www.cad-journal.net
Computer-Aided Design and Applications™

Application of Reinforcement Learning in Network Security in Dynamical Adjusting Defense Strategies

Ying Li[1], Lijuan Yang[2] and Haiyan Liu[3]

[1,2,3] College of Computer, North China Institute of Aerospace Engineering, LangFang 065000, China,
[1]lynciae@163.com, [2]ylj23@nciae.edu.cn, [3]liuhy@nciae.edu.cn

Corresponding author: Ying Li, lynciae@163.com

Abstract. With the update of new network technologies and the continuous changes in network attack methods, traditional network security defense models can no longer fully cope with large-scale network attacks. Therefore, by deeply analyzing the basic principles and characteristics of reinforcement learning algorithms, combined with the practical needs in the field of network security, this paper proposes an intelligent defense system based on reinforcement learning. This system can generate self-learning network attack features through self-immune dynamic attacks, dynamically adjust defense strategies, and effectively respond to complex and changing network threats. The experimental results show that the model proposed in this paper can effectively determine the optimal attack path in large-scale network attack environments and achieve the goal of dynamically adjusting defense strategies under different network attack intensities through multi-agent joint methods, with good stability listed in the model table. In addition, the model proposed in this article can effectively detect the most harmful network attack paths, conduct targeted internal network security checks, strengthen internal network defense vulnerabilities, and help improve the overall security defense of the internal network.

Keywords: Reinforcement Learning; Computer-Aided; Network Security; Offensive and Defensive Game; Autoimmune Dynamic Attack
DOI: https://doi.org/10.14733/cadaps.2025.S7.1-13

1 INTRODUCTION

In the context of the Internet era, the importance of cyberspace has climbed to an unprecedented strategic height. Its far-reaching influence has not only reshaped the way of life, work paradigm, and social structure of mankind but has also become the core field of maintaining national security, driving economic prosperity, promoting cultural exchanges, and optimizing social governance [1]. With the rapid evolution of cutting-edge technologies such as cloud computing, mobile computing, and the Internet of Things, the network environment is undergoing unprecedented changes. Although this process greatly expands the application boundaries of information technology, it also introduces more complex and ever-changing security challenges, forcing network security protection to become an indispensable part of sustainable development in cyberspace [2]. The traditional network security technology system is rooted in a relatively static and specific network environment threat model, designed to resist known types of attacks. However, in the face of the new attack surface and potential vulnerabilities brought about by the flexibility of cloud computing, the popularity of mobile devices, and the surge of IoT devices, traditional defense mechanisms have shown a lack of effort. Especially with the rise of Advanced Persistent Threats (APTs), this type of attack, relying on novel attack methods and technological paths, can cleverly evade traditional detection mechanisms based on pre-set rules or patterns, posing a severe challenge to traditional defense systems [3]. In addition, traditional network security measures are often accompanied by complex configuration processes and user operation requirements, which, to some extent, sacrifice the smoothness of user experience. Users may choose to sacrifice some security settings in pursuit of operational convenience, which invisibly increases the risk of exposure of the system to attacks [4]. Furthermore, traditional network security architectures often rely on hardware devices such as network firewalls and intrusion detection systems as security barriers, but these devices themselves are not flawless, and their inherent vulnerabilities become targets coveted by attackers. Once the defense line is breached, the entire network security system will face the risk of complete collapse. It is particularly noteworthy that with the rapid increase in the scale and complexity of network attacks, traditional defense methods have shown significant shortcomings in resource consumption and response speed, making it difficult to resist large-scale attacks in situations where resources are limited effectively, and may ultimately lose their defense capabilities due to resource depletion. At the level of protection strategy, traditional methods focus on the deployment and application of static rules, which are difficult to adapt to rapidly evolving network environments flexibly and constantly upgrading threat situations, and their limitations are becoming increasingly prominent [5].

