International Journal for Housing Science and Its Applications
Publish August 10, 2025. Volume 46, Issue 4 Pages 510-523

https://doi.org/10.70517/ijhsa46446

# Research on Multi-source Data Fusion and Security Situational Awareness Technology for Rolling Stock Networks

**Kaizhou Xiong**[1,2,*]
[1] Department of Postgraduate, China Academy of Railway Sciences, Beijing, 100081, China
[2] Locomotive & Car Research Institute, China Academy of Railway Sciences Co. Ltd., Beijing, 100081, China
Corresponding authors: (e-mail: xiongkaizhou@foxmail.com).

**Abstract** The security of the rolling stock network is directly related to the stability and security of the railroad transportation system. With the continuous development of the rolling stock network, the network security problem is becoming more and more prominent. Based on this, this paper proposes a security situational awareness model based on multi-source data fusion. First, based on the network topology, host information and alarm information, the fusion framework of multi-source heterogeneous data is established. Second, the network security posture is evaluated and predicted by applying algorithms such as Bayesian network and Kalman filter. The experimental results show that the model can effectively improve the accuracy of security posture assessment when dealing with multi-source data. By comparing the detection results of different methods, the model proposed in this paper shows high accuracy and low false detection rate in a variety of network attack scenarios, especially in the types of attacks such as privacy data stealing and network bandwidth consumption, the recognition effect is most significant. The experimental data show that the proposed method has an error of less than 0.03 during the evaluation process and has good real-time performance and stability. Therefore, the security situational awareness method based on this model can provide more accurate security protection support for the rolling stock network.

**Index Terms** Rolling stock network, security situational awareness, multi-source data fusion, network security, Bayesian network, Kalman filtering

## I. Introduction

With the progress of science and technology, the problems of low bandwidth and slow rate of traditional train communication network are gradually exposed, which is difficult to meet the needs of the new era of rolling stock [1]. In 2014, the International Electrotechnical Commission released the standards related to Ethernet control of trains, which opens the research on Ethernet control of trains [2]. In 2018, the feasibility of Ethernet-controlled vehicles was demonstrated through the application verification of the Fuxing train set with a speed of 250 km/h [3]. In 2019, the Beijing-Zhangzhou High-Speed Railway, an intelligent high-speed railroad with a speed of 350 km/h, was officially opened for operation, also using Ethernet technology [4]. In 2022, the research related to the Ethernet network control system for high-speed rolling stock with a speed of 400 km/h was started, and in the Intelligent Rolling Stock System Architecture 2.0, the Ethernet control was taken as one of the key tasks of intelligent rolling stock, and it was pointed out that multi-source fusion technology would be one of the research focuses of the next-generation rolling stock [5].

Modern high-speed locomotives usually use train network control management system to complete the control, monitoring and diagnosis and protection tasks of the vehicle [6], [7]. The train network control management system, as the nerve center of the train set, can control and manage various subsystems of the train, such as traction, braking and doors, etc. The control and signal acquisition of most of the vehicle subsystems are summarized to the electrical cabinets of each car through hardwires [8]-[10]. Then, the mutual communication with the train network is accomplished through input-output modules (IOMs) [11]. Therefore, the function of the train network input-output module is mainly to control the field devices while receiving their feedback signals, and whether its communication is normal or not is directly related to the monitoring and control of each vehicle by the central control unit [12]-[15]. At the same time, rolling stock is an important part of railroad transportation equipment, so it is of great practical significance to carry out a scientific and reasonable network security posture assessment of rolling stock to find out the existence of security risks to reduce or even eliminate the security risks [16].

With the acceleration of informationization process, rolling stock network, as the core component of modern railroad system, has gradually become a complex multi-level network environment. The train set network not only includes the operation and control system of the train set itself, but also covers the related on-board equipment,

track equipment and ground monitoring system and other diversified information sources. Its complex network structure and massive data flow make the network security problem increasingly serious, especially in the face of various network attacks, system failures and data leakage and other security threats, the existing security protection measures can not provide real-time, comprehensive security.

In order to cope with this challenge, security situational awareness technology has emerged. This technology is able to monitor network status and predict threats by acquiring multi-source data in the network environment in real time and combining data fusion, anomaly detection, risk assessment and other means. However, most of the current research focuses on the security situation assessment in a single data source or local environment, and lacks the technical breakthrough of effective fusion of heterogeneous data from multiple sources.

In this paper, we propose a network security situational awareness model for rolling stock based on multi-source data fusion. The model evaluates and predicts the security posture of the network by collecting and fusing network information from different levels, including network topology, host status, alarm data, etc., and utilizing algorithms such as Bayesian networks and Kalman filtering. Compared with the traditional single data source method, the proposed method in this paper can better reflect the network security status, discover potential threats and make early warnings in time. Especially when dealing with complex and dynamic security threats, the model shows strong adaptability and accuracy.

The goal of this study is to improve the existing security situational awareness method by introducing multi-source data fusion technology, with a view to providing more comprehensive and accurate technical support for the security protection of the rolling stock network. Through a large number of experiments and comparative analyses, the study verifies the effectiveness and feasibility of the method in practical application, and provides theoretical basis and technical guarantee for the security management of rolling stock network.

## II. Security posture assessment of data utilization in the rolling stock network

### *II. A.Network Security Situational Awareness System Framework for Rolling Stock*

Multi-source information layer, multi-source heterogeneous network security data through the network's key equipment Snort intrusion detection system in the acquisition. Level 1 metrics include network topology information, host information, and alarm information. Among the host information second-level indicators include host weights, host vulnerabilities and vulnerability static severity, services and service weights.

As the middle layer of network security situational awareness, network security posture assessment has the role of top and bottom. The assessment layer provides historical data for cybersecurity posture prediction by collecting security elements from the information layer and quantifying them into specific values [17].

The network security posture prediction layer predicts the state of security development in the future period of time based on the analysis results of the security posture assessment in the previous layer of the network. Managers can make early warnings and take the initiative to make reasonable decisions in a timely manner according to the prediction and analysis results, and take some corresponding measures to avoid losses.

Figure 1 for the network security situational awareness model, security situational awareness system operation mainly has four links: situational awareness elements of information extraction, situational assessment (information fusion situational analysis), situational change prediction, visualization situational analysis. This paper focuses on the fusion of heterogeneous data from multiple sources, i.e., situational assessment.

In the analysis and processing of multi-source heterogeneous fusion information, there have been many academic researchers to combine a variety of new fusion analysis algorithms with each other to carry out research, such as the combination of Bayesian networks and attack graphs, Bayesian networks and rough set theory, the combination of Bayesian networks and Kalman filtering, this algorithm combined research will also be a future trend of academic research [18]. For the optimization of data fusion algorithms in time and space complexity is also a research trend. The current posture value in the network security posture assessment is derived from the fusion of multiple perspectives of the indicator elements, but lacks the calculation of modular posture value. The study of modularized situational values is important for network managers to make timely decisions.

