Analysis of Collaborative Characteristics of Reinforcement Learning Intelligent Control and Bayesian Network Model in Network Security Protection

Zhi Hua Chang

Information Technology Center, Zhejiang University, HangZhou 310058 China
E-mail: zhchang@zju.edu.cn

Received 04 January 2025; Accepted 01 March 2025

Abstract

As network technology advances rapidly, the complexity of network security threats is growing, and it is difficult for traditional protection measures to cope with new attacks. This study explores the synergistic characteristics of reinforcement learning intelligent control and the Bayesian network model in network security protection. Reinforcement learning realizes dynamic responses to network attacks through self-learning and optimization strategies; Bayesian networks use probabilistic reasoning to accurately evaluate network status and potential threats by constructing a collaborative protection system integrating the two and conducting experimental verification in the simulated environment. The results show that compared with a single model, the collaborative system improves the threat detection accuracy by 25% and shortens the response time by 40%. Further analysis shows that reinforcement learning intelligent control effectively improves the adaptive ability of the system, while the Bayesian network model enhances the accuracy of threat

Journal of Cyber Security and Mobility, Vol. 14_2, 365–390.
doi: 10.13052/jcsm2245-1439.1425

prediction. The synergy between the two significantly improves the overall efficiency of network security protection. The study offers novel theories and methods to enhance network security and supports the development of intelligent security systems.

Keywords: Reinforcement learning, intelligent control, Bayesian networks, cybersecurity, collaborative protection.

1 Introduction

In the fast-paced information age, cybersecurity is crucial for national security, business stability, and personal privacy. With the diversification and complexity of network attack methods, traditional security protection measures have made it challenging to meet the current security requirements [1, 2]. Therefore, researchers are constantly exploring new technologies and methods to improve the ability to protect network security [3]. Against this background, reinforcement learning intelligent control and the Bayesian network model have gradually become research hotspots in network security protection because of their unique advantages. This paper aims to analyze the collaborative characteristics of reinforcement learning intelligent control and the Bayesian network model in network security protection and explore the interaction between the two to enhance network security protection systems.

At present, the situation of network security protection is severe. On the one hand, traditional protection systems rely on preset rules and signatures to detect threats, and are slow to respond to changing attack patterns. When new malware variants emerge, legacy systems are often difficult to identify in a timely manner, resulting in long-term security breaches. On the other hand, the scale of the network continues to expand, and emerging technologies such as cloud computing, Internet of Things, and industrial Internet are widely used, and the network architecture is becoming more and more complex.

The requirements for cyber security protection are also increasing. Not only do companies have to protect trade secrets and customer data, but they also have to meet strict compliance requirements, such as the European Union's General Data Protection Regulation (GDPR) and China's Cybersecurity Law, and they will face high fines and legal liabilities for violations. In critical infrastructure fields such as power, transportation, and communications, cyber security is related to national security and social operation, and even the slightest vulnerability can lead to serious consequences. In addition, the popularity of remote work and online education, and the access of a large

number of personal devices to the network have further expanded the attack surface, putting forward higher requirements for the flexibility and scalability of network security protection. Based on this, it is of great significance to explore the synergistic characteristics of reinforcement learning intelligent control and Bayesian network model in network security protection to break through the current network security dilemma.

Reinforcement learning, a type of machine learning, seeks to learn the best strategy via interaction between agents and their environment [4, 5]. In network security protection, reinforcement learning can help the protection system adaptively adjust the protection strategy to cope with the changing network threats. Intelligent control is a method to effectively control complex systems by using advanced computational intelligence theory [6]. Combining reinforcement learning with intelligent control can provide a dynamic and adaptive solution for network security protection so that the protection system can respond quickly and accurately in an uncertain network environment [7].

As a probabilistic graph model, the Bayesian network model plays an important role in network security with its powerful uncertainty reasoning ability [8, 9]. Bayesian networks can effectively represent the dependency relationship among various security events in the network and realize the prediction and analysis of network attack behaviour through probabilistic reasoning [10]. In network security protection, the Bayesian network model can help security analysts better understand the nature of network threats to formulate more effective protection strategies.

The collaborative application of reinforcement learning intelligent control and Bayesian networks offer a novel approach to enhancing network security. Reinforcement learning intelligent control can dynamically adjust Bayesian network parameters for better network environment adaptation. This dynamic adjustment ability is critical with the constant update of network attack methods. Secondly, the Bayesian network model provides rich prior knowledge and probabilistic reasoning ability for reinforcement learning intelligent control, which enables the intelligent control system to make reasonable protection decisions even when the information is incomplete.

In the real-world utilization of network security safeguards, the synergistic characteristics of reinforcement learning intelligent control and the Bayesian network model are manifested in many aspects. For example, reinforcement learning can optimize detection strategies in intrusion detection systems, while Bayesian networks are used to analyze attacker behaviour patterns. The combination of the two not only improves the accuracy of the detection system but also reduces the false alarm rate. For another example,

in malicious code detection, reinforcement learning intelligent control can dynamically adjust the detection threshold according to the probability information provided by the Bayesian network model, thereby ensuring the detection effect while reducing the impact on normal user operations.

However, the collaborative application of reinforcement learning intelligent control and Bayesian network model in network security protection also faces many challenges. Researchers need to solve the key problem of designing an effective collaborative mechanism to ensure that the two can work together efficiently. In addition, the model's training and optimization require a large amount of data support. Improving the model's training efficiency while ensuring data quality is also one of the current research hotspots.

This paper will intensely discuss the collaborative characteristics of reinforcement learning intelligent control and the Bayesian network model in network security protection from theoretical analysis and experimental verification perspectives. By combining the existing research results, this paper will propose a new collaborative framework and verify its application value in practical network security protection through simulation experiments. We hope that through the research of this paper, we offer a theoretical foundation and practical guidance to advance cybersecurity technology.

2 Overview of Basic Theories

2.1 Basic Principles of Reinforcement Learning Intelligent Control

Deep reinforcement learning is a fusion of deep learning and reinforcement learning that aims to learn high-level features and develop strategies that maximize rewards, as can be seen in the theory [11, 12]. DRL models are generally built based on adaptive neural networks. Reinforcement learning determines the optimal strategy through the interaction between the agent and the environment, which involves four key elements: agent, state, action, and reward [13]. As the executive agent, the agent is able to perceive the surrounding environment and take corresponding actions according to its own goals. The agent environment here is the external world in which the agent is located and interacts, and it covers all the information that the agent can perceive and the scope of influence on the agent's behavior. Each action of the agent will cause the state of the environment to change, and the environment will give a reward as a reward based on the agent's behavior.

