aerial vehicles is an important combat mode that combines the advantages of manned aerial vehicles and unmanned aerial vehicles. The division of responsibilities is vital in cooperative combat. In the course of combat, the manned aerial vehicle sends the decision and control information to the unmanned aerial vehicle and commands the unmanned aerial vehicle to finish the combat task. The unmanned aerial vehicle sends the battlefield information obtained by its sensors to the manned aerial vehicle to aid the manned aerial vehicle in making decisions. The unmanned aerial vehicle acts as the “eyes” of the manned aerial vehicle, and the manned aerial vehicle acts as the “brain” of the unmanned aerial vehicle. The collaborative task process is illustrated in Figure 1.

2.2. Bayesian Network. The Bayesian network contains the structure and the parameters. The structure is a directed acyclic graph consisting of nodes and edges. The parameters are a set of conditional probability distributions. For each variable X_j , we denote the parents of X_j as U_j . The parameter is the conditional probability $P(X_j | U_j)$.

If all conditional distributions are specified, then the joint probability can be calculated as follows:

$$P(X_1, X_2, \dots, X_n) = \prod_{i=1}^n P(X_i | U_i). \quad (1)$$

The decomposition of the joint probability makes possible a compact representation of the joint probability and also efficient implementation of probabilistic reasoning.

2.3. Model of Edge Existence in the Bayesian Network. In general, the edge existence is the probability of the edge or arc between two nodes. More concretely, there are two nodes denoted as variables x and y in Bayesian network, and there are three ways to connect x and y . If x is a parent of y , an arc exists from x to y . If y is a parent of x , an arc exists from y to x . If x has nothing to do with y , there is no arc. To express these relationships conveniently, if x is a parent of y , $(x \rightarrow y)$ is used. If y is a parent of x , $(y \rightarrow x)$ is used. If x has nothing to do with y , $(x \cdots y)$ is used. These expressions are obtained by probability theory as formulated in

$$p(x \rightarrow y) + p(y \rightarrow x) + p(x \cdots y) = 1. \quad (2)$$

If $p(x \rightarrow y) = 0.8$, the distribution of the remainder states is uniformed; formula (3) is obtained as follows:

$$p(y \rightarrow x) = p(x \cdots y) = \frac{1 - p(x \rightarrow y)}{2} = 0.1. \quad (3)$$

If there are three nodes in a Bayesian network, the diagram of the edge existence is represented as shown in Figure 2. In

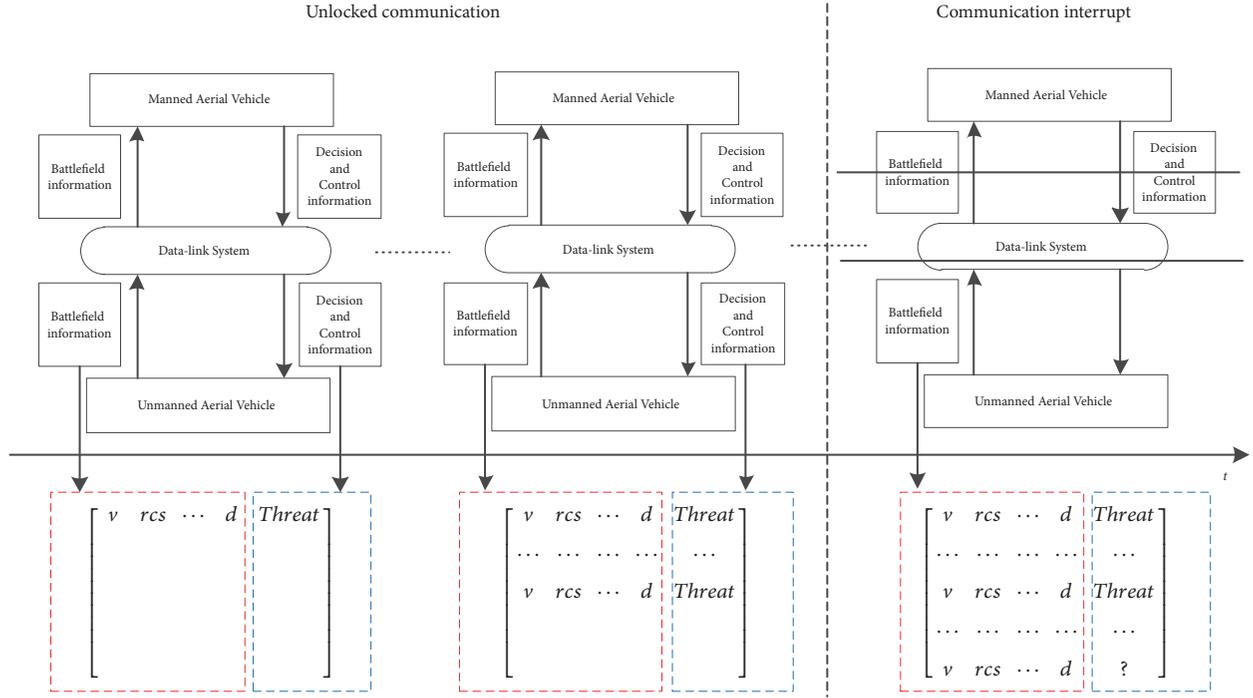


FIGURE 1: Collaborative task process for manned/unmanned aerial vehicles.

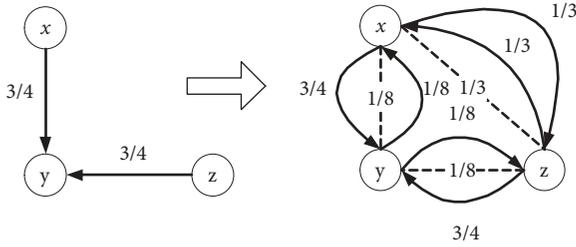


FIGURE 2: Diagram of the edge existence model.

Figure 2, if $p(x \rightarrow y) = 3/4$ and $p(z \rightarrow y) = 3/4$, then $p(y \rightarrow x) = p(x \cdots y) = 1/8$ and $p(y \rightarrow z) = p(z \cdots y) = 1/8$ using the formula (3); if there is no knowledge between x and z , the distribution of the states between x and z is regarded as uniform distribution.

3. Threat Assessment Method Based on Improved BIC Score

3.1. Improved BIC Score. The decomposition of the score is an important property of the scoring function. To understand the decomposition of the score, the BIC score is used as an example in

$$\text{Score}_{\text{BIC}} = \sum_{i=1}^n \sum_{j=1}^{q_i} \sum_{k=1}^{r_i} m_{ijk} \log \frac{m_{ijk}}{m_{ij*}} - \sum_{i=1}^n \frac{q_i (r_i - 1)}{2} \log m \quad (4)$$

where m is the number of data, m_{ijk} is the number of data when the node $X_i = k$ and $\pi(X_i)$ (the parents of X_i) is j , $m_{ij*} = \sum_{k=1}^{r_i} m_{ijk}$, n is the number of all nodes, q_i is the number of the parents' state, and r_i is the number of X_i 's states. The family BIC score is represented by

$$\text{BIC}((X_i, \pi(X_i)) | D) = \sum_{j=1}^{q_i} \sum_{k=1}^{r_i} m_{ijk} \log \frac{m_{ijk}}{m_{ij*}} - \frac{q_i (r_i - 1)}{2} \log m. \quad (5)$$

The BIC score can be represented as shown in

$$\text{Score}_{\text{BIC}} = \sum_{i=1}^n \text{BIC}((X_i, \pi(X_i)) | D). \quad (6)$$

In addition to the BIC score, the CH, AIC, and MDL score all have the property of decomposition. The BIC score is a Laplace approximation based on a large sample. Therefore, if the BIC score is directly used to learn the structure with the small data size, the ideal results are hard to obtain. To obtain a satisfactory structure using the BIC score, an improved BIC score [31] is proposed based on the property of the decomposition and the model of edge existence. The improved BIC score is referred to as the I.BIC score.

$$\text{Score}_{\text{I.BIC}} = \sum_{i=1}^n \left[\sum_{j=1}^{q_i} \sum_{k=1}^{r_i} m_{ijk} \log \frac{m_{ijk}}{m_{ij*}} - \frac{q_i (r_i - 1)}{2} \log m + m \log \left(1 + \frac{\delta e}{m} \right) \right] \quad (7)$$

Algorithm I_BIC_K2(ρ, D, μ, e)
Input: ρ - the order of nodes μ - an upper bound of the number of the parent
 D - data set e - expert constraint for edge existence
Output: G

- (1) Initialize graph G_0 with no edge from variable set V ;
- (2) for $i = 1$ to n
- (3) $Pa(X_i) = \phi$;
- (4) $S_{old} = I_BIC(< X_i, \pi(X_i) > | D)$;
- (5) while $|\pi(X_i)| < \mu$ is true
- (6) $j \leftarrow \arg \max_{1 \leq j < i, X_j \notin Pa(X_i)} BIC(< X_i, \pi(X_i) \cup \{X_j\} > | D)$;
- (7) $S_{new} = I_BIC(< X_i, \pi(X_i) \cup \{X_j\} > | D)$;
- (8) if $S_{new} > S_{old}$
- (9) $S_{old} = S_{new}$;
- (10) $\pi(X_i) = \pi(X_i) \cup \{X_j\}$;
- (11) Add an edge $X_j \rightarrow X_i$ to graph G_0 ;
- (12) else
- (13) break;
- (14) end if
- (15) end while
- (16) end for
- (17) $G = G_0$;
- (18) return G ;

ALGORITHM 1: Flowchart of the I_BIC_K2 algorithm.