### *II. B.Application of multi-source data fusion technology in network security posture assessment*

#### II. B. 1) Data fusion techniques

Data fusion is the core of cybersecurity posture assessment and runs through all stages of posture assessment. Data fusion technology is to combine data from multiple information sources, and by analyzing these multi-source heterogeneous data, valuable data information needed for further decision analysis is obtained, which is more accurate compared with single data source information.

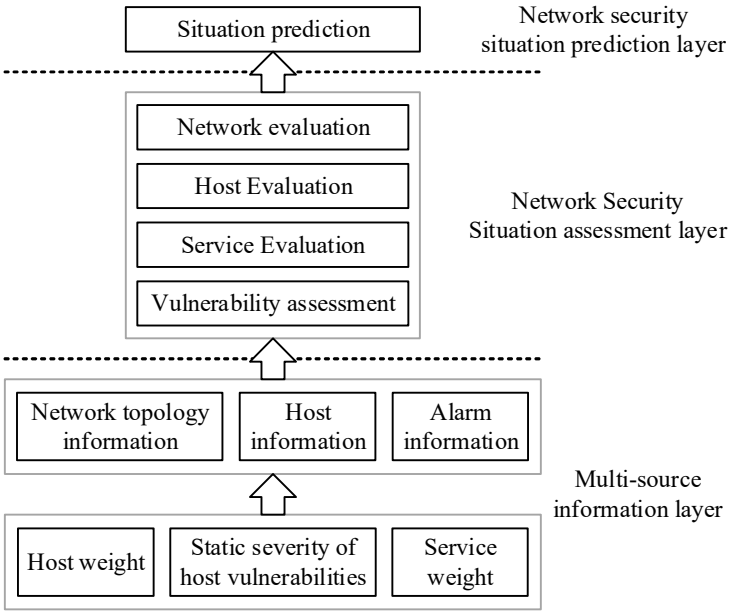



Figure 1: Network Security Situation Awareness Model

(1) Data fusion hierarchy

According to the hierarchical differences, data fusion is divided into three categories from the level of fusion in order from low to high: data-level based fusion, feature-level based fusion and decision-level based fusion.

(2) JDL data fusion model

Over the past three decades, with the in-depth research and continuous exploration of data fusion technology by many scholars, its application field has further increased, and the proposed data fusion model has been up to dozens of models. Among them, the most far-reaching is the proposed JDL data fusion model. Many researchers have modified and improved the JDL data fusion model on this basis. Some scholars have made large modifications to the JDL data fusion model and can be applied in network security posture assessment.

#### II. B. 2) Cybersecurity posture assessment models

Network security posture assessment method is to assess by level, the idea of the method is the complex process of posture assessment hierarchical processing, according to the actual network environment, the network system is divided into four layers of attack/vulnerability, services, hosts and systems, to intrusion detection system provides alarm information and vulnerability information as a baseline, combined with the network environment information, the use of “bottom-up, first local evaluation and then overall evaluation” approach to the comprehensive evaluation of the network security situation to provide network administrators with clear, intuitive posture assessment results. Using the alarm information and vulnerability information provided by the intrusion detection system as the benchmark, combined with the network environment information, it adopts the “bottom-up, first local evaluation and then overall evaluation” approach to comprehensively evaluate the security status of the network, providing network administrators with clear and intuitive posture assessment results.

#### II. B. 3) Cybersecurity posture assessment model based on data fusion

Based on summarizing the existing literature, this paper proposes a hierarchical network security posture assessment model [19]. The model is divided into three levels, namely, posture element extraction layer, host posture analysis layer and network posture analysis layer, and is mainly realized in two parts: firstly, the posture indicators obtained from the posture element extraction layer are fused to obtain the security posture values of all host nodes in the network, and secondly, the posture values of all host nodes in the network are fused to obtain the security posture values of the overall network.

## III. Network security situational awareness model based on multi-source data fusion

### *III. A. Multi-source data fusion*

Among the methods of data fusion, the D-S evidence fusion method is more widely used, which can combine and process the same events in multiple data sources, so as to get the overall probability of occurrence of the event [20]. However, in the real network environment, different security sensors have different detection degrees and

detection strengths, and may have different detection credibility for the same security events, so it is counterintuitive to directly perform D-S evidence fusion on the data. In this regard, the paper utilizes the combination of ant colony algorithm and D-S fusion rules to deal with the multi-source data fusion problem, and the core is to determine the fusion credibility weights for different data through the optimization-seeking ability of ant colony algorithm.

In the ant colony algorithm (ACO), if the optimization process contains $M$ ants, each ant needs to evaluate a $k$-dimensional data source combination (i.e., a path) in one iteration, where $k$ denotes the number of data sources, and $A_k^m$ is labeled as the $m$ th ant evaluating a $k$-dimensional data source combination. Based on the paths they search to determine the fusion order of data sources in the data processing process of the dataset data, the fusion priority of the data sources can be determined based on the probability calculated in equation (1):

$$P_{A_i^*}^{I_i}(t)=\begin{cases}\dfrac{\tau_{I_i}(t)^{\alpha}\cdot\eta_{I_i}^{\beta}}{\sum\limits_{I_i\in J_{i_i}<I_i}(\tau_{I_i}(t)^{\alpha}\cdot\eta_{I_i}^{\beta})}, I_i\in J_{A_i^*}(t)\\ 0, otherwise\end{cases} \tag{1}$$

where $I_i$ denotes a data source that is combined as a whole into a single dataset, $\tau_{I_i}(t)$ denotes the pheromone left over from $I_i$ being selected at the $t$ th iteration, $\eta_{I_i}$ is the $I_i$-related heuristic information, which can be determined by a priori knowledge, $\alpha$ and $\beta$ denote the pheromone factor and heuristic factor, respectively, which are generally specified by the user, or the default value of 1 can be selected, and $J_{A_i^*}(t)$ denotes the set of datasources that have not yet been selected.

The selection of datasets has a one-time constraint and cannot be used reproducibly. In order to prevent premature death and local optimal solutions, random selection is performed using equation (2):

$$P_{A_i^*}^{I_i}(t)=\begin{cases}1, I_i=rand(J_{A_i^*}(t))\\ 0, otherwise\end{cases} \tag{2}$$

Each ant chooses these two probability functions randomly with a certain frequency $\theta$, i.e., the frequency of choosing the first probability function is $\theta$, and the frequency of choosing the second probability function is $1-\theta$. This strategy helps to realize the diversity of the search and increase the likelihood of the algorithm obtaining the global optimal solution, thus determining the problem of the order of selection of the data set and the corresponding weighting problem.

Through iterative processing, the optimal solution is solved, which contains the selection probability of the data source, according to which the importance of the data source can be determined as the expression $(w_1,w_2,\cdots,w_n)$.