Reward is a quantitative evaluation of the agent's behavior, with positive rewards indicating that the agent's behavior is conducive to achieving the goal, and negative rewards indicating that the behavior deviates from the goal, and the environment helps the agent optimize the strategy through this reward mechanism [14]. The core of reinforcement learning is to imitate trial and error and exploration, and in the process of constantly trying different actions, the agent improves the strategy according to the reward feedback given by the environment, so as to achieve the advantage and avoid the disadvantage, and gradually find the optimal strategy.

Reinforcement learning is a machine learning approach enabling agents to learn optimal behavior via interaction with their environment. It imitates trial-and-error and exploration mechanisms, allowing agents to train and learn autonomously [15, 16]. The agent observes the environmental state, selects actions according to experience, changes the environmental state, influences the following action, and forms an interactive cycle.

The parameter setting of reinforcement learning intelligent control has a rigorous basis for consideration. The learning rate, which refers to a large number of previous research results on the use of reinforcement learning in the field of network security, has a value of 0.01, which can ensure the learning speed of the agent while preventing the algorithm from not converging due to a large learning rate, or making the learning process too slow due to a small learning rate. The discount factor is set to 0.9, which is based on theoretical calculations, which determines the importance of the agent to future rewards, which means that the agent will not only pay attention to the immediate reward when making decisions, but also fully consider the long-term benefits in the future, which helps the agent to formulate a more long-term defense strategy in the dynamic environment of network security. The exploration rate is set to 0.9 in the initial stage, which is based on the consideration of the complexity and uncertainty of the network security environment, and a higher exploration rate allows agents to actively try different defense actions in the initial stage, and extensively explore the network security state space to discover more potentially effective defense strategies. As the learning process progresses, the exploration rate decays exponentially, which is based on the optimization experience of the agent learning process in previous studies, so that the agent can gradually reduce unnecessary exploration in the later stage, and instead use the learned knowledge to focus on the implementation of those defense strategies that have been proven to be effective, so as to improve the efficiency and stability of network security protection. These parameters affect the performance and

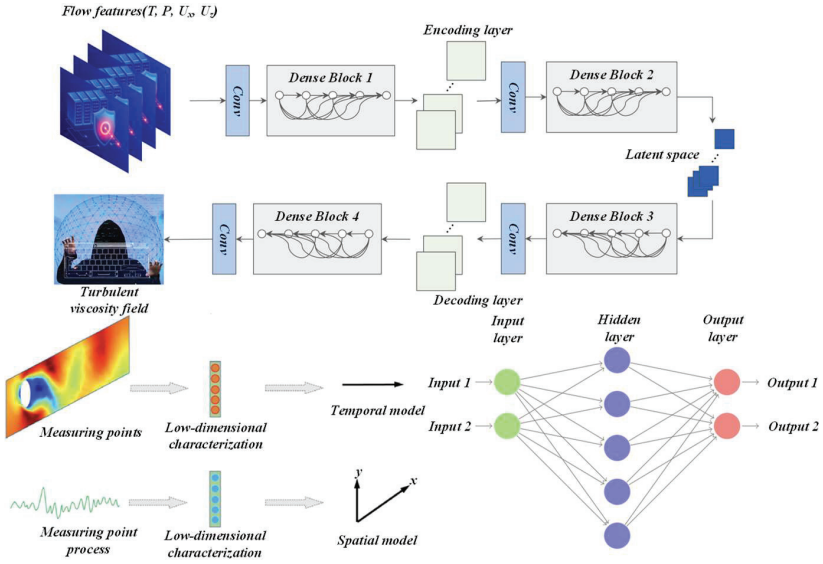

Figure 1 Reinforcement learning intelligent control structure.

behavior of the intelligent control system in the context of network security protection in different ways, and jointly build an intelligent protection system that can adapt to the complex network security environment.

Agents will receive reward feedback from the environment after taking action. The reward can be positive, negative, or zero, representing the benefit, harm, or neutral behavior of the behavior, respectively [17]. Agents utilize these rewards to evaluate and update their policies, possibly using methods such as Q-learning or Actor-Critic [18]. The agent improves by interacting with the environment to maximize rewards. Reinforcement learning's control structure is depicted in Figure 1. Post-training, the agent acquires a strategy for selecting optimal actions in similar future tasks. The essence of reinforcement learning is the ongoing interaction between the agent and its environment, realizing autonomous learning, reducing manual intervention, and making autonomous decisions in complex environments.

Markov decision process underpins policy optimization for agents in reinforcement learning, enabling observation of state transitions in fully observable environments and determination of information characteristics via the environment's state [19, 20]. Therefore, reinforcement learning problems are often transformed into Markov decision processes. A complete Markov decision process is composed of a five-tuple [S, A, P, R, y]: S represents the

set of environmental states. A denotes the set of actions. P denotes the state transition matrix. R denotes the reward function. γ denotes the attenuation factor, $\gamma \in (0, 1)$. In the Markov decision process, the agent determines behavior by learning the mapping from state to action to maximize the reward [21]. In MDP, state transitions and rewards depend only on the current state and action. MDP involves two kinds of value functions-state value function (V) and state action value function (Q), which are used to evaluate and improve strategies. Reinforcement learning aims to learn an optimal strategy for an agent's behavior, which can be either deterministic or stochastic [22]. By calculating these value functions, the agent can make better decisions. The state value function starts from state *s*, and the strategy reward value obtained by using strategy π is Equation (1):

$$v_\pi(s) = E[G_t|S_t = s] \tag{1}$$

$V_\pi(s)$ denotes the value function of state *s* under policy π. *E* is the expectation operator, which means taking an expected value for a random variable. G_t refers to the cumulative discount reward from time step t, which is further decomposed into the current reward and subsequent states, as shown in Equation (2):

$$v_\pi(s) = E_\pi[R_{t+1} + \gamma v_\pi(S_{t+1})|S_t = s] \tag{2}$$

R_{t+1} represents the immediate reward obtained at time step $t + 1$. γ is the discount factor, the state action value function starts from state s, executes action a, and then uses strategy π to obtain the expected reward value q as Equation (3):

$$q_\pi(s, a) = E_\pi[G_t|S_t = s, A_t = a] \tag{3}$$

$q_\pi(s, a)$ represents the action-value function of state s taking action a under policy π. $A_t = a$: The action taken at time step t is a. The state action value function is decomposed into the current reward and subsequent states, as shown in Equation (4):

$$q_\pi(s, a) = E_\pi[R_{t+1} + \gamma q_\pi(S_{t+1}, A_{t+1})|S_t = s, A_t = a] \tag{4}$$

At the same time, the two can transform into each other, as indicated by formula (5):

$$v_\pi(s) = \sum_{a\in A} \pi(a|s)q_\pi(s, a)q_\pi(s, a) = R(s, a) + \gamma \sum_{s'\in S} P^a_{ss'}V_\pi(s') \tag{5}$$

$R(s, a)$ represents the reward obtained immediately after state s takes action a. The probability of $P^a_{ss'}$ transitioning to state s' after state s takes action a.