In formula (7), e represents the probability of the edge existence given by the domain expert and δ is an adjustable parameter. The range of δ is [10, 100]. With the increase of the data, the term $m \log(1 + \delta e/m)$ makes the significance of data in the score gradually stronger while that of expert knowledge weaker. The term was proposed based on the property $\lim_{x \rightarrow \infty} (1 + 1/x)^x \approx 2.72$ and the decomposability of the BIC score.

3.2. Algorithm of Structure Learning Based on I_BIC Score.

The I_BIC score is provided in Section 3.1. In this section, the I_BIC_K2 algorithm is provided by combining the I_BIC score with the K2 algorithm. The K2 algorithm is a greedy heuristic method. For each node X_i , the algorithm starts by assuming an empty parent set. In each step, the variable preceding X_i whose addition maximally increases the family score is added to the parent set. The search stops when the number of variables in the parent data size set exceeds the predetermined maximum number of the parents or when the addition of variables cannot further increase the score. The details of I_BIC_K2 algorithm are presented in Algorithm 1.

3.3. Algorithm of Parameter Learning Based on Monotonic Constraint Estimation Algorithm. If the parameters of Bayesian network $\theta_{ijk} = p(X_i = k | \pi(X_i) = j)$ satisfy

$$\theta_{ij1} \leq \theta_{ij2} \leq \dots \leq \theta_{ijr} \text{ or } \theta_{ij1} \geq \theta_{ij2} \geq \dots \geq \theta_{ijr}, \quad (8)$$

the θ_{ijk} is considered to be consistent with the monotonic constraint, and then the parameter learning process is given as follows.

Step 1. Establish monotonic constraint based on the expert knowledge.

Step 2. Obtain interval of every parameter using (9); the scope of θ_{ij1} will be obtained first.

$$\begin{cases} \theta_{ij1} + \theta_{ij2} + \dots + \theta_{ijr} = 1 \\ \theta_{ij1} \leq \theta_{ij2} \leq \dots \leq \theta_{ijr} \end{cases} \Rightarrow \begin{cases} 0 \leq \theta_{ij1} \leq \frac{1}{r} \\ \theta_{ij1} \leq \theta_{ij2} \leq \frac{1 - \theta_{ij1}}{r - 1} \\ \vdots \\ \theta_{ijr-2} \leq \theta_{ijr-1} \leq \frac{1 - \sum_{k=1}^{r-2} \theta_{ijk}}{2} \\ \theta_{ijr} = 1 - \sum_{k=1}^{r-1} \theta_{ijk} \end{cases} \quad (9)$$

Step 3. the parameter θ whose scope is given by (9) can be presumed to conform to the uniform distribution $U(\theta_1, \theta_2)$. Then convert uniform distribution into Beta distribution using (10) and obtain virtual sample α_1 and α_2 .

$$\min_{\alpha_1, \alpha_2} \int_0^1 (U(\theta_1, \theta_2) - B(\alpha_1, \alpha_2))^2 d\theta \quad (10)$$

TABLE 1: The Hamming distance with $\delta \in [1, 10]$.

| | $\delta = 1$ | $\delta = 2$ | $\delta = 3$ | $\delta = 4$ | $\delta = 5$ | $\delta = 6$ | $\delta = 7$ | $\delta = 8$ | $\delta = 9$ | $\delta = 10$ |
|------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|---------------|
| 100 | 5.34 | 4.74 | 4.22 | 2.22 | 2.20 | 2.08 | 2.04 | 2.04 | 2.04 | 2.10 |
| 300 | 3.84 | 3.42 | 3.20 | 2.54 | 2.22 | 2.22 | 2.10 | 2.04 | 2.04 | 1.98 |
| 500 | 3.46 | 3.08 | 2.70 | 2.46 | 2.14 | 1.98 | 2.04 | 2.06 | 1.98 | 1.92 |
| 800 | 3.04 | 2.66 | 2.48 | 2.26 | 2.06 | 1.90 | 1.94 | 2.02 | 2.02 | 1.98 |
| 1000 | 2.94 | 2.46 | 2.46 | 2.14 | 2.02 | 1.94 | 1.98 | 1.96 | 1.96 | 1.96 |

TABLE 2: The Hamming distance with $\delta \in [10, 100]$.

| | $\delta = 10$ | $\delta = 20$ | $\delta = 30$ | $\delta = 40$ | $\delta = 50$ | $\delta = 60$ | $\delta = 70$ | $\delta = 80$ | $\delta = 90$ | $\delta = 100$ |
|------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|----------------|
| 100 | 2.12 | 2.12 | 2.06 | 2.12 | 2.18 | 2.22 | 2.30 | 2.48 | 2.54 | 2.82 |
| 300 | 2.02 | 2.06 | 2.02 | 2.04 | 2.14 | 2.08 | 2.14 | 2.40 | 2.30 | 2.72 |
| 500 | 1.98 | 2.00 | 2.02 | 2.04 | 2.10 | 2.06 | 2.16 | 2.26 | 2.22 | 2.46 |
| 800 | 1.96 | 2.02 | 1.96 | 1.96 | 2.02 | 2.02 | 2.08 | 2.16 | 2.26 | 2.34 |
| 1000 | 1.98 | 2.04 | 1.92 | 2.02 | 2.00 | 1.96 | 2.10 | 2.14 | 2.20 | 2.26 |

Step 4. Merge the virtual sample information into Bayesian estimation and get the estimations of parameters:

$$\theta_{ijk} = \frac{\alpha_1 + N_{ijk}}{\alpha_1 + \alpha_2 + N_{ij}} \quad (11)$$

where N_{ijk} is the number of samples when the value of node i is k and the value of its parent is j .

Step 5. Take obtained parameters as lower bound of the next parameter, return to Step 2, and repeat Steps 2-5 until all the parameters are obtained.

4. Experiments

4.1. *Results of Structure Learning Using the I.BIC.K2 Algorithm.* The Asia model is used as the simulation model. The Asia network has 8 variables and 8 edges. To represent the Asia model easily, the nodes are indicated using numbers from one to eight. The model is given in Figure 2. For the evaluation criterion, in the score-and-search paradigm, the aim of structure learning is to determine the maximal scoring value. Therefore, we choose the standard BIC score as the criterion, and the higher the BIC score is, the better the network approximates the distributions of the data sets. At the same time, the Hamming distance is chosen as the other criterion to evaluate the structure, and the smaller the Hamming distance is, the more accurate the structure will be. Formula (4) expresses the standard BIC score, and formula (12) describes the Hamming distance. In formula (12), D_1 represents the number of redundant edges, D_2 represents the number of missing edges, and D_3 represents the number of wrong edges. The experimental data set is generated using Bayes Net Toolbox (BNT) supplied by K. Murphy. Using these data sets and the selected criterion, we perform a comprehensive analysis of the improvements proposed in Section 3. All algorithms are programmed using the Matlab 2010 software package.

$$D_H = D_1 + D_2 + D_3 \quad (12)$$

The Asia model as in Figure 3 is selected as the experimental model. The experiments are divided into three groups: the expert constraint is correct, the expert constraint is incompletely correct, and the expert constraint is wrong. The proposed algorithm is compared with the traditional K2 algorithm under the three conditions. To eliminate error, the experiments are repeated 100 times. The experimental results are presented in Figures 4–12. It is concluded that the proposed method produces more accurate results when the expert constraint is correct or incompletely correct. Particularly, the advantage is quite obvious when the data size is less than 500. As the data size increases, the advantage of the proposed method decreases. The results indicate that the proposed method is slightly weaker than the traditional K2 algorithm. However, with increasing data size, the results from the proposed method approach the results from the traditional K2 algorithm. In other words, the proposed method has good adaptability to the wrong expert constraint.

4.2. *The Influence of δ on the Results of Structure Learning Using the I.BIC.K2 Algorithm.* In order to study the influence of the parameter δ on the results of the structure learning using the I.BIC.K2 algorithm, the Hamming distance is computed using the different δ . The experimental results are presented in Figures 13-14 and Tables 1 and 2. The parameter δ is set to two different discrete intervals such as $[1, 10]$ and $[10, 100]$. Some conclusion can be concluded from the experimental results. First, all the Hamming distances obtained by the proposed method are less than the Hamming distance obtained by K2 algorithm when the parameter $\delta \in [10, 100]$. However, when the parameter $\delta \in [1, 10]$, the Hamming distance obtained by the proposed method is not always less than the Hamming distance obtained by K2 with increase of the data. Second, the results under the condition of $\delta \in [10, 100]$ are better than the results under the condition of $\delta \in [1, 10]$ in most cases. These above are the reason why $\delta \in [10, 100]$ is selected in the paper.