Based on the above solution, the D-S evidence combination rule is improved:

$$m(A)=\frac{\sum\limits_{A_1\cap A_2\cap\cdots\cap A_n=A} m_1(A_1)^{w_1}m_2(A_2)^{w_2}\cdots m_n(A_n)^{w_n}}{1-K} \tag{3}$$

where $K=\sum\limits_{A_1\cap A_2\cap\cdots\cap A_n=\varnothing} m_1(A_1)^{w_1}m_2(A_2)^{w_2}\cdots m_n(A_n)^{w_n}$, $A\neq\varnothing$.

### *III. B. Network security situational awareness based on multi-source data fusion*

#### III. B. 1) Feature selection

Feature extraction is an important stage of data fusion that provides information based on the text, such as the maximum and minimum term frequencies for each document. Relevant features are selected and the range of influence on machine learning is determined. In addition, the ability to extract data from the training model is a good feature.

TF-IDF is a well known technique which is used as a weighting technique and its performance is still comparable even with newer techniques. Documents are considered as factors in the term weighted values. Feature selection process is considered as the main preprocessing process required to index the documents.

TF-IDF is one of the most efficient methods for calculating term weights.TF-IDF is equal to TF multiplied by IDF, where TF is an acronym for Term Frequency used to calculate the descriptive power of the term: IDF is an acronym for Inverse Document Frequency used to calculate the discriminative power of the term:

$$IDF = \log \frac{N}{n} \tag{4}$$

where, $N$ is the total number of texts in all categories, and $n$ is the number of texts containing the term $t$.

Term Frequency (TF) indicates the frequency of occurrence of the term (keyword) in the text, this number is usually normalized to prevent it from biasing long documents with the formula:

$$tf_{ij} = \frac{n_{i,j}}{\sum_k n_{k,j}} \tag{5}$$

where, $n_{i,j}$ is the frequency of the word $t_i$ in the text $j$, and $\sum_k n_{k,j}$ is the total of all vocabularies in the document $j$.

That is:

$$TF_w = \frac{\text{The number of times the given word } w \text{ appears in the current article}}{\text{The total word count in the current article}} \tag{6}$$

TF is generally computed online at the time of word splitting in applications.

IDF denotes the inverse document rate, defined as the total number of documents divided by the number of documents containing the occurrence of a given word, which is given by:

$$idf_i = \log \frac{|D|}{|1 + \{j : t_i \in d_i\}|} \tag{7}$$

where, $|D|$ is the total number of documents in the corpus $d$, and $|\{j : t_i \in d_i\}|$ is the number of documents in the corpus $d$ that contain the word $t_i$ in document $j$.

To avoid a denominator of 0, add 1 to the denominator bit. That is:

$$IDF = \log \frac{\text{The number of corpus documents}}{\text{The number of documents containing the word } w} \tag{8}$$

TF-IDF denotes the inverse document frequency, which is defined as:

$$TF - IDF_{ij} \rightarrow tf_{i,j} \times idf_i = \frac{n_{ij}}{\sum_k n_{kj}} \times \log \frac{|D|}{|1 + \{j : t_i \in d_i\}|} \tag{9}$$

TF-IDF can determine whether the word occurs frequently in the text but the opposite in other texts.TF as a metric can highlight the document characteristics. It can be said that the larger the TF is, the better its ability to distinguish different texts according to TF-IDF. Therefore, this paper introduces the IDF through which the weights TF are adjusted.The purpose of adjusting the weights TF is to emphasize the key items and suppress the subordinate items. Finally, the weights of all items are ranked.

TF and IDF in the original TF-IDF are item frequency and inverse document frequency, respectively. Since there are certain problems with the TF-IDF as described above, in this paper a weight must be added to the original TF-IDF. The added weight takes into account the frequency of the term, which belongs to a specific category in the whole collection of texts, instead of simply taking into account the frequency of the term that exists in other documents in the whole collection of texts.

Suppose N is the total number of texts, N is the number of texts containing $t$, and m is the maximum number of texts containing $t$ in a given category. Therefore, to improve the ability to distinguish between categories, i.e., the category to which the text belongs contains the term $t$ belongs to and other categories, a new weight $c_i = 1/(n-m+1)$ is introduced.

The formula is then modified to:

$$TF * IDF_{ij} * C_i \rightarrow tf_{ij} \times idf_i * C_i = TF \times \log \frac{N}{n} * 1 \Big/ (n-m+1) \tag{10}$$

where, $(n-m)$ is the value of $d$ between the number of all texts containing the term $t$ and the maximum number of texts containing the term $t$ in a category.

When the number of texts in a category containing the term $t$ is large, the number of texts in other categories containing the term $t$, i.e. $(n-m)$, is small. Then the text whose terms can represent the characteristics of the category contains the most terms $t$, so the weighted value is large, i.e., the term $t$ has inversely proportional to the ability to represent the characteristics of the text in all the categories except for the category containing the largest number of terms $t$. Because the term $t$ is contained in only a single category, i.e., $m$ is equal to $n$ the denominator should be $(n-m+1)$.

**III. B. 2) Data fusion classification**

Data fusion is an important part of the multi-source data fusion module, this paper on the preprocessed text vectors fused in the time dimension, and ultimately form a “superset” of information.

The main body of situational awareness is data fusion, and the core of data fusion is classification algorithms, the algorithm is selected from the hottest and most popular machine learning. The best machine learning is deep neural network.

The typical structure of DNN, as shown in Figure 2:

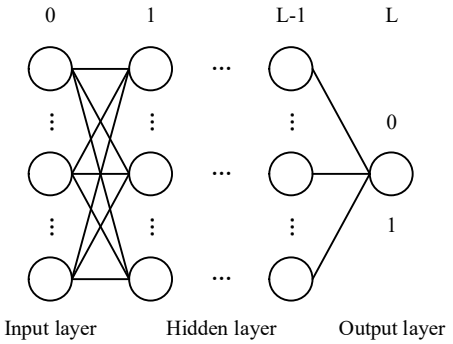



Figure 2: The Basic structure of DNN

A deep neural network consists of an input layer, a large number of hidden layers, and an output layer. In Figure 2, the balls represent neurons, and each link between neurons is a causal chain that can be learned and trained. Any neuron in any of these layers connects to every neuron in the next layer. The entire DNN model is composed of a linear function and an activation function in Eq:

$$a = \sum_{w_i x_i + b_i} \tag{11}$$

where, $x_i$ is the input value of each neuron, $W_i$ is the linear relationship coefficient, $b_i$ is the deviation.

Assuming that there are L hidden layers in the DNN, the computation of the output values can be expressed as follows:

$$f(x) = w^L x + b \tag{12}$$

where, $L$ denotes the Lth layer, $X$ is the matrix of input variables, and $W$ and $b$ are high-dimensional matrices.

$f(x)$ is the activation function introduced to increase the nonlinearity of the neural network to approximate any nonlinear function of many nonlinear models.