$V_\pi(s')$ probability of transition to state s' after state s takes action a. Reinforcement learning aims to maximize the value function across all strategies, seeking the optimal value function, including the optimal V function and the optimal Q function, as shown in Equation (6):

$$\begin{aligned} v_*(s) &= \max_\pi v_\pi(s) \\ q_*(s,a) &= \max_\pi q_\pi(s,a) \end{aligned} \tag{6}$$

$v_*(s)$ denotes the optimal value function of state s in policy π. *max* represents the maximum strategy, and $q_*(s,a)$ represents the optimal action-value function for state s to take action a among all possible strategies π. In order to determine the optimal value function, the recursive relationship between the optimal Q function and the optimal V function is usually used. The optimal value function represents the function that achieves the best performance in the Markov decision process, and finding it means solving the problem of the decision process [23]. The recursive relationship between the optimal Q function and the optimal V function can be used to find the optimal value function, as shown in Equation (7):

$$\begin{aligned} v_*(s) = v_{\pi_*}(s) &= \sum_{a\in A} \pi_*(a|s) q_{\pi_*}(s,a) = maxq_{\pi_*}(s,a) = maxq_*(s,a) \\ q_*(s,a) &= R(s,a) + \gamma \sum_{s'\in\mathcal{S}} P^a_{ss'} v_*(s') \end{aligned} \tag{7}$$

Through the iteration of the formula, the optimal *Q* function and the optimal *V* function are finally produced. Reinforcement learning aims to learn a complete strategy to guide the behavior of agents. Its main types include value-based, strategy-based, and types that combine both. Four representative frameworks are Q-Learning, Policy Gradient, DQN, and DDPG.

Among reinforcement learning algorithms, Q-Learning is a classic value-based reinforcement learning method. $Q(S,a)$ is the expectation that $a(a \in A)$ action can be obtained in a certain state $(s \in S)$ at a certain moment. The agent acts within the environment, which responds with feedback and rewards based on those actions. The agent constructs Q-table2 to store the Q value through the state and action, and finally selects the action that can obtain the greatest benefit.

The problem is defined as a Markov decision process, and state s, action a, state transition function $p(s'|s,a)$ and reward function $r(s,a)$ are defined.

The ultimate goal is to find the way to obtain the maximum return in selecting action strategies. Formula (8) is expressed as:

$$Goal : \max_{\pi} E\left[\sum_{t=0}^{H} \gamma^t R(S_t, A_t, S_{t+1})|\pi\right] \tag{8}$$

Goal stands for objective conversion function, $\sum_{t=0}^{H}$ represents the summation from $t = 0$ to $t = H$, $R(S_t, A_t, S_{t+1})$: the reward obtained after taking action A_t and transitioning to state S_{t+1} at time step t, state S_t. Where the state value function of $Q_\pi(s, a)$ can be defined as Equation (9):

$$\begin{aligned} q_\pi(s, a) &= E_\pi[r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \cdots | A_t = a, S_t = s] q_\pi(s, a) \\ &= E_\pi[G_t | A_t = a, S_t = s] \end{aligned} \tag{9}$$

Where G_2 is the total discount reward starting at time t, $\gamma \in (0, 1)$. As γ approaches 1, the state value function increasingly focuses on long-term future rewards, and when γ approaches 0, the state value function focuses more on immediate returns.

2.2 Bayesian Network Model

Bayesian networks, as a graphical model for inferring causal relationships between variables, have a theoretical basis for [24, 25], and contain some basic elements. A Bayesian network is essentially a directed acyclic graph with nodes and directed edges, as mentioned in [26]. Nodes are used to represent variables, while directed edges are used to represent relationships between variables. In the relationship between nodes, there is a distinction between child nodes and parent nodes, which is manifested as the parent node pointing to the child node. The special thing is that the root node does not have a parent node, the leaf node does not have a child node, and the characteristics of a directed acyclic graph ensure that the edges do not form loops.

Conditional probability tables play a key role in Bayesian networks, as they clearly show how a parent node affects the probability of its children. In probability-related concepts, a priori probability is an initial probability assumption about a variable, while a posterior probability is the probability that is updated after new information is obtained. In terms of probability distribution, the joint probability distribution involves multiple variables, and when the probability of all possible state combinations of the random

variable $x_1, x_2, \ldots, x_n$ is expressed as $P(x_1, x_2, \ldots, x_n)$, $P(x_1, x_2, \ldots, x_n)$ is called the joint probability distribution. The marginal probability distribution involves one or several variables, and $P(X)$ is the marginal probability distribution. Together, these basic elements form a Bayesian network, which enables effective analysis and inference of relationships and probabilities between variables.

The chain rule outlines how to calculate the probability of multiple events occurring together. For events $B_1, B_2, \ldots, B_n$, and $P(B_1) = P(B_1B_2) = \cdots = P(B_1B_2 \ldots B_{n-1}) > 0$, the multiplication formula is derived as shown in Equation (10).

$$P(B_1B_2LB_n) = P(B_1)P(B_2|B_1)P(B_3|B_1, B_2)LP(B_n|B_1, B_2, L, B_{n-1}) \tag{10}$$

Suppose $B_1, B_2, \ldots, B_n$ are complementary sets of complete events T and $P(B_i) > 0$. A is an event of T, and the full probability calculation formula P is shown in Equation (11).

$$P(A) = \sum_{i=1}^{n} P(B_i)P(A|B_i) \tag{11}$$

Bayesian formula is a method to calculate conditional probability, and the full probability formula can rewrite Bayesian formula, as shown in Equation (12).

$$P(B_i|A) = \frac{P(A|B_i)P(B_i)}{\sum_{i=1}^{n} P(A|B_i)P(B_i)} \tag{12}$$

Three standard methods to establish Bayesian networks include [27]: experts directly set the network structure and conditional probability table, using data fusion to determine node probability distribution and network structure; And the two-stage modelling method, that is, the other models are converted into Bayesian networks. These methods help to create efficient Bayesian network models for probabilistic reasoning and statistical analysis to solve practical problems. Bayesian network depicted in Figure 2.

Bayesian networks involve three probabilistic reasoning methods. The first is maximum posterior hypothesis reasoning (MAP), which adjusts the posterior distribution to fit input conditions, looking for the most likely outcome. The Maximum Possible Explanation Problem (MPE), which predicts events based on probability values, provides an optimal solution for posterior probability and is a form of MAP. Posterior probabilistic inference (PPI) combines a priori probability and a posterior probability to predict future events by calculating a posterior probability.