The Asia mode is selected in the experiments. However, the conclusion is of general significance. This is because the

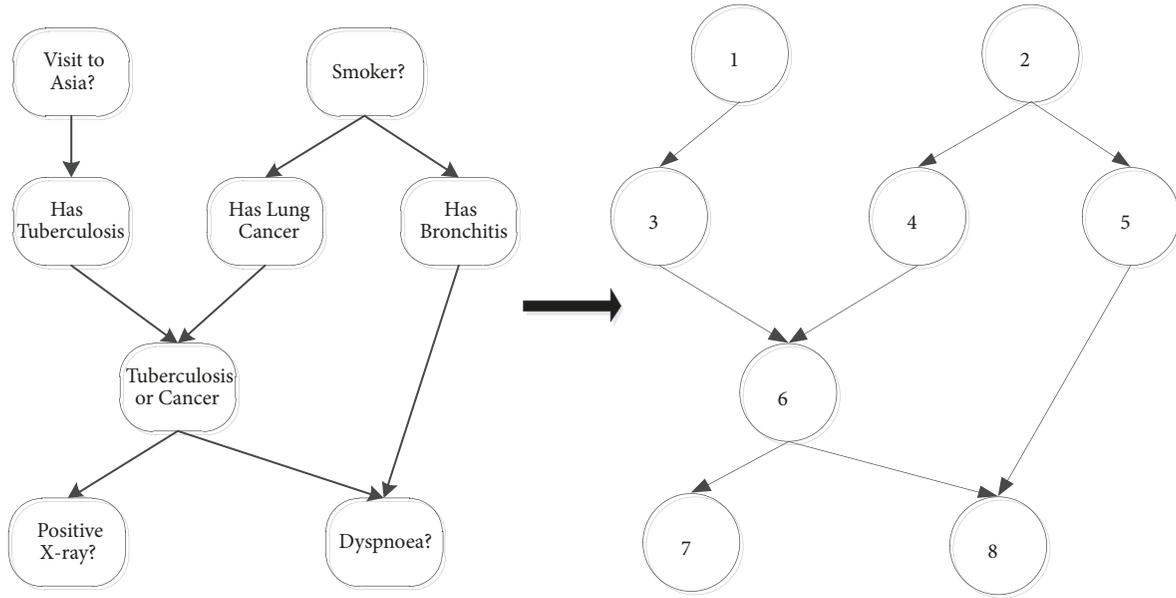


FIGURE 3: Asia model.

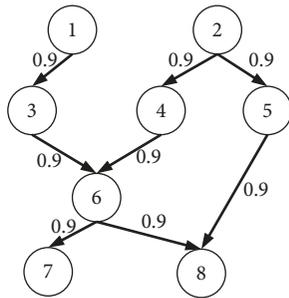


FIGURE 4: Correct expert constraints of Asia.

parameter δ is used to optimize the family BIC score. The selection of δ has little relation with the number of nodes in the whole network. Therefore, the experimental results in Figures 13-14 can be regarded as a guide for selecting the parameter δ . In the paper, $\delta = 10$ is used to compute the BIC score because the results under the condition of $\delta = 10$ are better in most cases.

4.3. Collaborative Threat Assessment by Manned/Unmanned Aerial Vehicles

4.3.1. Simulation Scenario. The battlefield is composed of red and blue groups. The red group contains one manned aircraft and three unmanned aerial vehicles. The blue group contains two missile positions, two flack battlefields, and one radar position. The military task states that the manned aircraft and unmanned aerial vehicle make a cooperative attack on the blue group. At a certain point, the communication between the manned aircraft and unmanned aerial vehicles is interrupted or the manned aircraft is destroyed, and the

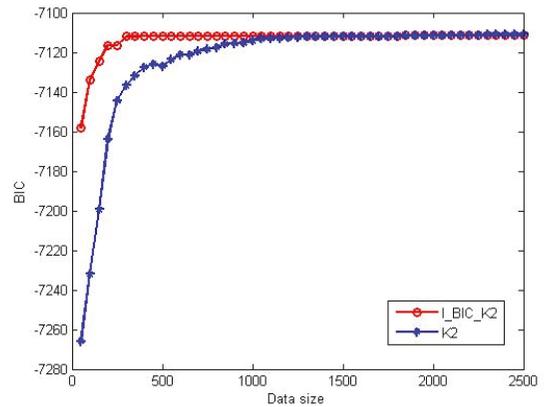


FIGURE 5: Asia's average BIC score comparison with correct constraints.

unmanned aerial vehicle must execute the tasks independently. The combat mission scenario is shown in Figure 15. The parameters of the red group are given in Table 3, and the parameters of the blue group are given in Table 4.

4.3.2. Pretreatment of Threat Factors. In the course of combat, the battlefield information contains information from both sensors and the decision information from the manned aerial vehicle. The battlefield information is too complicated to use these sources directly. For example, certain information is continuous and other information is discrete. Therefore, pretreatment of battlefield information must be applied. The detailed work is described as follows:

- (1) (Sharp, Sp):Sp = {1,2} = {rectangle, column}.

TABLE 3: Parameters of the red group.

| Red group | Longitude (deg) | Latitude (deg) | Height (m) | Velocity (X, Y, Z) (m/s) |
|-----------------------|-----------------|----------------|------------|--------------------------|
| Manned aerial vehicle | 117.85 | 24.66 | 8000 | (0, 0, 278) |
| UAV_1 | 117.9 | 24.7 | 8000 | (0, 0, 280) |
| UAV_2 | 118 | 24.8 | 8000 | (0, 0, 300) |
| UAV_3 | 117.95 | 24.75 | 8000 | (0, 0, 282) |

TABLE 4: Parameters of the blue group.

| Blue group | Longitude (deg) | Latitude (deg) | Height (m) | Anti-attack capability |
|--------------------|-----------------|----------------|------------|------------------------|
| Missile position_1 | 120.96 | 23.30 | 0 | 3 |
| Missile position_2 | 120.97 | 23.27 | 0 | 1 |
| Flack position_1 | 121.16 | 23.27 | 0 | 1 |
| Flack position_2 | 121.16 | 23.30 | 0 | 2 |
| Radar position | 120.76 | 23.28 | 0 | 1 |

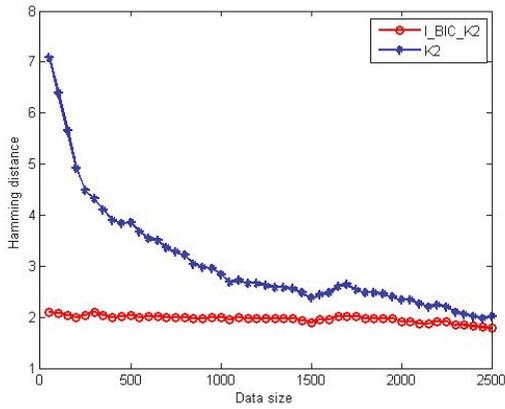


FIGURE 6: Average Hamming distance comparison with correct constraints.

- (2) (Ele-radiation, Er): $Er = \{1,2\} = \{\text{radioactive, nonradioactive}\}$.
- (3) (Size, Si): $Si = \{1,2\} = \{\text{several meters, several centimeters}\}$.
- (4) (Velocity, V): Velocity of target, where V_u represents the velocity of UAV.

$$V = \begin{cases} 1 & V \geq 1.5V_u \\ 2 & 0.6V_u \leq V < 1.5V_u \\ 3 & V < 0.6V_u \end{cases} \quad (13)$$

- (5) (Height, H):

$$H = \begin{cases} 1 & H \geq 3Km \\ 2 & H < 3Km. \end{cases} \quad (14)$$

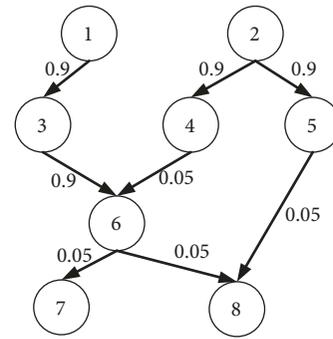


FIGURE 7: Incomplete correct expert constraints of Asia.

- (6) (Angle, A): $A = A_1 + A_2$, where A_1 is the angle between the velocity of target and target line and A_2 is the angle between the velocity of the UAV and the target line.

$$A = \begin{cases} 1 & A < \pi \\ 2 & A \geq \pi \end{cases} \quad (15)$$

- (7) (Value, Vl): $Vl = \{1, 2\} = \{\text{greater value, lesser value}\}$.
- (8) (Distance, Dis):

$$Dis = \begin{cases} 1 & 40km \leq D < 60km \\ 2 & 20km \leq D < 40km \\ 3 & D < 20km. \end{cases} \quad (16)$$

- (9) (Identification friend or foe, IFF): $IFF = \{1, 2\} = \{\text{enemy, friendly}\}$.
- (10) (Target number, Tn): $Tn = \{1, 2\} = \{\text{large, small}\}$.
- (11) (Attack scope, As): $As = \{1, 2, 3\} = \{\text{large, middle, small}\}$.

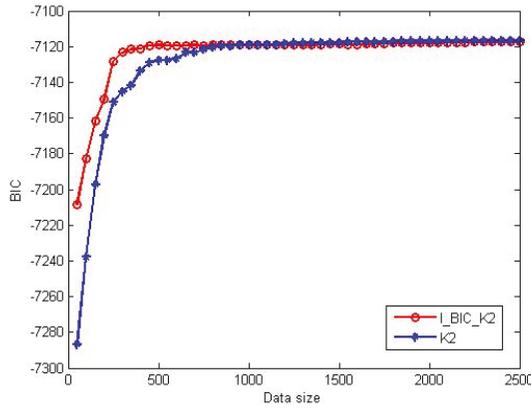


FIGURE 8: Average BIC score comparison with incomplete correct constraints.

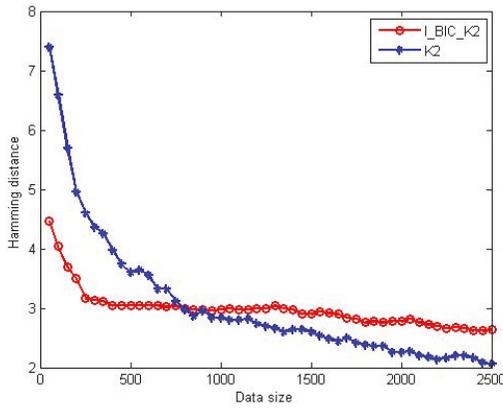


FIGURE 9: Average Hamming distance comparison with incomplete correct constraints.