The values of W and b are determined automatically considering the minimum of the loss function during training. These values are non-negative real numbers, and they guide the learning of the network parameters by back-propagation of the errors generated by the labeling of the predicted and real samples.

Adding a dual topology to the DNN described above solves the problem of preference of decisions over multiple objectives, makes the DNN more suitable for accurate and inaccurate comparisons (results are given in approximations or on an interval scale), and resolves the conflict between the size of the training samples and the difficulty of performing boring and lengthy “smart tests” for the decisions.

According to Hecht-Nielson, there exists a feedforward neural network ANN(Z) $\approx$ V(Z), and if both ANN(1) and ANN(2) are consistent with the ANN, then for two different criterion vectors $z^1$ and $z^2$ and ANN( $z^1$ ) and ANN( $z^2$ ) are the DM preference values, respectively.

The output of DNN( $z^1, z^2$ ), which is equal to ANN( $z^1$ )/ANN( $z^2$ ), is the "value" of the comparison. Therefore, we can use the comparison result as a training sample for the DNN.

The proof of Proposition 1 follows directly from the theorem proved by Hecht-Nielson, which shows that feedforward neural networks can present any continuous mapping.

Proposition 2: Given $DNN(X,Y) = ANN(X) / ANN(Y)$ and a $MAUFV(Z)$ where $X, Y, Z$ belongs to $R^n$, $k$ is a constant if there is $ANN(X) / ANN(Y) = V(X) / V(Y)$.

Proof: then let $Y$ be:

$$\frac{ANN(X)}{ANN(Y_0)} = \frac{V(X)}{V(Y_0)} \quad X \in R^n \tag{13}$$

It can be rewritten as:

$$ANN(X) = \frac{ANN(Y_0)}{V(Y_0)} V(Y) \quad X \in R^n \tag{14}$$

Let $K = ANN(Y_0) - V(Y_0)$ have it:

$$ANN(X) = K * V(X) \tag{15}$$

Proposition 2 says that a neural network is proportional to the MAUF if there exists a neural network that returns the same comparison result as given by the MAUF of any pair.

Let $n$ be the number of nodes of the ANN (where ANN(1) and ANN(2) are equal to the ANN), and $w_{ij}$ be the value of the weighted link connecting node $i$ to node $j$. For simplicity, the thresholds for all nodes are discarded. Furthermore, it is assumed that all nodes have an identical activation function $f$. Let $(z_k^1, z_k^2, y_k)$ be a training sample, where $k = (1, 2, 3, \ldots, n)$ and $n$ are the number of all training samples. The subscript $k$ is used as the index of the training samples, which is the default in the following discussion. And $z_k^1$ and $z_k^2$ are the inputs of ANN(1) and ANN(2), and respectively $O_{1ik}$ is set as the output of node $i$ of ANN(1), $O_{1ik}$ is the output of node $i$ of ANN(2), and $\hat{y}_{1k}$ is the output of output of the ANN(1) node, $\hat{y}_{2k}$ is the output of the ANN(2) node, and $\hat{y}_k$ is the output of the DNN. The obvious $\hat{y}_k = \hat{y}_{1k} / \hat{y}_{2k}$. Then the $j$ input to the ANN(1) node is:

$$net_{ljk} = \sum_i w_{i,j} o_{1k} \tag{16}$$

Let E denote the error level, where,

$$E = \frac{1}{2} \sum_{k=1}^{N} (y_{k-} \overline{y}_k)^2 \tag{17}$$

The training procedure is to minimize E by adjusting weight $w_{ij}$:

$$E_k = \frac{1}{2} (y_k - \overline{y}_k)^2 \tag{18}$$

Then:

$$E = \sum_{k=1}^{N} E_k = \frac{1}{\hat{y}_{2k}} \cdot \frac{\partial \hat{y}_{1k}}{\partial W_{i,j}} - \frac{\hat{y}_{1k}}{\hat{y}_{2k}^2} \cdot \frac{\partial \hat{y}_{2k}}{\partial W_{i,j}} \tag{19}$$

In ANN(1), let:

$$\delta_{l\,jk} = \frac{\partial \hat{y}_{1k}}{\partial net_{1\,jk}} \tag{20}$$

Then:

$$\frac{\partial \hat{y}_{1k}}{\partial W_{ij}} = \frac{\partial \hat{y}_{1k}}{\partial net_{1\,jk}} \cdot \frac{\partial net_{1\,jk}}{\partial w_{i,j}} = \delta_{1\,jk} \cdot o_{1ik} \tag{21}$$

If it is an output node of ANN(1) (for DNN, for ANN(1) there is only one output node), then:

$$\delta_{1jk} = \frac{\partial y_{1k}}{\partial net_{1k}} = f'(net_{1jk}) \tag{22}$$

Otherwise:

$$\delta_{1jk} = \frac{\partial y_{1k}}{\partial net_{1jk}} = \frac{\partial \hat{y}_{1k}}{\partial o_{1jk}} . \frac{\partial o_{1jk}}{\partial net_{1jk}} = \frac{\partial \hat{y}_{1k}}{\partial o_{1jk}} . f'(net_{1jk}) \tag{23}$$

For ANN(2), a similar conclusion can be drawn $\partial y_{2k} / \partial_{Wij}$. The difference exists only in the subscripts. Using $\partial y_{lk} / \partial w_{ij}$ and $\partial y_{2k} / \partial w_{ij}$, it can be calculated using $\partial_{E_k} / \partial_{W_{ij}}$ equation (23). Finally it is possible to calculate $\partial E / \partial_{W_{ij}}$ out.

### *III. C. Quantification of security situational awareness*

#### III. C. 1) Hierarchical Cybersecurity Assessment Models

Hierarchical network security assessment model adopts the idea of “from low to high, from point to surface”, and divides the network into four layers: attack/vulnerability, service, host and system. Layer by layer, it calculates the posture values of attack/vulnerability layer, service layer and host layer, and finally fuses the posture of host layer of each host into the posture value of the whole network. The hierarchical quantitative assessment model is shown in Figure 3.

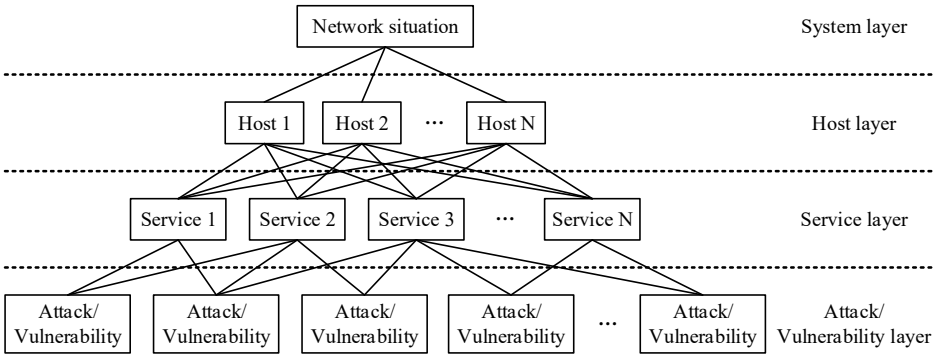



Figure 3: Hierarchical quantitative Evaluation model

The model takes the attack information as the original data, firstly, it evaluates the security status of each service at the attack layer through the factors of attack threat value and attack frequency, then combines with the vulnerability information of hosts, comprehensively evaluates the security posture status of hosts in the network, and finally realizes the posture evaluation of the whole network based on the importance of the attacked hosts in the network and the posture information.