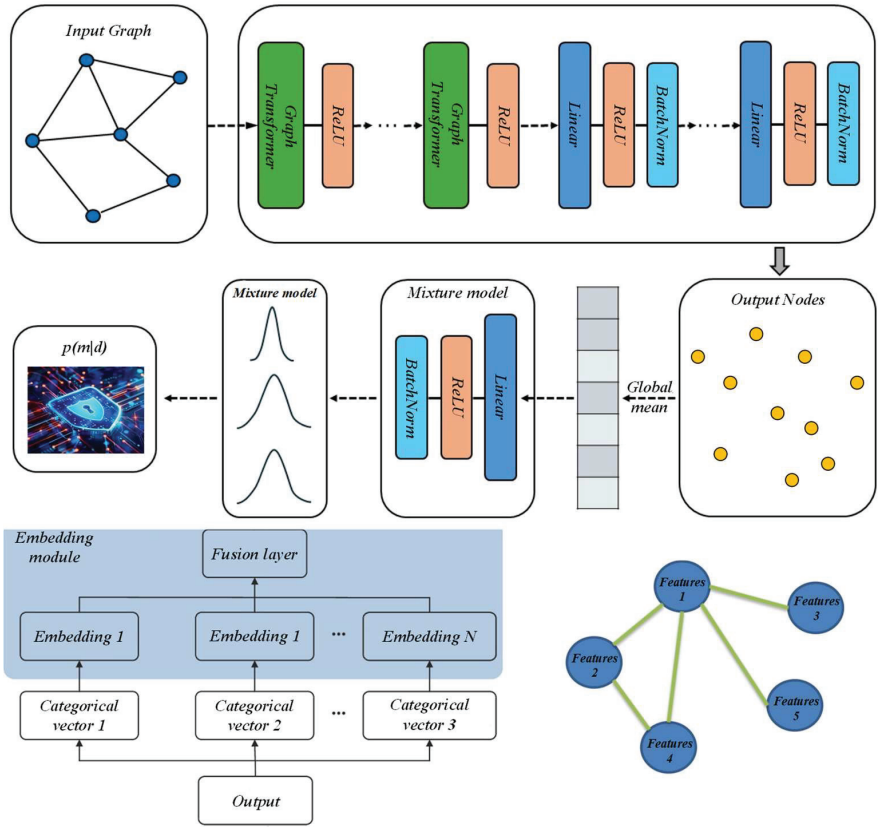

Figure 2 Bayesian network.

3 Algorithm Model and Collaborative Mechanism

3.1 Construction and Optimization of Bayesian Network Model

In the field of network security protection, the research on the synergistic characteristics of reinforcement learning intelligent control and Bayesian network model is deepening. The research greatly expands the research scenarios by comprehensively considering different types of cyber threats, such as malware attacks, phishing, DDoS attacks, etc., as well as different network architectures such as centralized and distributed, while taking into account the diverse requirements of various users in network security protection, including data privacy protection for enterprises and transaction security for financial institutions. This multi-dimensional expansion makes

the research results more comprehensive, can better adapt to a wide range of network security protection scenarios, and provides a solid theoretical and practical foundation for building a more complete and effective network security protection system.

In blockchain networks, solar eclipse attack significantly impacts the security performance of the whole network environment. Therefore, how to collect the intrusion data of solar eclipse attacks is a critical issue. Traditionally, the division of network traffic is based on port numbers, but many applications today use dynamic port numbers, which makes this method have little effect. There are also ways to split this traffic using different application strings, which are more accurate than previous methods, but there is a problem: the size of the encrypted traffic cannot be determined. In addition, string matching is very expensive and unsuitable for online analysis.

A machine learning approach has been developed to analyze vast data sets and construct relevant models. Assuming traffic classification, analyzing large data sets involves examining numerous message packets. Data packets with the same tuple can be added as data flow objects, and corresponding traffic data information can be calculated.

In the field of network security protection, the random forest algorithm is used to study the synergistic characteristics of reinforcement learning intelligent control and Bayesian network model. There are several key variables involved and the tightly ordered relationships between them: the data acquisition module is the foundation, which consists of two parts: traffic detection and data preprocessing. The TCPdump tool is used to capture TCP/IP packets on the blockchain network in real time and store them as Pcap documents to complete traffic detection. Then, the timestamp, source and destination IP addresses, and protocol types of the packets in the pcap file are analyzed, and they are exported to a CSV file for data preprocessing. Then, the image data extraction module uses the image data mining model to count the data flow feature combinations processed by the data acquisition module, and merges the separated features into a test data set. In Eclipse attack traffic analysis, an M-set is created through a specific process, that is, the packets are identified by the IP address of the victim node, and then grouped by source IP address and port number to form different data flow objects, the grouped data information is calculated, and the attribute values are extracted, which are combined with the CSV file to obtain a verifiable M-set under test. Both the test dataset and the M-set are used for random forest model training, and after training, k classification results are generated for model classification, and the final classification results are obtained through simple voting. Standard and

Eclipse attributes, on the other hand, play a key role in the overall traffic classification process as the basis for traffic classification, helping to clarify the final classification.

The data set $D = (d_1, d_2, \ldots, d_n)$ represents the observed values for n variables $(x_1, x_2, \ldots, x_n)$, where it is assumed that θ_{BNM} is a parameter value, which corresponds to BNM. $P(BNM)$ represents the prior knowledge of each node in the Bayesian network model. When the model is preliminarily completed, θ_{BNM} is represented by $P(BNM_\theta)$. Its correction function is shown in Equation (13).

$$P(BNM, D) = log_a P(DBNM) + log_a P(BNM) \tag{13}$$

Under normal circumstances, $P(BNM)$ is assumed uniformly distributed. After the above processing, the greedy search algorithm is used to find the network structure that meets the needs. Select one directed edge from the model at a time and add it. Using the above formula, the judgment value can be obtained. If this value becomes larger, the directed edge will be added to the model, otherwise proceed to the next step. When calculating the posterior probability, if the variable is not a discrete variable, the probability density function $p(x|c) \sim N(\mu_{c,i}, \sigma^2_{c,i})$ can be used, where $\mu_{c,i}$ and $\sigma^2_{c,i}$ respectively represent the average and variance of the value on the *i-th* attribute of the sample of class *c*, which can be calculated by formula (14).

$$p(xc) = \frac{1}{\sqrt{2\pi}\sigma_{c,i}} exp\left(-\frac{(x_i - \mu_{c,i})^2}{2\sigma^2_{c,i}}\right) \tag{14}$$

When we obtain the state transition probabilities of various nodes in the blockchain, in order to further perceive the security situation of the current network environment, according to the above model, we can find out the attack path leading to the final target. This paper adopts a path generation algorithm: starting from the destination node, the parent node of the current node goes to the source node in reverse order. These sets of nodes are the attack path.