The methods of cyber attacks are constantly evolving, and traditional static defense mechanisms are often difficult to deal with. Reinforcement learning algorithms can dynamically adjust defense strategies based on changes in attack behaviour, effectively resisting new threats. When discussing the increasingly severe network threats faced by Cyber-Physical Systems (CPS), it must be emphasized that as CPS network connectivity increases, its risk of exposure to potential network attacks also increases [6]. Therefore, designing and implementing efficient and intelligent automatic network security defense mechanisms is crucial for ensuring the secure and stable operation of CPS. Reinforcement learning, as a machine learning method, enables agents to continuously learn and optimize their behavioral strategies through interaction with the environment in order to adapt to complex and ever-changing network environments. How to efficiently utilize limited defense resources is key. Reinforcement learning algorithms can learn how to allocate security resources optimally, such as firewall rules, the sensitivity of intrusion detection systems (IDS), etc., to achieve the best security protection effect [7]. These attacks may not only cause environmental damage but also result in significant casualties and economic losses. By analyzing historical data and the current network state, reinforcement learning algorithms can predict potential attack paths and patterns, deploy defense measures in advance, and reduce the risk of being attacked. In this context, the application of computer-assisted reinforcement learning (RL) algorithms in network security defense has shown great potential and value. Intelligent data preprocessing and analysis: By automating data cleaning, feature extraction, and anomaly detection, high-quality training data is provided for reinforcement learning algorithms [8]. The application of computer-assisted reinforcement learning algorithms in network security defense relies on powerful computing power and efficient data processing algorithms. In the process of constantly engaging in a "game" with attackers, reinforcement learning algorithms can accumulate experience, continuously optimize their defense strategies, and form a virtuous cycle of "getting braver with more battles" [9]. The network environment is highly complex and dynamically changing, making it a major challenge to construct a simulation environment that accurately reflects the actual situation. Improve the interpretability of reinforcement learning models to facilitate security experts' understanding and trust in their decision-making process. Support large-scale data processing and training of complex models to ensure efficient operation of reinforcement learning algorithms. Such as distributed training, model pruning, etc., to reduce training time and improve model performance.

In the application of network security defense, reinforcement learning algorithms continuously learn and optimize their strategies through the interaction between intelligent agents and the environment, enabling them to adapt to the evolution of network threats autonomously. This ability enables defense systems to respond to new and unknown attacks rather than relying solely on pre-set rules and patterns. Faced with complex and ever-changing network environments, reinforcement learning algorithms can adjust their defense strategies in real time to cope with constantly changing threat situations [10]. Compared to traditional pattern-matching methods, it can identify new types of attacks faster and reduce false positives and false negatives. The goal of reinforcement learning algorithms is to maximize cumulative rewards, that is, to achieve the best long-term results in network defense processes. Through continuous trial and error and learning, algorithms can find the most effective defense strategies and improve the intelligence level of defense systems. Through self-play models, diverse defense strategies can be trained to increase the difficulty of opponents' attacks. This article will combine reinforcement learning algorithms to construct a network security defense model. However, considering that reinforcement learning may lead to low utilization of network vulnerability information in solving attack path methods, this article constructs a self-immune dynamic attack generation module based on a biological immune mechanism. By simulating the attacker's network, the attacker discovers problems in the network system in a short period of time and then uses the reinforcement learning defense decision module combined with the attack defense game model for security defense and targeted defense adjustments.

2 RELATED WORK

The development of network security defense is a constantly evolving process, accompanied by continuous technological innovation and increasingly complex threats. In this process, various algorithms and technical means played a key role. The symmetric encryption algorithm is one of the earliest encryption algorithms, characterized by using the same key for encryption and decryption. The advantage of this algorithm is its fast encryption speed and suitability for encrypting large amounts of data. However, its disadvantage is that the transmission and management of keys are relatively complex, and once the key is leaked, the security of encrypted data will be threatened. With the advancement of technology, symmetric encryption algorithms are gradually being replaced by more secure encryption algorithms. Muradova and Khaytbaev [11] solved the problem of key transmission and management in symmetric encryption algorithms with the emergence of asymmetric encryption algorithms. In this algorithm, encryption and decryption use different keys, namely public key and private key. Public keys can be publicly disseminated, while private keys are kept confidential. Only those who possess the private key can decrypt data encrypted with the public key. Asymmetric encryption algorithms have high security, but their encryption speed is relatively slow. At present, RSA, ECC, and other algorithms are representative of the field of asymmetric encryption. In order to combine the advantages of symmetric encryption and asymmetric encryption, a hybrid encryption system is proposed. In this system, asymmetric encryption algorithms are used to transmit symmetric encryption keys securely, and then symmetric encryption algorithms are used to encrypt and decrypt large amounts of data. This method ensures data security and improves encryption efficiency. With the continuous development of technology, biometric technology has been widely applied in various fields. Nguyen et al. [12] used personal biometric features for identity verification, such as fingerprint recognition, retinal recognition, facial recognition, etc. These features are unique and non-replicable. Therefore, biometric technology has high security.