#### III. C. 2) Service security posture

Based on multi-source fusion and feature selection, the perception of service security posture is firstly dealt with according to the perception hierarchy of the model. The service security posture perception is composed of factors such as attack intensity, attack weight, and number of attacks, from which the service security posture can be quantized and expressed as obtained from the attack threat weight $w_i$ and the corresponding number of attacks $N_i$ as shown in equation (24):

$$S_{s_i} = \sum_{i=1}^{n} (N_i 10^{w_i}) \tag{24}$$

where $n$ is the number of attack categories that the service has suffered and the importance of each category.

#### III. C. 3) Host security posture

The host security posture is determined by the posture of each type of service running on the host and the number of each type of service, also taking into account the weight factor of the service in the host. The host security posture can be expressed in equation (25):

$$S_{H_i} = \sum_{j=1}^{u} (w_{sj} S_{sj}) \tag{25}$$

where $u$ services present on host $H_i$, according to importance.

The weights are assigned ( $w_{s1}^1, w_{s2}^2, \cdots, w_{su}^u$ ) and normalized as:

$$w_{sj} = w_{sj}^j / \sum_{i=1}^{u} w_{si}^i \tag{26}$$

A larger value of $S_{H_i}$ indicates that the host is in a threatened state, otherwise it indicates that the host is in a safer state.

**III. C. 4) Cybersecurity posture**

Network security situational awareness is determined by the situational awareness of each type of host on the network and the share of each type of host in the network. The network security posture can be represented by equation (27):

$$S_N = \sum_{i=1}^{v} (w_{hi} S_{hi}) \tag{27}$$

where there are $v$ hosts on the network $N_i$, weights are assigned according to importance ( $w_{h1}^1, w_{h2}^2, \cdots, w_{hv}^v$ ) and normalized as:

$$w_{hi} = w_{hi}^i / \sum_{j=1}^{v} w_{hj}^j \tag{28}$$

Based on the above analysis process, if the value of $S_{N_i}$ is larger it means that the network is now in a threatening state, otherwise, the network is in a safer state.

**III. C. 5) Ecological calculations**

The hierarchical model of Section 3.3.1 is used to achieve a quantitative assessment of the situational awareness results by weighted fusion of the risk values of the elements of each layer of a multi-source event. The quantitative formula for each layer's security posture is given below to analyze the posture of each layer during the time window $\Delta t$ time period in a given unit of time.

Service layer $S_j$ posture values:

$$F_{s_j}(t) = \sum_{i=1}^{n} 10^{P_{ji}} C_i R_i \tag{29}$$

where $F_{s_j}$ is the value of the posture of the service $S_j$ at time $t$, $C_i$ is the number of attacks, $P_{ji}$ is the value of the threat posed by the attack to the service, $n$ is the number of types of attacks generated against the service at time $\Delta t$, and $R_i$ is the confidence level of the attack, and its value is derived from the DS evidence theory algorithm in Section 3.1.

Host layer $H_j$ posture value:

$$F_{H_j}(t) = \sum_{j=1}^{n} V_j F_{s_j}(t) + \sum_{i=1}^{k} (In10)^{P_i} \tag{30}$$

where $F_{H_j}$ is the posture value of the host $H_j$ at time $t$, $n$ is the number of services opened by the host, $V_j$ is the weight of the service $S_j$, and $P_i$ is the value of the threat posed by the vulnerability to the host.

Network layer L posture value:

$$F_L(t) = \sum_{i=1}^{n} W_i F_{H_i}(t) \tag{31}$$

where $n$ is the number of hosts in the LAN $L$ and $W_i$ is the weight of the hosts in the network.