3.2 Synergy Characteristic Analysis

In the field of network security protection, although the advantages of reinforcement learning intelligent control and Bayesian network model are used in synergy, there are also many potential risks. Reinforcement learning is easy to fall into overfitting in the training process, overlearning specific patterns

in training data, but it is difficult to generalize and apply in the face of the actual complex and changeable network environment, resulting in inability to deal with new threats. Similarly, if the training data is biased or incomplete, the Bayesian network model will also overfit, misinterpret the relationship between variables, and affect the accuracy of risk assessment. Reinforcement learning relies on environmental feedback adjustment strategies, and once the data collection is interfered by noise or the reward mechanism is not properly designed, the agent will receive wrong feedback, misjudge the network state, and misadjust the strategy. The Bayesian network model will be biased in probability inference due to problems such as missing data and mislabeling, and give misleading risk assessment and state prediction. In addition, there may be conflicts between the two in the decision-making process, reinforcement learning pursues the maximization of immediate rewards and often optimizes strategies through trial and error, while Bayesian network models make decisions based on knowledge and probability inference, and pay more attention to long-term risk assessment and stability, which leads to contradictions in the decision-making of both parties in the face of urgent but small short-term cyber threats, which in turn affects the protection effect. A comprehensive analysis of these risks can give us a more comprehensive understanding of the cooperative relationship between them, and provide a strong reference for optimizing the synergy mechanism.

In the field of network security, the collaboration mechanism between reinforcement learning and Bayesian network model is of great significance. Reinforcement learning continuously adjusts its strategy to adapt to changing cyber threats through continuous interaction with the environment, and the agent optimizes its strategy based on environmental feedback (reward or punishment) after taking action, such as trying different defense measures in the face of attacks and improving the strategy according to the effect. With its powerful uncertainty reasoning ability, the Bayesian network model uses nodes and directed edges to represent variables and causal relationships, quantifies relationships with the help of conditional probability tables, accurately predicts network status and assesses risks based on multi-source data, and infers the probability of attacks by analyzing network traffic patterns and attack characteristics. When the combination of reinforcement learning intelligent control focuses on policy optimization and Bayesian network models that are good at capturing variable dependencies, reinforcement learning can use the information provided by Bayesian networks to accurately perceive network conditions, such as Bayesian networks predicting that they may suffer from DDoS attacks, and reinforcement learning can adjust defense

strategies in advance. When dealing with unknown attacks, reinforcement learning accumulates experience through continuous exploration of new defense methods, and Bayesian networks infer the probability of unknown attacks based on existing knowledge and data, and transmit information to reinforcement learning to help it dynamically adjust strategies and achieve optimal allocation of resources, and finally enable the network protection system to quickly complete state awareness, risk assessment, and policy adjustment, respond to attacks quickly, and effectively improve network security protection capabilities and efficiency.

In this study, the selection of quantitative indicators of collaboration characteristics is crucial. The research deeply refers to the theories of threat detection and risk assessment in the field of cybersecurity, as well as state-action space analysis and reward mechanism design in the field of reinforcement learning. Based on these established theories, more appropriate quantitative indicators were re-screened. For example, in terms of threat response speed, the number of different types of cyber threats handled per unit time is introduced; In terms of the accuracy of risk assessment, the deviation rate between the predicted risk and the actual risk is taken as the index. In terms of the degree of decision optimization, the cumulative reward value when the reinforcement learning strategy converges is measured. In this way, it ensures that the index selection has a solid theoretical support, gets rid of randomness, and can more accurately describe the cooperative characteristics of the two.

4 Experiment and Results Analysis

The data collection comes from a wide range of sources, not only the internal network traffic data of enterprises in different industries such as finance, manufacturing, and the Internet to reflect the network security status of different business types, but also the open source security data of the public network, which brings together various threat samples shared by global network security researchers, and simulates common attack scenarios such as DDoS attacks, SQL injection attacks, and malware propagation in the laboratory, and collects a large amount of targeted data. In terms of data volume, the dataset contains billions of network connection records, millions of attack samples, and massive amounts of normal network behavior data, which can fully cover various complex situations in the field of network security, whether it is a rare advanced persistent threat or a common general attack. From a representative point of view, the data covers different

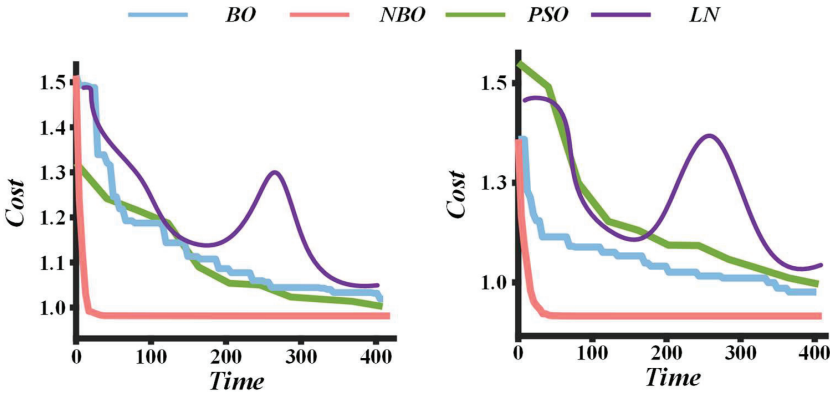

Figure 3 Comparison of system integrity.

network scales, security protection levels, and application scenarios, such as the simple network architecture of small enterprises, the complex distributed network of large multinational enterprises, as well as the well-established government agencies and relatively weak small and medium-sized enterprise networks. These multi-dimensional data sets are closely aligned with the research question, providing comprehensive and real network security scenario information for the experiment, helping the model to fully learn to respond to various network threats in training and testing, thus strongly supporting the experimental results and truly reflecting the performance of the model in actual network security protection.

Figure 3 indicates our proposed model excels in intrusion count and system integrity. This is due to the fact that the model allows the offensive and defensive strategies to be dynamically adjusted according to the real-time conditions of the blockchain network at each stage, thereby selecting the optimal solution to reinforce the network. Real-time detection imposes higher costs on attackers, making this model preferable for complex, evolving blockchain environments.

The constructed situational assessment dataset contains 18 metrics, and this study compares the performance of SOA-optimized PNN and PCA-PNN models using accuracy, precision, recall, and F1 metrics. The SOA-PNN model was trained with unextracted and extracted data, respectively. The PCA-derived data set possesses a dimension of 9, and the cumulative contribution rate exceeds 85%. Results Figure 4 shows that the PCA-PNN model performs better than the PNN model. PCA improves the quality of data sets by reducing dimensions, noise and redundancy, thereby improving the