In recent years, with the urgent demand for efficient defense mechanisms in the field of network security, the establishment of process optimization, mechanism innovation, and rapid response indicator systems has significantly enhanced the protection capability of network security. To overcome these obstacles, computer-aided reinforcement learning algorithms have shown great potential in network security defense. Nguyen and Reddi [13] proposed an innovative synthetic sample generation method based on this background, which focuses on the synthesis of malicious portable executable binary files (PE files) and DoS network attacks. These challenges not only require technological breakthroughs but also urgently require efficient data support. The core of this method lies in a carefully designed reinforcement learning engine that can deeply learn baseline data of different malware families and DoS attack characteristics and generate new, mutated, and powerful samples. This process not only reduces dependence on real environmental resources but also improves sample diversity and representativeness by simulating attack behaviour. Reinforcement learning (RL) is a method of learning to take optimal actions in a specific environment through trial and error. CARL further integrates computer technologies such as feature engineering and parallel computing to optimize the learning process and enhance defense efficiency. Feature selection also plays a crucial role in applying reinforcement learning to network security defense. ISSA, as an optimized feature selection method, is characterized by improving population diversity and introducing new local search algorithms to optimize the selection of feature subsets. Therefore, through feature selection techniques such as the improved Salp Swarm algorithm (ISSA) proposed in this paper, noise and irrelevant features can be effectively removed, data dimensionality can be reduced, and the learning efficiency and defense performance of reinforcement learning models can be improved. In the context of network security defense, ISSA can help identify the most critical feature combinations for identifying malicious behaviour, thereby constructing more accurate and efficient defense models. To verify the effectiveness of ISSA in network security defense, it can be applied to network security datasets (such as KDD Cup 99, NSL-KDD, etc.) and compared with other optimization algorithms (such as genetic algorithm, particle swarm optimization algorithm, etc.). Evaluation metrics can include accuracy, recall, F1 score, and model training time to comprehensively assess the advantages of ISSA in improving defense performance and reducing computational costs. Its characteristic lies in the ability to dynamically adjust strategies based on environmental feedback, making it very suitable for dealing with complex and constantly changing network security threats. Network security data often contains a large number of redundant, irrelevant, and even misleading features, which not only increase computational complexity but may also reduce the accuracy and generalization ability of the model. However, in the face of the most challenging tasks - detecting, classifying, and eradicating malware, as well as defending against denial of service (DoS) network attacks, traditional defense methods seem inadequate. Representative samples are still a time-consuming and labor-intensive process involving complex exploration, sandbox environment configuration, and large data storage requirements, often resulting in imbalanced or unrepresentative sample sets. Shah et al. [14] found that reinforcement learning can optimize the sample generation process, reduce the cost of establishing a controlled environment, and improve the efficiency of sample evaluation through automation and intelligence. Specifically, the introduction of synthetic sample generation technology not only reduces the difficulty of obtaining real samples but also simulates and predicts new attack patterns to a certain extent, providing strong support for the formulation of defense strategies.

With the popularization of the Internet of Things (IoT) and the significant increase in consumer density, IoT systems are facing unprecedented security challenges. Regarding the issue of deception attacks in the Internet of Things, Tubishat et al. [15] proposed an innovative design scheme that combines reinforcement learning algorithms. Aim to effectively detect fraudulent behaviour by dynamically analyzing the probability distribution of received power in the area where mobile users are located. In response to this severe situation, it thoroughly explored potential vulnerabilities in the security of the Internet of Things environment, especially those that are insensitive to transmission conditions and easily exploitable. Especially threats from various identity-based attacks. These attacks are not only complex and diverse but also become even more difficult to prevent due to the widespread use of low-power access nodes. With the assistance of computers, reinforcement learning algorithms have shown great potential in network security defense due to their strong adaptability and learning ability. In addition, we also studied the impact of observer presence and absence on the confidentiality scope of target consumers to comprehensively evaluate the security status of the IoT environment. This design not only improves the accuracy of detection but also enhances the system's ability to respond quickly to changes in attacks. This algorithm utilizes fuzzy logic to process sensitive area information, combined with reinforcement learning to continuously optimize detection strategies, ensuring efficient and accurate detection and protection in areas with the highest attack opportunities. We validated the effectiveness and stability of the proposed design in different environments by simulating actual scenarios in three different regions. In terms of energy management, we compared various security algorithms designed for different modes and found that the MTFLA algorithm exhibits significant energy efficiency advantages while ensuring security. It effectively reduces the energy prerequisite for encrypting data, enabling IoT devices to operate for longer periods of time while ensuring security and extending device lifespan.