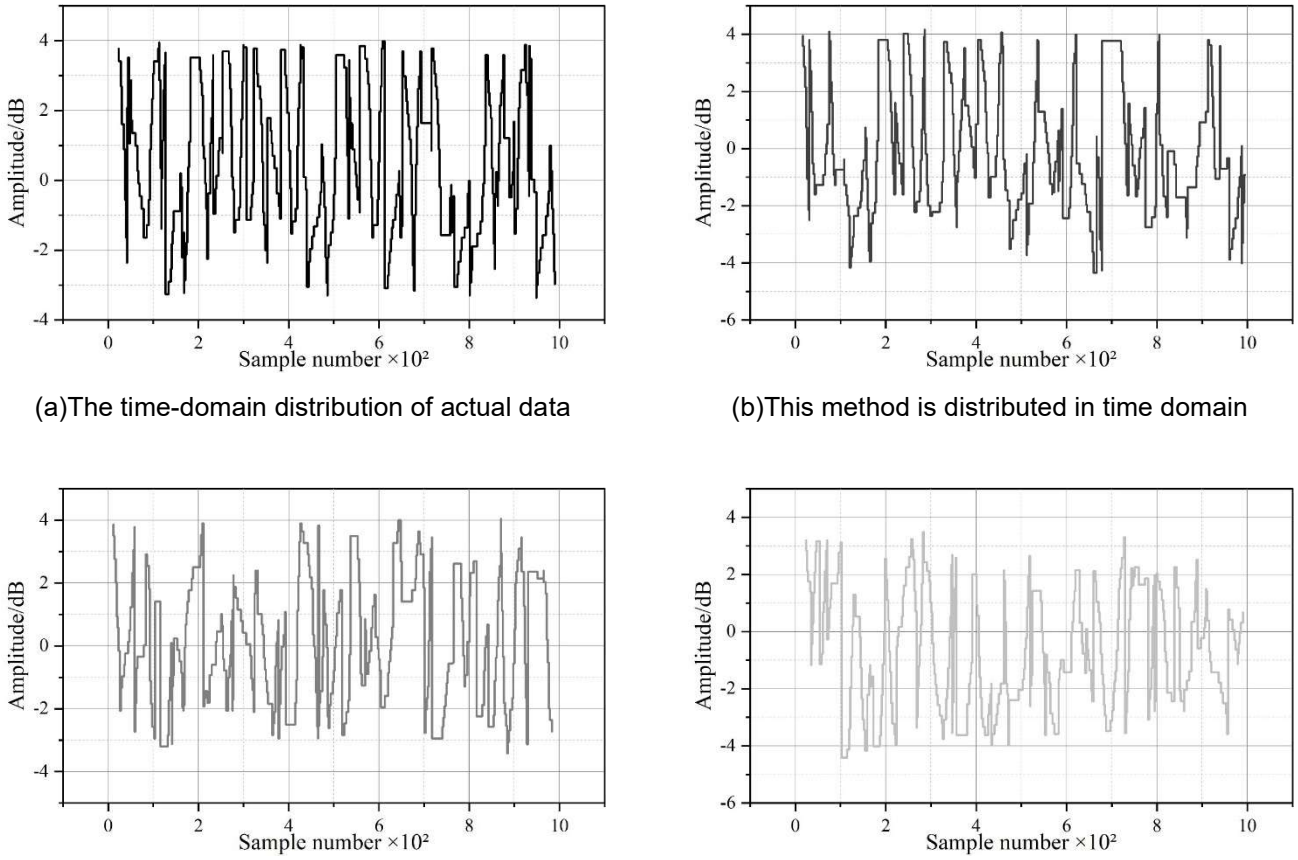
# IV. Experimentation and analysis

## *IV. A. Security situational awareness*

### IV. A. 1) Multi-source data time domain detection results

In order to verify the effectiveness of the proposed method for experimental testing, first of all, build a test network, based on the built test network, to the running network to join a set of multi-source network data, respectively, using the three methods proposed in this paper, intrusion detection, and the time factor, every 1s on the network operation of the data collection, in the data collection of the carrier frequency of the same premise, the multi-source network data analysis, and the analysis results are shown in Figure 4. The analysis results are shown in Fig. 4.

Under the premise of increasing the number of sampling points for multi-source data analysis, the real-time distribution of multi-source data is most similar to the actual one, with the time-domain amplitude distribution of the actual data in the range of [-4,4] dB, and the distribution of the proposed method in this paper in the range of [-4.02,4.02] dB, with the distribution trend being almost the same. This is because the proposed method utilizes the traffic detection module in the application layer of the network security posture assessment system, implements deep mining of network operation data, and combines with the ant colony algorithm in the multi-source information fusion module to realize multi-source data fusion processing at the head of network clusters, which improves the multi-source data analysis capability of the subsequent network security posture assessment system.




(a)The time-domain distribution of actual data

(b)This method is distributed in time domain

(c)Intrusion detection

(d)Time factor

Figure 4: Time domain detection results of multi-source data

### IV. A. 2) Exponential D-S evidence theory solution parameters

Multi-source data fusion will use the two-dimensional fusion strategy proposed in this chapter, which mainly consists of two parts: multi-evidence fusion and attack probability fusion, and the specific process of fusion is described in the previous section. Obviously the evaluation of the fusion result is an important part, so the analysis of the fusion result is added in this subsection.

The parameters that need to be adjusted in the process of solving the theoretical parameter values of exponentially weighted D-S evidence are the number of iterations $L$ and the learning rate. Too large a number of iterations $L$ may overfitting, the weaker the generalization performance, and too small a number of iterations $L$

may underfitting. Learning rate $\eta$ too large is prone to oscillations, too small convergence is slow and it is easy to fall into a local optimum. Table 1 shows the parameters of the exponential D-S evidence theory solution, and the highest convergence accuracy is achieved when the number of iterations is 500 and the learning rate is 0.000051. At this time, the accuracy rate of the training set and test set is 92.7568% and 90.9048, respectively, because the parameters are located in the exponential position, the slightest adjustment will have a greater impact on the output results, and it is easy to cause oscillation, so the learning rate is small.

Table 1: The index is the theoretical solution parameter

| Iteration number $L$ | Learning rate $\eta$ | Accuracy of training set (%) | Test set accuracy (%) |
|---|---|---|---|
| 400 | 0.000014 | 92.3485 | 90.6425 |
| 400 | 0.000052 | 92.6361 | 90.8669 |
| 400 | 0.0002 | 92.7165 | 90.7485 |
| 500 | 0.000013 | 92.5698 | 90.7485 |
| 500 | 0.000051 | 92.7568 | 90.9048 |
| 500 | 0.0002 | 92.7248 | 90.7822 |
| 600 | 0.000012 | 92.5686 | 90.7369 |
| 600 | 0.000054 | 92.7169 | 90.8548 |
| 600 | 0.0003 | 92.7458 | 90.8436 |

### *IV. B. Security posture assessment results*

#### IV. B. 1) Comparison of network posture detection

In order to verify the feasibility of the proposed method, 1200 pieces of each of the four types of network data for the purpose of privacy data theft, network privilege invasion, and network bandwidth consumption are selected to launch an attack on the network, and the proposed method, intrusion detection, and time factor are used to identify and detect the network security posture under different network attack behaviors, and the results are shown in Table 2.

The proposed method can effectively identify different types of network attacks, which proves that the proposed method is highly feasible for network security posture assessment. This is because the proposed method utilizes the security posture assessment module to implement the division of network threat level, combined with the theory of weight factor to threaten the network security threat factor for the threat degree assignment, in addition to the network infringement invasion of the false detection rate of 0.2486%, the rest of the types of network attacks false detection rate and omission rate are 0%. Utilizing the hierarchical division, the network structure is divided into three parts: host layer, server layer and network layer, and the overall security posture of the three-layer network structure is evaluated according to the results of the threat factor assignment. Therefore, the proposed method has a high recognition rate for multiple types of network attacks.

Table 2: Comparison of network situation detection

| Network attack type | Network situation detection method | Error detection rate | Leakage rate |
|---|---|---|---|
| Privacy data theft | This method | 0 | 0 |
| | Intrusion detection | 2.1655 | 6.2152 |
| | Time factor | 3.9458 | 3.4854 |
| Internet infringement | This method | 0.2486 | 0 |
| | Intrusion detection | 5.6485 | 4.1655 |
| | Time factor | 4.2486 | 7.3965 |
| Network broadband consumption | This method | 0 | 0 |
| | Intrusion detection | 2.9856 | 5.7452 |
| | Time factor | 3.3496 | 2.6453 |

#### IV. B. 2) Comparison of real-time network posture assessment results

In order to verify the accuracy of the proposed method for network security situation assessment, random host devices within the network system artificially launched an attack, the length of the attack was set to 60min, and the network risk caused by artificial network attacks was quantified using the [0,1] interval. The proposed method, intrusion detection and time factor are used to evaluate the current network security situation, and the comparison of the evaluation results of the three methods is shown in Figure 5.

The risk assessment of the time factor method has stopped after the 35th minute, the risk assessment result of intrusion detection has a large deviation from the actual one, with the sum of errors being -1.09219, and the risk assessment of the proposed method is the closest to the actual one, with the mean value of the error being -0.03037. This is because the proposed method establishes a network security posture assessment system with the core of the 3-layer construction of the application layer, the control layer, and the data forwarding layer, with the controller realizing the system application module to the network security situation, and with the controller realizing the system application module to the network security situation. The controller realizes the information transfer between the system application module and the network equipment, and combines the multi-module data processing in the application layer to realize the accurate perception of the network security situation. Therefore, the proposed method is better than the intrusion detection and time factor methods in assessing the network security situation.