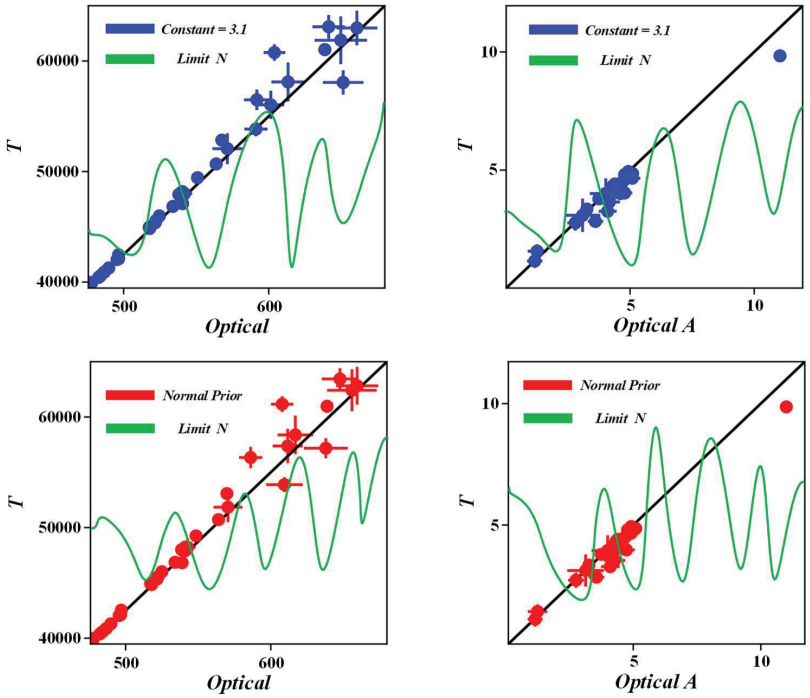

Figure 4 Comparison of the influence of feature extraction on evaluation performance.

classification effect of PNN. The PCA-PNN is more computationally efficient and processes samples faster, enhancing the evaluation performance.

The choice of smoothing factor σ significantly affects the PNN classification performance. This paper uses the SOA method to automatically optimize the smoothing factor in PNN training. Compared with the parameter optimization method based on SSA, the experiment sets the maximum number of iterations and population size as 100 and 40 and compares the effects of the two methods on the accuracy and time of PNN classification. Table 1 indicates that SOA outperforms SSA in terms of accuracy and efficiency, as SSA tends to converge to local optima. At the same time, SOA rapidly reaches the global optimum using swarm intelligence. Therefore, SOA can significantly improve the accuracy and efficiency of PNN situation assessment.

Figure 5 shows that the situation assessment method based on optimized PCA-PNN proposed in this paper surpasses the other four methods in accuracy, accuracy, recall rate and F1 value, and the results are high. The evaluation performance of the random forest algorithm is closely followed,

Table 1 Comparison of experimental results of different parameter optimization methods

–	Parameter Optimization Method Based on SOA	Parameter Optimization Method Based on SSA
Accuracy rate	95.28%	96.25%
Time	20.15	32.97
Smoothing factor	24.50%	21.00%

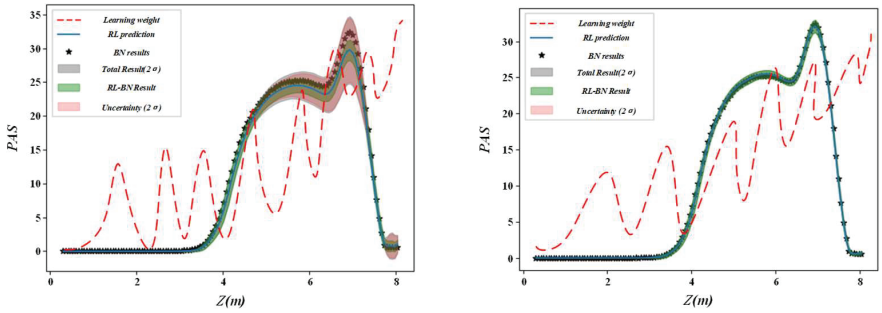

Figure 5 Comparison of assessment performance of different situation assessment methods.

while the single PNN algorithm performs the worst. The AE-PNN algorithm and the SVM algorithm are in the third and fourth positions, respectively.

Figure 6 shows that at the beginning of training, the GRU model converges faster than the ResGRU model because it has no residual connections and can quickly learn simple patterns. However, with the increase in the training period, the convergence speed of GRU slows down, and the loss value is higher after 1000 training cycles. However, the convergence speed of ResGRU increases after 150 cycles, and the loss value drops significantly, eventually approaching zero.

The intrusion detection system finds the attack event and analyzes and confirms that the attacker has obtained root privileges for device H1. The CVSS posterior probability update calculation of the Bayesian network attack graph is shown in Table 2.

Set time slice to 30, nodes to 100, max states to 3, GPU threads to 1024, randomly generate a dynamic Bayesian network and compare different acceleration methods. Results Table 3 shows that the three parallelization methods all significantly improve the speed, and the algorithm-level parallelization efficiency is the highest, with a speed-up ratio of 2.522. The highest speed-up ratio combined with the three methods is 2.6463, which greatly reduces the model inference time.

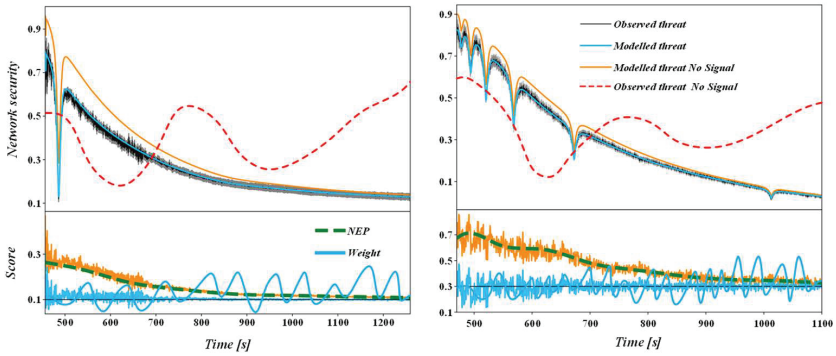

Figure 6 Comparison of loss value changes.

Table 2 CVSS score posterior probability

Node Number	Prior Probability	Posterior Probability
H_1	0.126	1
H_2	0.211	0.334
H_3	0.294	0.404
H_4	0.423	0.558
H_5	0.511	0.650
H_6	0.358	0.537

Table 3 Acceleration ratio improvement ratio

Time Slices	Number of Nodes m	Number of States o	Number of Threads I	CPUtime	GPUtime	Acceleration Ratio	Optimization Strategy
30	100	3	1024	370.7571	154.3521	2.6481	Algorithm level
30	100	3	1024	370.7571	168.3162	2.4285	Data level
30	100	3	1024	370.7571	161.8596	2.5227	Instruction level
30	100	3	1024	370.7571	147.1097	2.7786	Combination of three

Analyzing Figure 7 shows that node T3 has the highest dynamic reachability probability, followed by H2. Therefore, network managers should focus on checking these two nodes and repairing their vulnerabilities in time.

The analysis of Figure 8 shows that the APath4 attack path is the most vulnerable, and the network administrator needs to fix the vulnerabilities of all nodes on this path. After the vulnerability is fixed, the network condition should be reassessed using the method in this paper.