- (12) (Anti-attack Capability, Aac): $Aac = \{1, 2, 3\} = \{\text{strong, middle, weak}\}$.
- (13) (Target type, Ty): $Ty = \{1, 2, 3\} = \{\text{radar, flack position, missile position}\}$.
- (14) (Attack intent, Ai): $Ai = \{1, 2\} = \{\text{obvious, no obvious}\}$.
- (15) (Attack capability, Ac): $Ac = \{1, 2, 3\} = \{\text{strong, middle, weak}\}$.
- (16) (Threat, T):

$$T = \begin{cases} 1 & \text{locked by missile or flack} \\ 2 & \text{interfered by electromagnetics} \\ 3 & \text{tracked by radar.} \end{cases} \quad (17)$$

4.3.3. *Simulation Conditions and Process.* (1) The simulation environment is Windows 10, and the compilation environment is Matlab 2010.

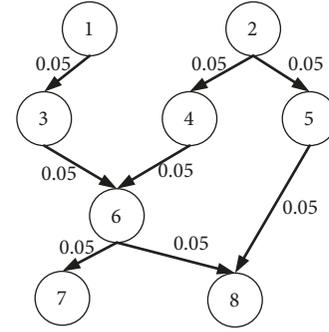


FIGURE 10: Error expert constraints of Asia.

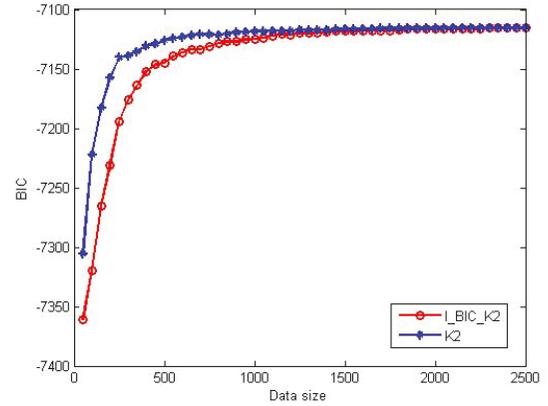


FIGURE 11: Average BIC score comparison with error constraints.

(2) Before the communication interruption occurs, the manned aerial vehicle makes decisions based on the information from the sensors and sends the decision information to the UAV. The UAV completes the mission based on the decision information. All information is handled by the method described in Section 4.3.2. A portion of the processed information is listed in Table 5.

(3) After the communication interruption occurs, the UAV uses the processed information to construct the structure of the threat assessment model using the I.BIC.K2 algorithm and obtains the parameters using the method described in [28].

(4) Under the same condition, the traditional K2 algorithm is used to construct the structure of the threat assessment model, and maximum likelihood estimation is used to learn the parameters of the model.

(5) After obtaining the structure and parameters, the junction tree algorithm is used to compute the probability of the threat. The junction tree algorithm is achieved in the BNT.

4.3.4. *Simulation Results.* To construct the threat assessment model based on the Bayesian network with small data size, the expert knowledge for the edge existence probability is given first. To represent the model more easily, the threat node is represented as node 1, the attack capability node is represented as node 2, the target type node is represented

TABLE 5: Portion of processed information.

| | <i>T</i> | <i>Ac</i> | <i>Ty</i> | <i>Ai</i> | <i>Tn</i> | <i>As</i> | <i>Aac</i> | <i>Sp</i> | <i>Er</i> | <i>Si</i> | <i>V</i> | <i>H</i> | <i>A</i> | <i>VI</i> | <i>Dis</i> | <i>IFF</i> |
|-----------|----------|-----------|-----------|-----------|-----------|-----------|------------|-----------|-----------|-----------|----------|----------|----------|-----------|------------|------------|
| 1 | 2 | 2 | 2 | 1 | 1 | 1 | 2 | 1 | 2 | 2 | 1 | 2 | 1 | 1 | 3 | 1 |
| 2 | 2 | 2 | 2 | 2 | 1 | 3 | 3 | 1 | 2 | 2 | 3 | 1 | 2 | 2 | 3 | 2 |
| 3 | 1 | 2 | 3 | 1 | 2 | 1 | 2 | 2 | 1 | 2 | 3 | 2 | 1 | 1 | 3 | 1 |
| 4 | 3 | 1 | 1 | 1 | 1 | 2 | 2 | 1 | 1 | 1 | 2 | 2 | 1 | 1 | 3 | 1 |
| 5 | 1 | 1 | 2 | 1 | 1 | 1 | 2 | 1 | 2 | 2 | 2 | 1 | 1 | 1 | 3 | 1 |
| 6 | 3 | 2 | 1 | 2 | 1 | 1 | 3 | 1 | 2 | 1 | 1 | 2 | 2 | 2 | 1 | 2 |
| 7 | 2 | 1 | 3 | 1 | 1 | 1 | 1 | 2 | 1 | 2 | 1 | 2 | 1 | 1 | 3 | 1 |
| 8 | 2 | 2 | 3 | 2 | 1 | 2 | 2 | 2 | 1 | 1 | 2 | 1 | 1 | 2 | 1 | 1 |
| 9 | 3 | 3 | 2 | 1 | 1 | 1 | 1 | 1 | 2 | 2 | 1 | 2 | 2 | 1 | 3 | 1 |
| 10 | 1 | 1 | 3 | 1 | 2 | 1 | 3 | 2 | 2 | 2 | 2 | 2 | 1 | 1 | 3 | 1 |

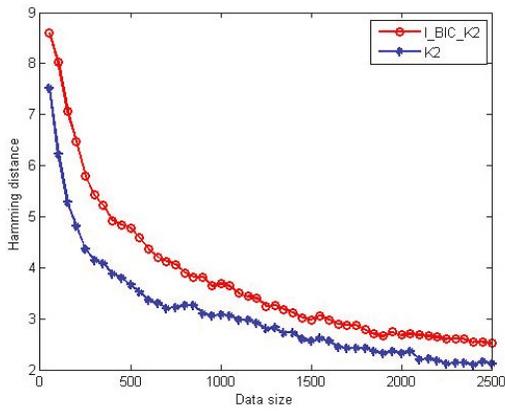


FIGURE 12: Average Hamming distance comparison with error constraints.

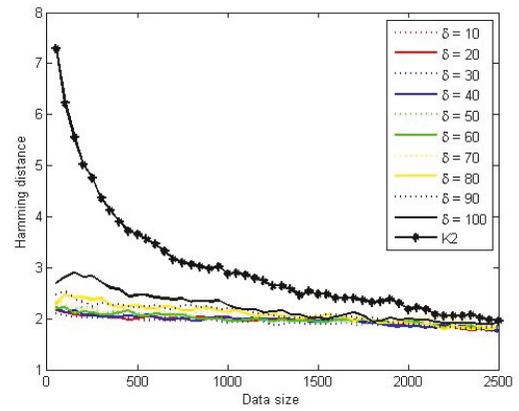


FIGURE 14: The Hamming distance with $\delta \in [10, 100]$.

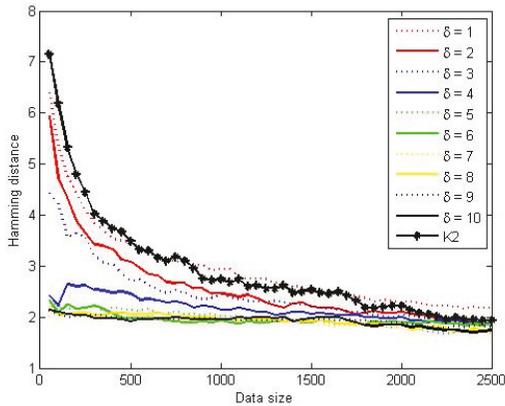


FIGURE 13: The Hamming distance with $\delta \in [1, 10]$.

as node 3, the attack intend node is represented as node 4, the target number node is represented as node 5, the attack scope node is represented as node 6, the anti-attack capability node is represented as node 7, the sharp node is represented as node 8, the ele-radiation node is represented as node 9, the size node is represented as node 10, the velocity node is

represented as node 11, the height node is represented as node 12, the angle node is represented as node 13, the value node is represented as node 14, the distance node is represented as node 15, and the identification of friend or foe node is represented as node 16. The constraint of the edge existence probability is given in

$$\begin{aligned}
 \text{Exis-prob}(1- > [2 \ 3 \ 4]) &= 0.8 \\
 \text{Exis-prob}(2- > [5 \ 6 \ 7]) &= 0.75 \\
 \text{Exis-prob}([2 \ 3 \ 4]- > 11) &= 0.85 \\
 \text{Exis-prob}([3 \ 4]- > 12) &= 0.8 \\
 \text{Exis-prob}(4- > 10) &= 0.05
 \end{aligned} \tag{18}$$

where the $\text{Exis-prob}(4- > 10) = 0.05$ indicates that the probability of the edge from node 4 to node 10 is 0.05. To illustrate the superiority of the proposed method in this paper, for a data size of 3000, the BN structure learned by the traditional K2 algorithm is treated as the right structure. The proposed method is compared with the traditional K2 algorithm for data sizes of 30, 100, 500, and 800.

The experimental results show that the expert constraint is useful in enhancing the accuracy of the structure learning.