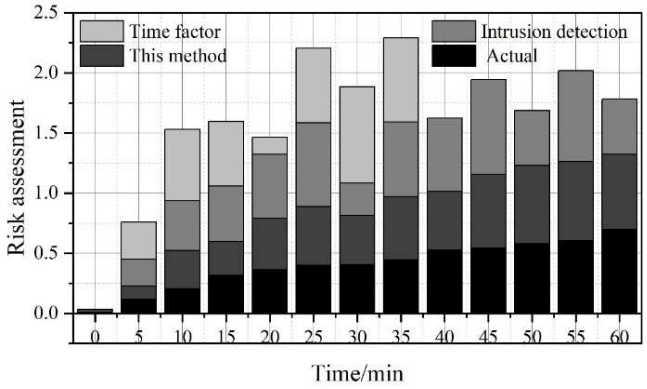



Figure 5: Comparison of real-time network situation assessment results

**IV. B. 3) Service layer posture**

Taking host 10 as an example, Figure 6 shows the service posture, presenting the two service postures of HTTP and SMTP. It can be visualized that the HTTP service was attacked twice in the 8th and 19th time windows, with service posture values of 1106.22026 and 1993.94973, respectively, and was attacked three times in a less severe manner. The SMTP service was attacked once in the 97th time window, with a less severe attack, with a service posture value of 246.62307. 246.62307. Administrators should be aware of the host's operation during the above time period and implement policies accordingly.

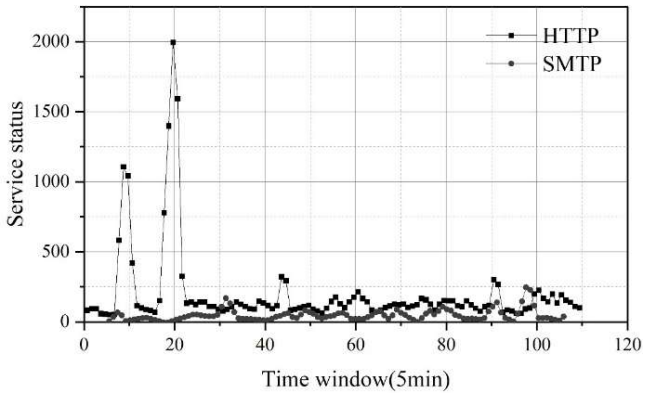



Figure 6: Service situation

**IV. B. 4) Host layer posture**

Based on the posture values of various services on the hosts and their corresponding weights, the posture values of the corresponding hosts are calculated. It is found that host 11, host 13 and host 14 are under more severe attacks during this period of time, and Fig. 7 shows the changes in the posture of the three hosts. Host 11 suffered two serious attacks near the 21st time window and 42nd time window, and the values of the hosts' posture were 1892.92844 and 2075.46562, respectively. host 13 suffered one serious attack near the 92nd time window, and

host 14 suffered two serious attacks near the 67th time window and 102nd time window, and all the three hosts suffered multiple small attacks during the test time were all subjected to multiple small-scale network attacks.

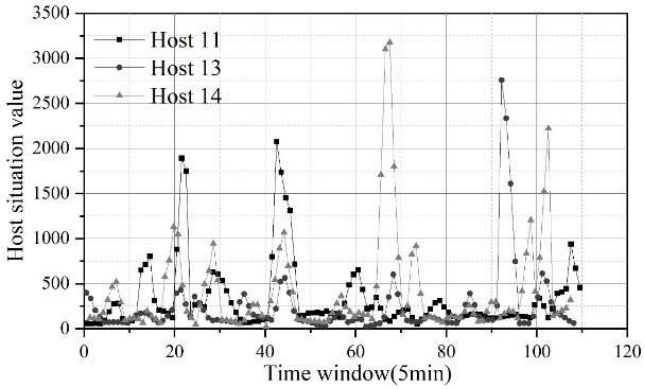



Figure 7: Changes in the dynamics of the three hosts

**IV. B. 5) Host posture values**

Table 3 shows the host potential values and the network layer potential values as a whole show a range of 10-60. A deeper analysis of the host potential values reveals that the peak potential values are relatively few at around 1000. Hosts 1-9 and host 20 have overall potential values below 20, and the number of hosts with potential values above 1000 is relatively small, host 11 has a potential value of 1342.6485 in the 46th time window, and host 18 has a potential value of 1312.2485 and 1324.4855 in the 61st and 76th time windows, respectively.Special Note: Hosts with a potential value of 0 indicate that they have not been attacked in the table. Some of the peaks are not shown in the table because they are not in the listed time windows.

Table 3: Host situation value

| Host | Time window 1 | Time window 16 | Time window 31 | Time window 46 | Time window 61 | Time window76 | Time window 91 | Time window 110 |
|---|---|---|---|---|---|---|---|---|
| Host 1 | 2.4485 | 0 | 0 | 2.4699 | 8.7596 | 1.4185 | 5.9485 | 8.4855 |
| Host 2 | 5.2364 | 2.7495 | 2.6485 | 13.3485 | 3.5455 | 8.5185 | 0.9425 | 0 |
| Host 3 | 7.0484 | 0.0569 | 0 | 4.5265 | 7.9485 | 2.8596 | 13.4545 | 8.4685 |
| Host 4 | 8.6585 | 8.3485 | 0 | 10.8485 | 2.9365 | 0.1458 | 0 | 5.0486 |
| Host 5 | 7.2486 | 0.7486 | 4.3585 | 0.0648 | 6.5158 | 7.8488 | 0 | 5.1265 |
| Host 6 | 0.0866 | 3.5269 | 0 | 6.6321 | 6.2369 | 14.3985 | 6.0415 | 11.8695 |
| Host 7 | 2.3485 | 1.7485 | 3.6485 | 0 | 18.6258 | 1.2648 | 5.0648 | 3.7485 |
| Host 8 | 9.8185 | 0 | 5.9485 | 2.8399 | 4.3486 | 1.2669 | 0 | 5.3266 |
| Host 9 | 10.2488 | 1.4986 | 0.9487 | 3.0789 | 3.4485 | 3.0545 | 0 | 3.3425 |
| Host 10 | 38.6485 | 50.9645 | 104.5395 | 482.8956 | 167.8558 | 135.3485 | 376.4866 | 70.0645 |
| Host 11 | 66.0548 | 370.4869 | 50.0958 | **1342.6485** | 690.1589 | 51.9485 | 74.0485 | 736.6248 |
| Host 12 | 46.1685 | 97.1635 | 590.0485 | 28.1358 | 68.6485 | 187.8845 | 322.4685 | 104.1658 |
| Host 13 | 69.1785 | 68.4215 | 37.6185 | 654.1588 | 65.2485 | 61.4898 | 74.0648 | 50.5369 |
| Host 14 | 54.6489 | 76.1699 | 41.7569 | 73.2948 | 167.9685 | 102.9348 | 322.4758 | 45.7858 |
| Host 15 | 58.7699 | 454.5748 | 84.6285 | 260.4528 | 392.1452 | 70.7466 | 60.0485 | 58.9888 |
| Host 16 | 167.3485 | 56.2348 | 53.7928 | 44.5936 | 142.3189 | 91.2485 | 216.4895 | 62.3785 |
| Host 17 | 81.4958 | 89.8487 | 293.2348 | 178.0315 | 92.4869 | 292.7485 | 657.4258 | 198.6485 |
| Host 18 | 206.4864 | 31.5269 | 74.8496 | 70.6484 | **1312.2485** | **1324.4855** | 48.8855 | 60.2448 |
| Host 19 | 28.3364 | 52.4458 | 96.8452 | 64.0658 | 276.8984 | 50.4485 | 38.7456 | 667.4558 |
| Host 20 | 2.4987 | 0 | 0 | 2.4398 | 8.7899 | 1.4458 | 152.444 | 8.4589 |