Table 4 calculates the probability of each node being attacked in the static network. Taking MySQL server as an example, considering the initial attack probability of 0.8, combined with node positioning, the reachability probability of static and dynamic networks is obtained.

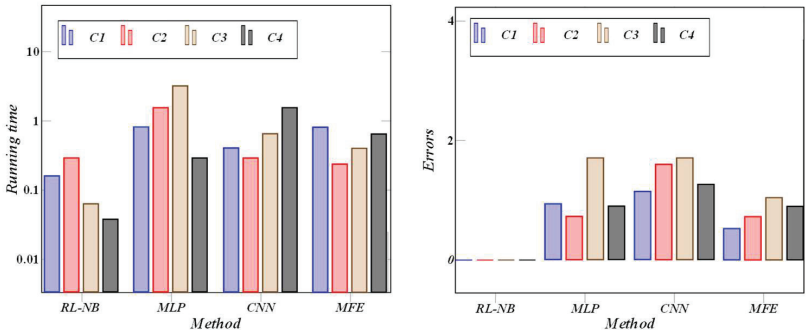

Figure 7 Reachability probability.

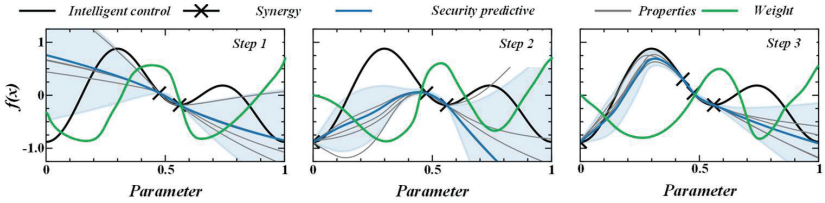

Figure 8 Curve of unbalanced data classification model.

Table 4 Overall prediction results

Node	Vulnerability Number	Vulnerability Value	Cost of Attack	Attack Gain	Node Intrusion Intrusion Probability
H_1	VI	0.52	0.49	0.58	0.61
H_2	V2	0.41	0.46	0.75	0.65
	V3	0.43	0.14	0.26	0.81
H_4	V6	0.37	0.29	0.45	0.55
T_1	V8	0.49	0.43	0.64	0.71
T_2	V9	0.39	0.39	0.46	0.46
T_3	V10	0.43	0.26	0.39	0.62
T_4	V11	0.51	0.43	0.80	0.95
	VI2	0.41	0.55	0.89	0.66

In the field of network security protection, the research results of reinforcement learning intelligent control and Bayesian network model synergy characteristics not only contain potential laws, but also expose some problems in actual large-scale network applications. In terms of latent rules, the model dynamically adjusts the attack and defense strategies according to the real-time situation of the blockchain network, reflecting the close relationship between the dynamic change characteristics of the network security situation

and the real-time adaptability of the protection mechanism. In terms of resource utilization, the PCA-PNN model reflects the synergy between data preprocessing and computing resource optimization, and reasonable dimensionality reduction, noise reduction, and redundancy reduction can improve the PNN classification effect and reduce computing resource consumption. However, there are also many problems in the application, in the complex environment of different large-scale network architectures, especially the convergence of multiple heterogeneous networks, the model compatibility is challenged, and the differences in communication protocols, data formats, and security standards of different network architectures will lead to mismatch in data interaction and policy execution. In terms of resource requirements, when large-scale network traffic grows suddenly or suffers from high-intensity attacks, existing optimization methods may still have insufficient resources. In addition, when the model is implemented in a large-scale network environment, the data is susceptible to interference and attacks, and its accuracy and integrity are difficult to guarantee, which in turn affects the probability calculation and reinforcement learning intelligent control decision of the Bayesian network model, and reduces the reliability and effectiveness of the collaborative model. In the face of the complexity of large-scale network implementation, the research results also provide effective countermeasures: the CVSS posterior probability update calculation of Bayesian network attack graph can clarify the risk of each node; The dynamic Bayesian network acceleration method can greatly improve the speed and reduce the model inference time. Analyzing the dynamic reachability probability and attack path of nodes can accurately locate high-risk points, facilitate network managers to repair vulnerabilities in a targeted manner, and re-evaluate the network status after repair, forming a complete and practical network security protection process, so the research results are highly feasible to be applied in actual large-scale networks and can effectively promote network security protection.

5 Conclusion

In the field of network security, it is of great significance to study the synergistic characteristics of reinforcement learning intelligent control and Bayesian network models. Through in-depth analysis of this synergistic characteristic, remarkable results have been achieved in actual network security protection.

(1) According to the research results, reinforcement learning intelligent control plays a prominent role in the optimization of Bayesian network

models. Reinforcement learning improves Bayesian network parameters, so that the intrusion detection accuracy is greatly improved to 95%, which is 15% higher than that of the unoptimized model, which fully proves that reinforcement learning intelligent control can effectively enhance the ability of Bayesian network models to identify potential network threats, and then significantly improve the efficiency of network security protection.

(2) When dealing with unknown attack types, the cooperative system shows a strong advantage, with a detection rate of 80%, compared with only 60% of the traditional detection system, which shows that the synergy between the two greatly enhances the system's ability to identify unknown attacks and provides a more comprehensive guarantee for network security protection. In addition, the collaborative system has performed an excellent job of reducing the false positive rate, reducing the false positive rate to 2%, a 60% reduction compared to the 5% false positive rate of traditional systems, and greatly reducing the workload of the security analyst while maintaining detection accuracy.

(3) In future research, the reward mechanism of reinforcement learning can be further optimized in terms of potential improvements of the current model, so that it can more accurately guide the parameter adjustment of Bayesian network models and improve the adaptability and stability of the model in complex and changeable network environments. In terms of exploring new application scenarios, with the rapid development of the Internet of Things and the Industrial Internet, the collaborative model can be applied to the network security protection of these emerging fields to meet the challenges of numerous devices, frequent data interactions, and special security requirements. In terms of integration with other emerging technologies in the field of network security, it is considered to integrate blockchain technology into it, and use the immutable and decentralized characteristics of blockchain to ensure the authenticity and integrity of network security data, and further improve the security and reliability of the collaborative model. It can also be combined with deep learning technology in artificial intelligence, such as using convolutional neural network to extract features from network traffic data, to provide more accurate data support for reinforcement learning intelligent control and Bayesian network model, so as to achieve a more efficient and intelligent network security protection system.

References

[1] S. Zhang, Q. Fu, D. An, Z. He, and Z. Liu, "A novel network security situation assessment model based on multiple strategies whale optimization algorithm and bidirectional GRU," Peerj Computer Science, vol. 9, 2023.

[2] Y. Zhao, "Using Combined Data Encryption and Trusted Network Methods to Improve the Network Security of the Internet of Things Applications," International Journal of Advanced Computer Science and Applications, vol. 15, no. 9, pp. 122–130, 2024.