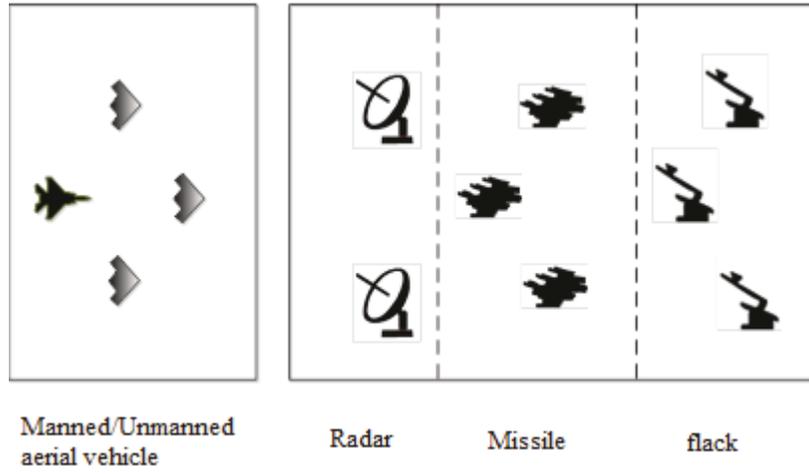


FIGURE 15: Combat mission scenario.

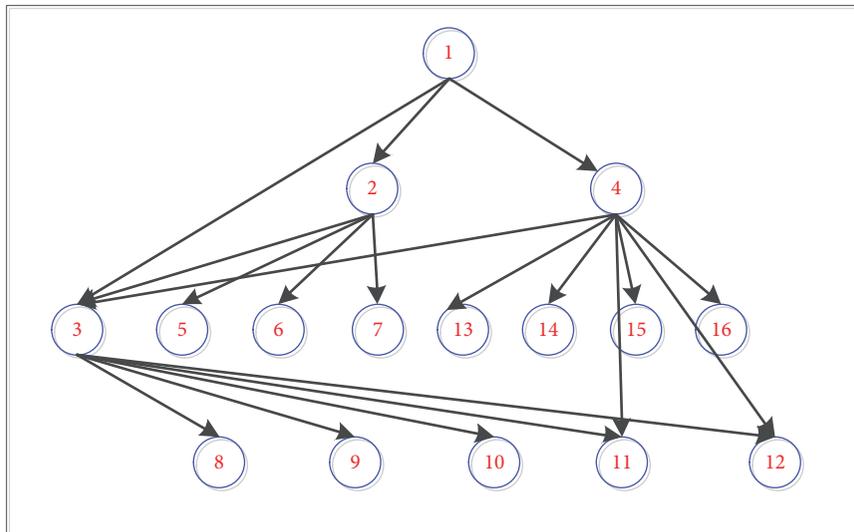


FIGURE 16: Structure learned by the traditional K2 algorithm with 3000 samples.

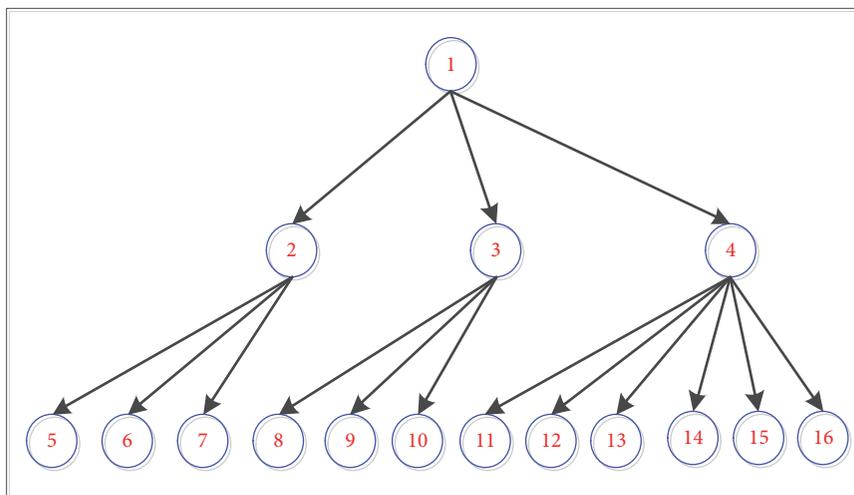


FIGURE 17: Structure learned by the L.BIC.K2 algorithm with 30 samples.

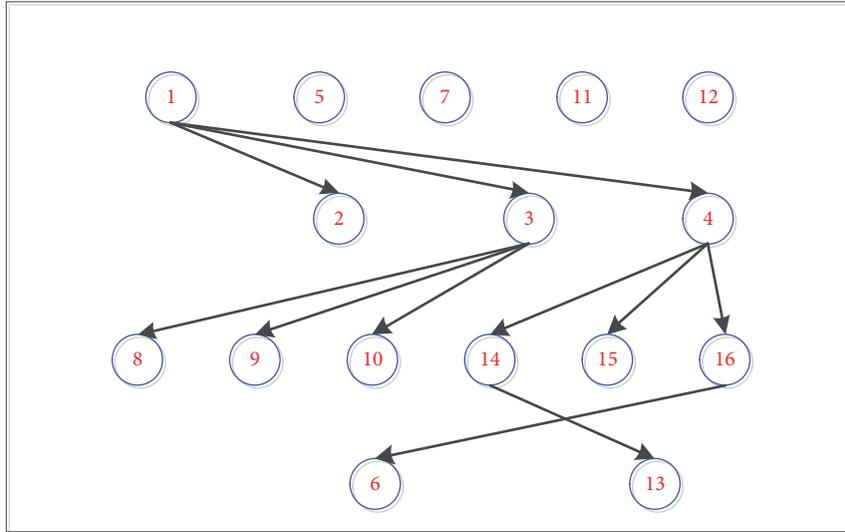


FIGURE 18: Structure learned by the traditional K2 algorithm with 30 samples.

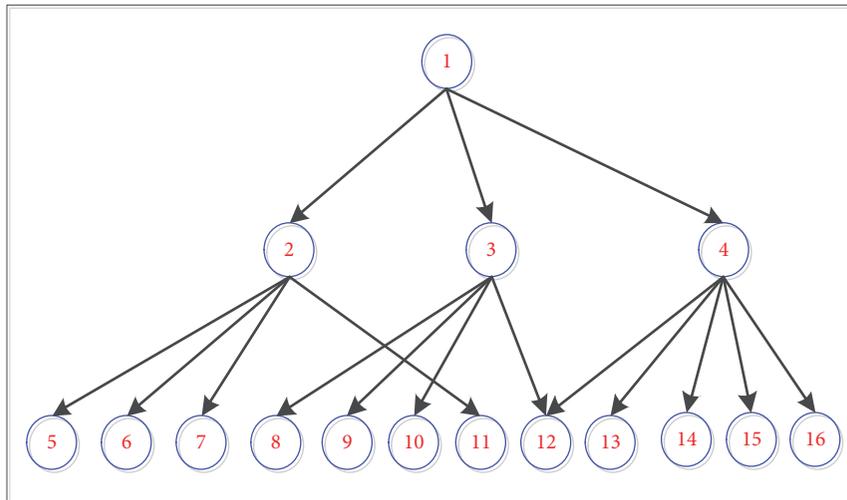


FIGURE 19: Structure learned by the I.BIC.K2 algorithm with 100 samples.

Figure 16 is obtained using the traditional K2 algorithm with 3000 samples, and Figure 16 is treated as the true structure. The experimental results are given in Figures 23 and 24 when the data size is 800. For a data size of 800, the I.BIC.K2 algorithm obtains the same structure as the structure in Figure 16, and the traditional K2 algorithm cannot obtain the accurate structure. The experimental results are given in Figures 17 and 18 when the data size is 30. For a data size of 30, the structure obtained by the I.BIC.K2 algorithm has more edges that are correct than the traditional K2 algorithm. The structure obtained by the I.BIC.K2 algorithm has no edges that are wrong, and the structure obtained by the traditional K2 algorithm has certain edges that are wrong. The experimental results are given in Figures 19–22 when the data size is 100 and 500. For data sizes of 100 and 500, although the structure obtained by the traditional K2 algorithm has no edges that are wrong, it still has fewer edges that are correct

than the structure obtained by the I.BIC.K2 algorithm. In view of the overall situation, the structures obtained by the I.BIC.K2 algorithm and the traditional K2 algorithm both become more accurate with increases in the size of the data. However, the I.BIC.K2 algorithm always obtains a more accurate structure than the traditional K2 algorithm.

After obtaining the structure of the threat assessment model, the parameters of the model must be learned for its application. In this paper, the monotonic constraint estimation (MCE) algorithm in [28] and the maximum likelihood estimation (MLE) are used to learn the parameters. The partial parameters are given in Tables 6–11. The 3000-MLE case represents the results of MLE with 3000 samples. The 100-MCE case represents the results of MCE with 100 samples. The 100-MLE case represents the results of MLE with 100 samples. As mentioned in [28], the MCE algorithm is superior to the MLE under the condition of small data size.

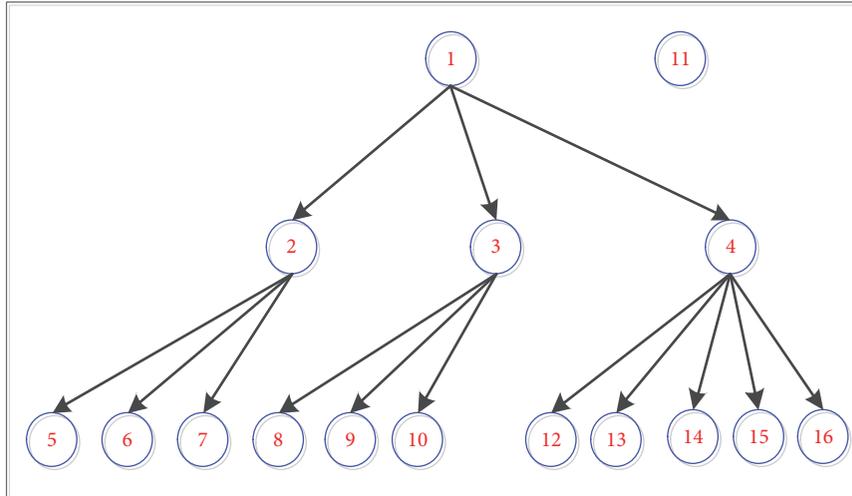


FIGURE 20: Structure learned by the traditional K2 algorithm with 100 samples.