## V. Conclusion

With the support of multi-source data fusion, the security situational awareness capability of the rolling stock network has been significantly improved. Through comparative experiments, the security assessment method based on Bayesian network and Kalman filter algorithm proposed in this paper performs well in the detection of different types of network attacks, especially in the types of attacks such as privacy data theft and network bandwidth consumption, the false detection rate is almost zero. The experimental results show that the mean error value of the proposed method is -0.03037 during the evaluation process, indicating its high real-time performance and accuracy.

Further analysis reveals that through multi-source data fusion, the system is able to provide stable security posture assessment in different network environments. Especially in the dynamically changing network environment, the combination of the optimization mechanism of the ant colony algorithm makes the process of data fusion more efficient, and it can accurately identify potential security risks and provide timely warnings, which provides a reliable basis for network managers to make decisions.

Through the model proposed in this paper, the security situational awareness capability of the rolling stock network is comprehensively improved, which not only enhances the ability to prevent various network attacks, but also optimizes the efficiency and accuracy of data fusion, and provides a more scientific technical support for the security of the rolling stock network.

## References


[1] Tan, X. D., Du, L., & Song, P. W. (2014). Application of wireless communication technology in the detection and debugging for control system on bullet trains. Applied Mechanics and Materials, 470, 673-676.

[2] Lyovin, B. A., Shvetsov, A. V., Setola, R., Shvetsova, S. V., & Tesei, M. (2019). Method for remote rapid response to transportation security threats on high speed rail systems. International Journal of Critical Infrastructures, 15(4), 324-335.

[3] Liu, Y., & Yuan, L. (2018, December). Research on train control system based on train to train communication. In 2018 international conference on intelligent rail transportation (ICIRT) (pp. 1-5). IEEE.

[4] He, D., Sun, D., Chen, Y., Liu, G., Guo, S., Ma, R., ... & Liu, J. (2021). Topology design and optimization of train communication network based on industrial ethernet. IEEE Transactions on Vehicular Technology, 71(1), 844-855.

[5] Antony, J., & Maity, T. (2023). Analysis of Ethernet control network. IETE Journal of Research, 69(3), 1588-1596.

[6] Wang, L., Wang, W., & Liu, P. (2019). Train network control algorithms of high-speed EMU based on OPNET simulation. Journal of Computational Methods in Science and Engineering, 19(1_suppl), 77-83.

[7] Zhang, T., Li, C. X., & Li, Z. L. (2017, October). Generalized predictive control and delay compensation for high—Speed EMU network control system. In 2017 6th International Conference on Computer Science and Network Technology (ICCSNT) (pp. 511-515). IEEE.

[8] Shang, J., Cui, Y., & Pei, Y. (2024, May). Networked cruise control method of high-speed EMU under transmission delay. In 2024 36th Chinese Control and Decision Conference (CCDC) (pp. 2554-2559). IEEE.

[9] Qian, M., Huang, X., & Li, M. (2025). Optimization Research of High-Speed Railway EMU Utilization Schedule Based on Three-Dimensional Space-Time Network. Engineering Letters, 33(4).

[10] Zhao, H., Huang, Z., & Mei, Y. (2017). High-speed EMU TCMS design and LCC technology research. Engineering, 3(1), 122-129.

[11] Wang, C., Wang, L., Chen, H., Yang, Y., & Li, Y. (2020). Fault diagnosis of train network control management system based on dynamic fault tree and Bayesian network. IEEE Access, 9, 2618-2632.

[12] Jun, Y., Yan, X., & Chunjie, Z. (2024, November). Risk-Driven Security Strategy Decision-Making for High-Speed Train Control and Monitoring System. In 2024 China Automation Congress (CAC) (pp. 3976-3981). IEEE.

[13] Bao, F., Yu, H., & Wang, H. (2022). TSN - Based Backbone Network of Train Control Management System. Wireless Communications and Mobile Computing, 2022(1), 5789444.

[14] Liu, Y., Neri, A., Ruggeri, A., & Vegni, A. M. (2016). A MPTCP-based network architecture for intelligent train control and traffic management operations. IEEE Transactions on Intelligent Transportation Systems, 18(9), 2290-2302.

[15] Jakovljevic, M., Geven, A., Simanic-John, N., & Saatci, D. M. (2018, January). Next-gen train control/management (TCMS) architectures:“Drive-By-Data” system integration approach. In ERTS 2018.

[16] Sokol, P., Staňa, R., Gajdoš, A., & Pekarčík, P. (2023). Network security situation awareness forecasting based on statistical approach and neural networks. Logic Journal of the IGPL, 31(2), 352-374.

[17] Li Xiao,Tianheng Pan,Xiaoling Wu & Youkang Zhu. (2024). Research on Security Situation Assessment and Prediction Model of Network System in Deep Learning Environment. Journal of Cyber Security and Mobility,13(6),1263-1282.

[18] Dongying Han,Yu Zhang,Yue Yu,Jinghui Tian & Peiming Shi. (2024). Multi-source heterogeneous information fusion fault diagnosis method based on deep neural networks under limited datasets. Applied Soft Computing,154,111371-.

[19] Mai Rui & Wu Mingzhu. (2020). Research on the Quantitative Assessment and Security Measures of Hierarchical Network Security Threat Situation. IOP Conference Series: Materials Science and Engineering,750,012171-012171.

[20] Li Zhaowen,Zhang Qinli,Liu Suping,Peng Yichun & Li Lulu. (2024). Information fusion and attribute reduction for multi-source incomplete mixed data via conditional information entropy and D-S evidence theory. Applied Soft Computing,151,111149-.