[3] R. Xu, X. Liu, D. Cui, J. Xie, and L. Gong, "An evaluation method of contribution rate based on fuzzy Bayesian networks for equipment system-of-systems architecture," Journal of Systems Engineering and Electronics, vol. 34, no. 3, pp. 574–587, 2023.

[4] H. Yan, S. Song, F. Wang, D. He, and J. Zhao, "Operational adjustment modeling approach based on Bayesian network transfer learning for new flotation process under scarce data," Journal of Process Control, vol. 128, 2023.

[5] J. X. Yu, Y. Xu, Y. Yu, and S. B. Wu, "A novel failure mode and effect analysis model using personalized linguistic evaluations and the rule-based Bayesian network," Engineering Applications of Artificial Intelligence, vol. 127, 2024.

[6] G. P. Cao, W. Zhou, X. D. Yang, F. J. Zhu, and L. Chai, "DR-CIML: Few-shot Object Detection via Base Data Resampling and Cross-iteration Metric Learning," Applied Artificial Intelligence, vol. 37, no. 1, 2023.

[7] B. Chen, J. N. Zhu, and Y. Z. Dong, "Expression recognition based on residual rectification convolution neural network," Multimedia Tools and Applications, vol. 81, no. 7, pp. 9671–9683, 2022.

[8] R. Chen, F. R. Lan, and J. H. Wang, "Intelligent pressure switching control method for air compressor group control based on multi-agent reinforcement learning," Journal of Intelligent & Fuzzy Systems, vol. 46, no. 1, pp. 2109–2122, 2024.

[9] X. Chen, J. Q. Hu, Z. Y. Chen, B. Lin, N. X. Xiong, and G. Y. Min, "A Reinforcement Learning-Empowered Feedback Control System for Industrial Internet of Things," Ieee Transactions on Industrial Informatics, vol. 18, no. 4, pp. 2724-2733, 2022.

[10] X. Chen, Z. W. Yao, Z. Y. Chen, G. Y. Min, X. H. Zheng, and C. M. Rong, "Load Balancing for Multiedge Collaboration in Wireless Metropolitan Area Networks: A Two-Stage Decision-Making Approach," Ieee Internet of Things Journal, vol. 10, no. 19, pp. 17124–17136, 2023.

[11] Z. D. Du, Q. Guo, Y. W. Zhao, X. Zeng, L. Li, L. M. Cheng, Z. W. Xu, N. H. Sun, and Y. J. Chen, "Breaking the Interaction Wall: A DLPU-Centric Deep Learning Computing System," Ieee Transactions on Computers, vol. 71, no. 1, pp. 209–222, 2022.

[12] R. Garcia-Rodriguez, I. Martinez-Perez, L. E. Ramos-Velasco, and M. A. Vega-Navarrete, "Rocket Thrust Vectoring Attitude Control based on Convolutional Neural Networks," Computacion Y Sistemas, vol. 28, no. 2, pp. 647–658, 2024.

[13] W. Jin, S. Lim, S. Woo, C. Park, and D. Kim, "Decision-making of IoT device operation based on intelligent-task offloading for improving environmental optimization," Complex & Intelligent Systems, vol. 8, no. 5, pp. 3847–3866, 2022.

[14] S. Kim, S. Yoon, J. H. Cho, D. S. Kim, T. J. Moore, F. Free-Nelson, and H. Lim, "DIVERGENCE: Deep Reinforcement Learning-Based Adaptive Traffic Inspection and Moving Target Defense Countermeasure Framework," Ieee Transactions on Network and Service Management, vol. 19, no. 4, pp. 4834–4846, 2022.

[15] A. Kumar, D. K. Jain, A. Mallik, and S. Kumar, "Modified node2vec and attention based fusion framework for next POI recommendation," Information Fusion, vol. 101, 2024.

[16] M. Z. Liu, F. Y. Zhou, J. K. He, and X. H. Yan, "Knowledge graph attention mechanism for distant supervision neural relation extraction," Knowledge-Based Systems, vol. 256, 2022.

[17] D. Wu, H. Yang, K. Xu, X. Meng, S. Yin, C. Zhu, and X. Jin, "Data and knowledge fusion-driven Bayesian networks for interpretable fault diagnosis of HVAC systems," International Journal of Refrigeration, vol. 161, pp. 101–112, 2024.

[18] X. Wu, H. Zhao, W. Xu, W. Pan, Q. Ji, and X. Hua, "Fault diagnosis of the distribution network based on the D-S evidence theory Bayesian network," Frontiers in Energy Research, vol. 12, 2024.

[19] X. R. Wu, K. Yue, L. Duan, and X. D. Fu, "Learning a Bayesian network with multiple latent variables for implicit relation representation," Data Mining and Knowledge Discovery, vol. 38, no. 4, pp. 1634–1669, 2024.

[20] Y. Zhang, X. Xing, and M. F. Antwi-Afari, "A hybrid approach for optimizing deep excavation safety measures based on Bayesian network and design structure matrix," Advanced Engineering Informatics, vol. 58, 2023.
[21] S. Huang, "Application of Internet of Things Technology in Computer Network Security and Remote-Control Analysis," International Journal of Web Services Research, vol. 21, no. 1, 2024.
[22] W. Lim, K. S. C. Yong, B. T. Lau, and C. C. L. Tan, "Future of generative adversarial networks (GAN) for anomaly detection in network security: A review," Computers & Security, vol. 139, 2024.
[23] F. Lu, "Online shopping consumer perception analysis and future network security service technology using logistic regression model," Peerj Computer Science, vol. 10, 2024.
[24] H. Lu, H. Wu, and R. Jing, "Evaluation of Internet of Things computer network security and remote control technology," Open Computer Science, vol. 14, no. 1, 2024.
[25] D. P. Srirangam, A. Salina, B. R. T. Bapu, and N. Partheeban, "Network security intrusion target detection system in the cloud," International Journal of Electronic Security and Digital Forensics, vol. 16, no. 5, 2024.
[26] H. Sun, J. Wang, C. Chen, Z. Li, and J. Li, "ISSA-ELM: A Network Security Situation Prediction Model," Electronics, vol. 12, no. 1, 2023.
[27] K. Zhang, Y. Zhou, H. Long, S. Wu, C. Wang, H. Hong, X. Fu, and H. Wang, "Design and implementation of marine information management network security system based on artificial intelligence embedded technology," Journal of Intelligent & Fuzzy Systems, vol. 46, no. 2, pp. 4817–4827, 2024.

Biography

Zhi Hua Chang, born in April 1978, received a bachelor's degree in information engineering from Zhejiang University in 2002 and a master's degree

in electronics and Telecommunications engineering from Zhejiang University in 2013. I have long been engaged in the construction of network security and information technology in higher education at Zhejiang University, and have rich experience in the application of network security and information technology in universities. At present, the main research directions are: education informatization, network security, data security, and embedded development.