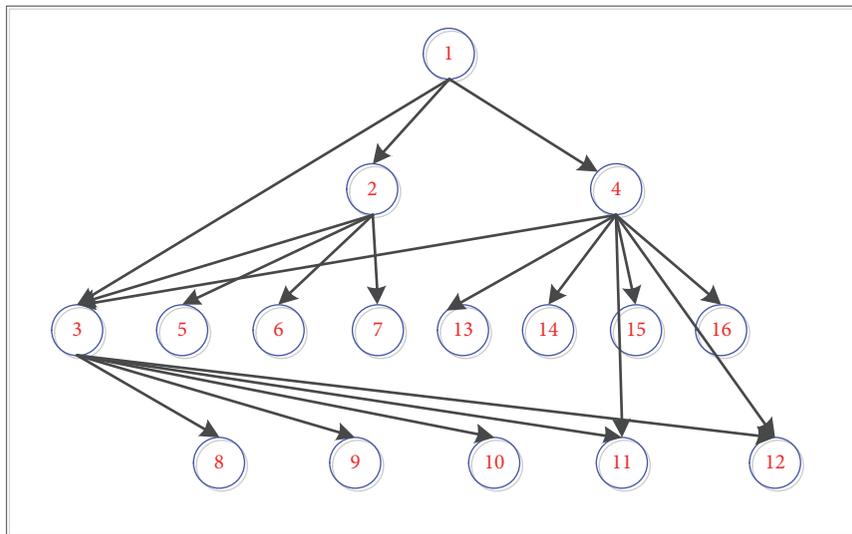


FIGURE 21: Structure learned by the I.BIC.K2 algorithm with 500 samples.

Therefore, the MCE algorithm is chosen as the algorithm for parameter learning in this paper.

After the structure and the parameters are obtained, the threat assessment model is achieved and is applied to assess the threat probability. When evidence is given in terms of IFF=1, Tn=1, Sp=2, Er=1, Si=2, Dis=3, Al=1, A=1, V=1, H=1, Dc=1, and As=1, the junction tree algorithm is used to compute the posterior probability of the threat, as shown in Figure 25. In Figure 25, the standard value is the reasoning result based on the threat assessment, which is constructed using the traditional method with 3000 samples. The other two results are both obtained with 500 samples. The reasoning result of the proposed method is closer to the standard value than the reasoning result of the traditional method. Therefore, the proposed method can construct a more accurate model with small data size.

In order to verify the feasibility of our method, the proposed method is compared with multiattribute decision-making algorithm for threat assessment. The multiattribute decision-making algorithm is one of the most widely used methods for threat assessment. The target attribute contains target type, velocity, distance, angle, and target value for the multiattribute decision-making algorithm in the simulation experiment. In the course of combat, the target parameters at some point of time are given in Table 12. At the moment, the multiattribute decision-making algorithm is used to compute the threat value of the targets. The results are shown in Figure 26. Meanwhile, the proposed method in the paper is used to get the threat value. The results are shown in Figure 27. As you can see in Figures 26 and 27, the results of the two methods are basically consistent. For example, when the threat value is greater than 0.5 in Figure 26, the

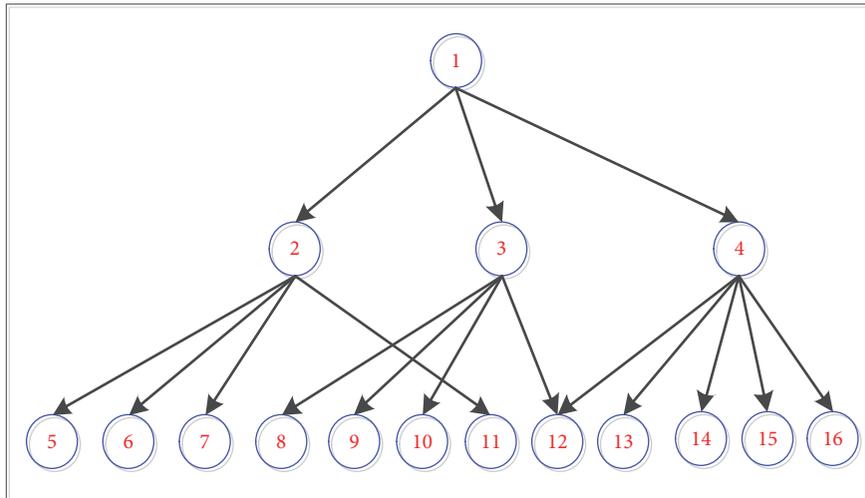


FIGURE 22: Structure learned by the traditional K2 algorithm with 500 samples.

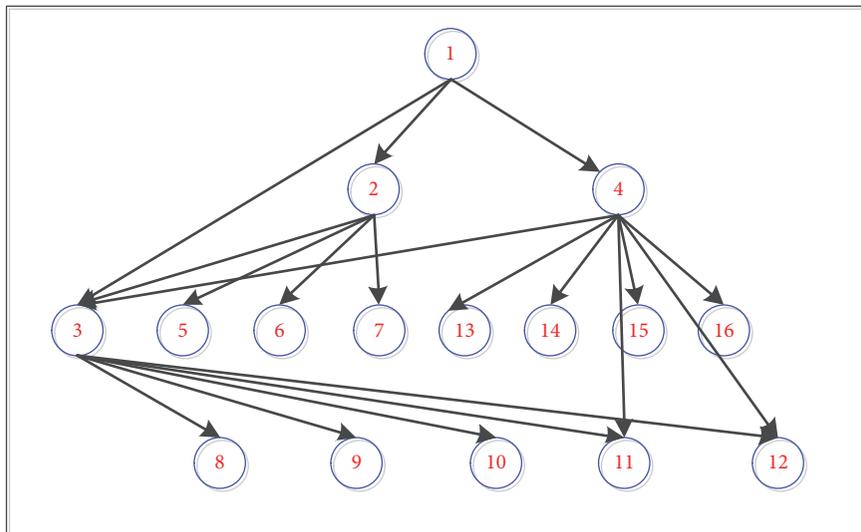


FIGURE 23: Structure learned by the I.BIC.K2 algorithm with 800 samples.

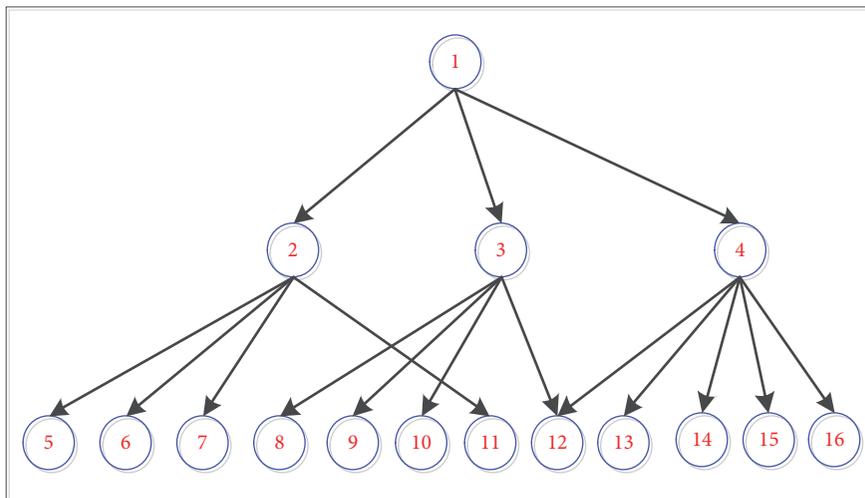


FIGURE 24: Structure learned by the traditional K2 algorithm with 800 samples.

TABLE 6: Parameters of node Ac learned by 100 samples.

| | 3000-MLE | 100-MLE | 100-MCE |
|-------------|--------------------------|--------------------------|--------------------------|
| T | Ac(1, 2, 3) | Ac(1, 2, 3) | Ac(1, 2, 3) |
| T1 (strong) | (0.8102, 0.1399, 0.0499) | (0.7500, 0.2500, 0) | (0.7921, 0.1308, 0.0771) |
| T2 (middle) | (0.3027, 0.4038, 0.2935) | (0.1795, 0.3846, 0.4359) | (0.2627, 0.4122, 0.3252) |
| T3 (weak) | (0.2482, 0.3411, 0.4107) | (0.2927, 0.1707, 0.5366) | (0.2398, 0.3322, 0.4279) |

TABLE 7: Parameters of node Ai learned by 100 samples.

| | 3000-MLE | 100-MLE | 100-MCE |
|-------------|------------------|---------------------|------------------|
| T | Ai(1, 2) | Ai(1, 2) | Ai(1, 2) |
| T1 (strong) | (0.9024, 0.0976) | (0.9500, 0.0500) | (0.9333, 0.0667) |
| T2 (middle) | (0.5520, 0.4480) | (0, 0.5641, 0.4359) | (0.5731, 0.4269) |
| T3 (middle) | (0.2604, 0.7396) | (0.2683, 0.7317) | (0.2725, 0.7275) |

TABLE 8: Parameters of node V learned by 100 samples.

| | | | 3000-MLE | 100-MLE | 100-MCE |
|-------------|-------------|----------|--------------------------|--------------------------|---------------------------|
| Ac(1, 2, 3) | Ty(1, 2, 3) | Ai(1, 2) | V(1, 2, 3) | V(1, 2, 3) | V(1, 2, 3) |
| 1 | 1 | 1 | (0.1036, 0.2091, 0.6873) | (0, 0.3333, 0.6667) | (0.0981, 0.2135, 0.6884) |
| 1 | 1 | 2 | (0.1652, 0.4986, 0.3362) | (0, 0.6000, 0.4000) | (0.0753, 0.4437, 0.4810) |
| 1 | 2 | 1 | (0.3115, 0.3242, 0.3643) | (0, 0.6667, 0.3333) | (0.2375, 0.4423, 0.3202) |
| 1 | 2 | 2 | (0.0129, 0.1357, 0.8514) | (0.3333, 0.3333, 0.3333) | (0.0910, 0.2042, 0.7048) |
| 1 | 3 | 1 | (0.0529, 0.1453, 0.8018) | (0.2500, 0.1667, 0.5833) | (0.1071, 0.1512, 0.7417) |
| 1 | 3 | 2 | (0.4004, 0.2945, 0.3050) | (0.1250, 0.5000, 0.3750) | (0.4443, 0.3100, 0.2457) |
| 2 | 1 | 1 | (0.0522, 0.1494, 0.7984) | (0, 0.2500, 0.7500) | (0.0333, 0.1451, 0.8216) |
| 2 | 1 | 2 | (0.7551, 0.2110, 0.0340) | (0.7500, 0, 0.2500) | (0.7007, 0.2177, 0.0816) |
| 2 | 2 | 1 | (0.8070, 0.1814, 0.0116) | (0.8000, 0.2000, 0) | (0.7805, 0.1718, 0.0477) |
| 2 | 2 | 2 | (0.0079, 0.0880, 0.9042) | (0, 0.2500, 0.7500) | (0.0258, 0.1497, 0.8245) |
| 2 | 3 | 1 | (0.0554, 0.1121, 0.8325) | (0, 0, 1.0) | (0.0362, 0.1527, 0.8111) |
| 2 | 3 | 2 | (0.2352, 0.3737, 0.3912) | (0.5000, 0.5000, 0) | (0.2587, 0.3361, 0.4052) |
| 3 | 1 | 1 | (0.3241, 0.5621, 0.1138) | (0.3333, 0.3333, 0.3333) | (0.3030, 0.5827, 0.1144) |
| 3 | 1 | 2 | (0.5580, 0.2853, 0.1568) | (0.7333, 0.1333, 0.1333) | (0.5797, 0.2805, 0.13981) |
| 3 | 2 | 1 | (0.5889, 0.2480, 0.1631) | (0.4444, 0.3333, 0.2222) | (0.5311, 0.2518, 0.2171) |
| 3 | 2 | 2 | (0.4282, 0.3757, 0.1961) | (1.000, 0, 0) | (0.5594, 0.2651, 0.1755) |
| 3 | 3 | 1 | (0.2715, 0.3737, 0.3548) | (0.2222, 0.3337, 0.4444) | (0.2533, 0.3028, 0.4439) |
| 3 | 3 | 2 | (0.2096, 0.2227, 0.5677) | (0, 0, 1.000) | (0.1721, 0.2751, 0.5527) |

TABLE 9: Parameters learned by 500 samples.

| | 3000-MLE | 500-MLE | 500-MCE |
|-------------|--------------------------|--------------------------|--------------------------|
| T | Ac(1, 2, 3) | Ac(1, 2, 3) | Ac(1, 2, 3) |
| T1 (strong) | (0.8102, 0.1399, 0.0499) | (0.8393, 0.1250, 0.0357) | (0.8172, 0.1381, 0.0447) |
| T2 (middle) | (0.3027, 0.4038, 0.2935) | (0.2552, 0.3906, 0.3542) | (0.2498, 0.4132, 0.3370) |
| T3 (weak) | (0.2482, 0.3411, 0.4107) | (0.2449, 0.3061, 0.4490) | (0.2409, 0.3081, 0.4510) |

TABLE 10: Parameters learned by 500 samples.

| | 3000-MLE | 500-MLE | 500-MCE |
|-------------|------------------|---------------------|------------------|
| T | Ai(1, 2) | Ai(1, 2) | Ai(1, 2) |
| T1 (strong) | (0.9024, 0.0976) | (0.9286, 0.0714) | (0.9213, 0.0787) |
| T2 (middle) | (0.5520, 0.4480) | (0, 0.5573, 0.4427) | (0.5625, 0.4375) |
| T3 (middle) | (0.2604, 0.7396) | (0.2704, 0.7296) | (0.2692, 0.7308) |

TABLE II: Parameters learned by 500 samples.

| Ac(1, 2, 3) | Ty(1, 2, 3) | Ai(1, 2) | 3000-MLE V(1, 2, 3) | 500-MLE V(1, 2, 3) | 500-MCE V(1, 2, 3) |
|-------------|-------------|----------|--------------------------|--------------------------|--------------------------|
| 1 | 1 | 1 | (0.1036, 0.2091, 0.6873) | (0.0667, 0.2667, 0.6667) | (0.0973, 0.2759, 0.6268) |
| 1 | 1 | 2 | (0.1652, 0.4986, 0.3362) | (0, 0.6000, 0.4000) | (0.0655, 0.4321, 0.5024) |
| 1 | 2 | 1 | (0.3115, 0.3242, 0.3643) | (0.2083, 0.2917, 0.5000) | (0.1945, 0.4583, 0.3471) |
| 1 | 2 | 2 | (0.0129, 0.1357, 0.8514) | (0.0714, 0.2500, 0.6786) | (0.1024, 0.2595, 0.6381) |
| 1 | 3 | 1 | (0.0529, 0.1453, 0.8018) | (0.1084, 0.1325, 0.759) | (0.1148, 0.1597, 0.7255) |
| 1 | 3 | 2 | (0.4004, 0.2945, 0.3050) | (0.2308, 0.3846, 0.3846) | (0.3346, 0.3330, 0.3324) |
| 2 | 1 | 1 | (0.0522, 0.1494, 0.7984) | (0, 0.2069, 0.7931) | (0.0488, 0.2200, 0.7311) |
| 2 | 1 | 2 | (0.7551, 0.2110, 0.0340) | (0.7727, 0.1818, 0.0455) | (0.7058, 0.2166, 0.0776) |
| 2 | 2 | 1 | (0.8070, 0.1814, 0.0116) | (0.8148, 0.1852, 0) | (0.7504, 0.2097, 0.0399) |
| 2 | 2 | 2 | (0.0079, 0.0880, 0.9042) | (0, 0.1212, 0.8788) | (0.0370, 0.2031, 0.7599) |
| 2 | 3 | 1 | (0.0554, 0.1121, 0.8325) | (0, 0.1176, 0.8824) | (0.0512, 0.1786, 0.7702) |
| 2 | 3 | 2 | (0.2352, 0.3737, 0.3912) | (0.3810, 0.4286, 0.1905) | (0.3150, 0.3294, 0.3556) |
| 3 | 1 | 1 | (0.3241, 0.5621, 0.1138) | (0.5000, 0.3750, 0.1250) | (0.3218, 0.5278, 0.1504) |
| 3 | 1 | 2 | (0.5580, 0.2853, 0.1568) | (0.5915, 0.2817, 0.1268) | (0.5868, 0.2798, 0.1334) |
| 3 | 2 | 1 | (0.5889, 0.2480, 0.1631) | (0.5806, 0.1290, 0.2903) | (0.4461, 0.2953, 0.2586) |
| 3 | 2 | 2 | (0.4282, 0.3757, 0.1961) | (0.6364, 0.1818, 0.1818) | (0.5444, 0.2777, 0.1780) |
| 3 | 3 | 1 | (0.2715, 0.3737, 0.3548) | (0.3333, 0.1667, 0.5000) | (0.2948, 0.3175, 0.3877) |
| 3 | 3 | 2 | (0.2096, 0.2227, 0.5677) | (0.0667, 0.2667, 0.6667) | (0.2009, 0.2825, 0.5166) |

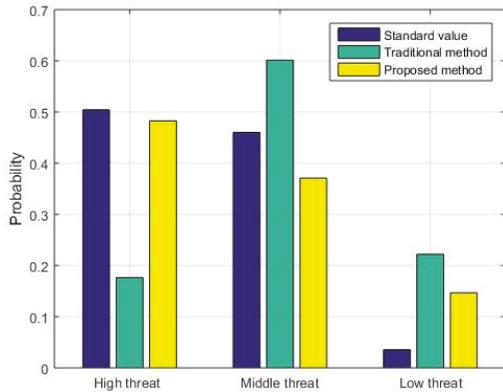


FIGURE 25: Reasoning result based on the threat assessment model.

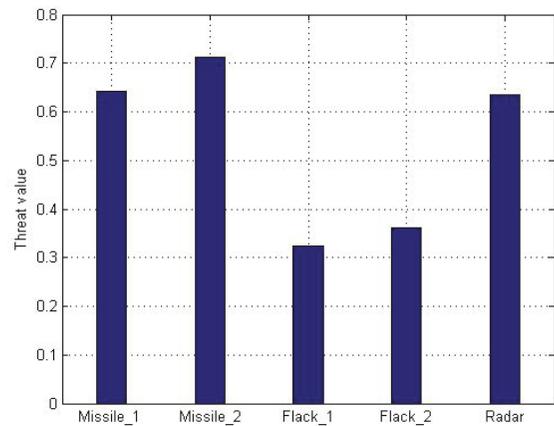


FIGURE 26: Threat value of the multiattribute decision-making algorithm.

TABLE 12: Target parameters.

| Target type | Relative position(m) | velocity (m/s) |
|--------------------|----------------------|----------------|
| Missile position_1 | (1000,0,400) | (0,0,0) |
| Missile position_2 | (1700,0,1200) | (0,0,15) |
| Flack position_1 | (2000,0,700) | (0,0,0) |
| Flack position_2 | (1500,0,1800) | (0,0,0) |
| Radar position | (600,0,2300) | (0,0,10) |

corresponding target has the high threat probability. When the threat value is less than 0.5 in Figure 26, the corresponding target has the low threat probability.

5. Conclusions

Under the condition of small data size, a method for constructing a threat assessment model based on the Bayesian network is proposed. The threat assessment model is built by the expert knowledge and data. Compared with the traditional method, the proposed method can obtain a more accurate threat assessment model with help of the expert knowledge. And the method for learning Bayesian network in the paper has the adaptability to the error expert knowledge. The research in this paper offers a feasible method for constructing the threat assessment model with small data

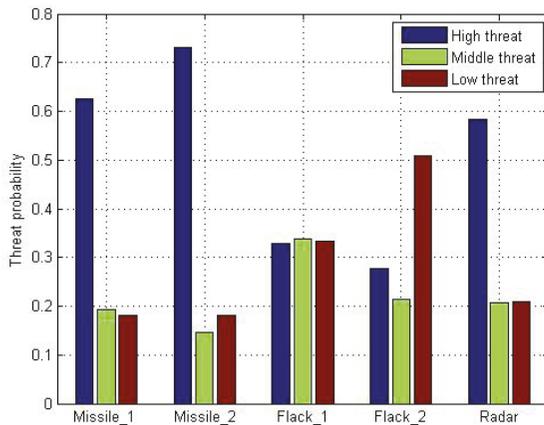


FIGURE 27: Threat probability of the proposed method.

size. Additionally, this research provides a new approach for modeling the problem in other areas.

Some of the future research directions are (i) to combine the proposed score with other search algorithms, (ii) to analyze the relationship between the parameter δ and the data size and give the way to select the parameter δ , and (iii) to get the optimal node ordering using some algorithms.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The authors are grateful to Professor Gao and Dr. Guo for discussions. This study was supported by the National Natural Science Foundation of China (Grants nos. 61305133 and 61573285).

References

- [1] K. C. Zhang, X. Z. Zhu, Q. Feng, and K. Zhang, "Visualization threat assessment for air combat based on interval-radar chart," *Systems Engineering and Electronics*, vol. 38, no. 5, pp. 1052–1058, 2016 (Chinese).
- [2] Y. Wang, Y. S. Liu, W. Zhang, and Y. N. Wang, "Threat assessment method with uncertain attribute weight based on intuitionistic fuzzy anti-attribute decision," *Acta Electronica Sinica*, vol. 12, pp. 2509–2514, 2014 (Chinese).
- [3] F. Mata, L. Martínez, and E. Herrera-Viedma, "An adaptive consensus support model for group decision-making problems in a multigranular fuzzy linguistic context," *IEEE Transactions on Fuzzy Systems*, vol. 17, no. 2, pp. 279–290, 2009.
- [4] J.-Q. Wang, K.-J. Li, and H.-Y. Zhang, "Interval-valued intuitionistic fuzzy multi-criteria decision-making approach based on prospect score function," *Knowledge-Based Systems*, vol. 27, pp. 119–125, 2012.
- [5] H. Lee, B. J. Choi, C. O. Kim, J. S. Kim, and J. E. Kim, "Threat evaluation of enemy air fighters via neural network-based Markov chain modeling," *Knowledge-Based Systems*, vol. 116, pp. 49–57, 2017.
- [6] A. Benavoli, B. Ristic, A. Farina, M. Oxenham, and L. Chisci, "An application of evidential networks to threat assessment," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 45, no. 2, pp. 620–639, 2009.
- [7] M. Cen, Y. Guo, and K. Lu, "Bayesian Network Based Threat Assessment Method for Vehicle," *Journal of Computers*, vol. 7, no. 7, pp. 252–258, 2012.
- [8] K. Golestan, B. Khaleghi, F. Karray, and M. S. Kamel, "Attention Assist: A High-Level Information Fusion Framework for Situation and Threat Assessment in Vehicular Ad Hoc Networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 5, pp. 1271–1285, 2016.
- [9] R. Goulding, N. Jayasuriya, and E. Horan, "A Bayesian network model to assess the public health risk associated with wet weather sewer overflows discharging into waterways," *Water Research*, vol. 46, no. 16, pp. 4933–4940, 2012.
- [10] E. Zarei, A. Azadeh, N. Khakzad, M. M. Aliabadi, and I. Mohammadfam, "Dynamic safety assessment of natural gas stations using Bayesian network," *Journal of Hazardous Materials*, vol. 321, pp. 830–840, 2017.
- [11] L. Blaser, M. Ohrnberger, C. Riggelsen, A. Babeyko, and F. Scherbaum, "Bayesian networks for tsunami early warning," *Geophysical Journal International*, vol. 185, no. 3, pp. 1431–1443, 2011.
- [12] C. Peter, W. De Lange, J. K. Musango, K. April, and A. Potgieter, "Applying Bayesian modelling to assess climate change effects on biofuel production," *Climate Research*, vol. 40, no. 2-3, pp. 249–260, 2009.
- [13] P. Spirtes and C. Meek, "Learning bayesian networks from data: an information-theory based approach," *Artificial Intelligence*, vol. 137, no. 1, pp. 43–90, 2002.
- [14] J. Cheng, D. Bell, and W. Liu, "Learning belief networks from data: an information-theory based approach," in *Proceedings of the 6th International Conference on Information and Knowledge Management*, pp. 325–331, 1997.
- [15] G. F. Cooper and E. Herskovits, "A Bayesian method for the induction of probabilistic networks from data," *Machine Learning*, vol. 9, no. 4, pp. 309–347, 1992.
- [16] J. Ji, R. Hu, H. Zhang, and C. Liu, "A hybrid method for learning Bayesian networks based on ant colony optimization," *Applied Soft Computing*, vol. 11, no. 4, pp. 3373–3384, 2011.
- [17] T. Wang and J. Yang, "A heuristic method for learning Bayesian networks using discrete particle swarm optimization," *Knowledge and Information Systems*, vol. 24, no. 2, pp. 269–281, 2010.
- [18] J. Ji, H. Wei, and C. Liu, "An artificial bee colony algorithm for learning Bayesian networks," *Soft Computing*, vol. 17, no. 6, pp. 983–994, 2013.
- [19] P. C. Pinto, A. Nagele, M. Dejori, T. A. Runkler, and J. M. C. Sousa, "Using a local discovery ant algorithm for Bayesian network structure learning," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 4, pp. 767–779, 2014.
- [20] J. Ji, C. Yang, J. Liu, J. Liu, and B. Yin, "A comparative study on swarm intelligence for structure learning of Bayesian networks," *Soft Computing*, vol. 21, no. 22, pp. 1–26, 2017.
- [21] I. Tsamardinou, L. E. Brown, and C. F. Aliferis, "The max-min hill climbing Bayesian network structure learning algorithm," *Machine Learning*, vol. 65, no. 1, pp. 31–78, 2006.

- [22] M. M. Zhu, S. Y. Liu, and C. F. Wang, "An optimization approach for structural learning Bayesian networks based on prior node ordering," *Zidonghua Xuebao/Acta Automatica Sinica*, vol. 37, no. 12, pp. 1514–1519, 2011.
- [23] G. Borboudakis and I. Tsamardinos, "Incorporating Causal Prior Knowledge as Path Constraints in Bayesian Networks and Maximal Ancestral Graphs," in *Proceedings of the 29th International Conference on Machine Learning (ICML12)*, pp. 1799–1806, Omnipress, 2012.
- [24] G. Borboudakis and I. Tsamardinos, "Scoring and searching over Bayesian networks with causal and associative priors," in *Proceedings of the 29th Conference on Uncertainty in Artificial Intelligence*, pp. 1210–1232, AUAI Press, July 2013.
- [25] C. P. de Campos and Q. Ji, "Efficient structure learning of Bayesian networks using constraints," *Journal of Machine Learning Research (JMLR)*, vol. 12, no. 2, pp. 663–689, 2011.
- [26] C. Andres, R. Andres, and Masegosa, "A method for integrating expert knowledge when learning bayesian networks from data," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 41, no. 5, pp. 1382–1394, 2011.
- [27] S. Isci, H. Dogan, C. Ozturk, and H. H. Otu, "Bayesian network prior: Network analysis of biological data using external knowledge," *Bioinformatics*, vol. 30, no. 6, pp. 860–867, 2014.
- [28] R.-H. Di, X.-G. Gao, and Z.-G. Guo, "The modeling method with bayesian networks and its application in the threat assessment under small data sets," *Acta Electronica Sinica*, vol. 44, no. 6, pp. 1504–1511, 2016.
- [29] Y. J. Chen, Y. Shen, A. Choi, and A. Darwiche, "Learning Bayesian Networks with ancestral constraints," in *Proceedings of the 30th Conference on Neural Information Processing Systems*, pp. 2321–2335, Barcelona, Spain, 2016.
- [30] H. Amirkhani, M. Rahmati, P. J. F. Lucas, and A. Hommersom, "Exploiting experts' knowledge for structure learning of bayesian networks," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 99, no. 11, pp. 2154–2170, 2017.
- [31] H. R. Di, G. Gao, and Z. G. Guo, "Bayesian networks structure learning based on improved BIC scoring," *Systems Engineering and Electronics*, vol. 39, no. 2, pp. 437–444, 2017.



Hindawi

Submit your manuscripts at
www.hindawi.com