In terms of defense algorithms, firewalls are the first line of defense for network security. With the development of technology, the functions of firewalls are becoming increasingly powerful. They can not only perform simple packet filtering and state detection but also achieve various functions such as deep packet detection and application layer filtering. Unlike firewalls, intrusion detection technology is an active defense technique. It detects abnormal behaviour and potential threats through real-time monitoring and analysis of network traffic and issues timely alerts. Intrusion detection technology can be divided into two types: signature-based detection and behavior-based detection. The former detects threats by matching known attack patterns, while the latter detects anomalies by analyzing the behavioural characteristics of network traffic. With the development of intelligent technology, researchers have trained machine learning models to automatically identify abnormal patterns and potential threats in network traffic, thereby improving the accuracy and efficiency of defense. For example, machine learning algorithms can be used to classify and detect malicious software or to predict and prevent network attacks. Machine learning algorithms can also analyze system behaviour and identify sequential behaviour instances that are unrelated to typical network behaviour. By building behavioural baselines, algorithms can alert security analysts about the transmission of payloads intended to exploit vulnerabilities, thereby taking timely defensive measures. In the detection of image malware and analysis of network behaviour, researchers have introduced convolutional neural networks to extract features from input data through convolutional layers, pooling layers, and fully connected layers. Researchers also use SVM to detect abnormal network traffic or malware samples and train models to distinguish between normal and abnormal data. In the field of network security, random forests can be used to detect network intrusions, malicious software, etc., by building multiple decision trees to identify various characteristics of threats. In summary, the development of network security defense is a constantly evolving process. In this process, various algorithms and technological means continue to innovate and improve, providing a more comprehensive and effective guarantee for network security. In the future, with the continuous development of technology and the increasing complexity of threats, network security defense will continue to develop in a more intelligent, automated, and collaborative direction.

3 CONSTRUCTION OF NETWORK SECURITY DEFENSE MODEL

3.1 Self-Immune Dynamic Attack Generation Module Based on Reinforcement Learning

The self-immune dynamic attack defense generation module is rooted in advanced reinforcement learning algorithm frameworks, ingeniously integrating the concept of the natural immune system into it, constructing an intelligent dynamic defense system that can autonomously perceive the environment, continuously learn and evolve, and effectively resist network threats. The core of this module is that the precise assessment of the host security situation relies closely on the quantitative analysis of host vulnerabilities. Among them, the rating level of vulnerabilities shows a significant negative correlation with the duration of potential attackers' attacks - the higher the vulnerability rating, the lower the risk of being quickly breached and the longer the required attack time. At the same time, these high scoring vulnerabilities are positively correlated with the permission values that attackers can obtain once they succeed, emphasizing the serious consequences of high-risk vulnerabilities being exploited, which will significantly increase the attacker's permissions. This design enables the self-immune module to intelligently predict risks, implement precise policies, and effectively enhance the proactivity and effectiveness of network defense. Therefore, in this module, the reward matrix values of the Q-learning algorithm are determined based on the vulnerability assessment scores of network devices. The expression for the relationship between vulnerability rating and weight is shown in (1):

$$\omega_{xy} = max(\varphi) + min(\varphi) - \varphi_{y} \tag{1}$$

In the formula, the source node and pointing node are represented as x and y respectively, the weight of the $x \rightarrow y$ node edge is represented as ω_{xy} , and the vulnerability score value is represented as φ .

After weight conversion, the goal of the Q-learning algorithm is opposite to the original goal, and the value function needs to be modified accordingly. That is, after modification, each selection will be the minimum value in the expected values, which will still remain as a convergence function, as shown in formula (2):

$$Q(s,a) \leftarrow Q(s,a) + \alpha(R + \gamma min_{a'} Q(s',a') - Q(s,a)) \tag{2}$$

Due to the fact that the generation of immune dynamic attack paths does not require consideration of the length of the path, $\gamma = 1$.

In the original optimal attack path loop solution approach, unknown loops are prone to occur in random direction selection actions due to limited training times. In such cases, increasing the number of nodes and directed edges in the topology network can solve the problem. However, the self-immune dynamic attack generation module uses network host information in a weighted manner. If the network attacker uses the direction of network communication to obtain the starting point of information transmission, the constructed attack graph requires larger weighted directed edges. In addition, under normal circumstances, if the trained nodes of a reinforcement learning algorithm have two mutually minimum expected values, it will form a dead loop. To avoid such problems, this article sets the selection of the next optimal action when the corresponding state of the selected action is in the current attack path. If the selection of the next optimal action is invalid, it will be removed from the combination of optional actions and continue to select the next optimal action.

3.2 Reinforcement Learning Defense Decision Module Combined with Attack Defense Game Model

Network defense decision-making refers to the comprehensive use of various technological means and management measures in the field of cybersecurity to predict and analyze potential threats to the network, and develop corresponding defense strategies to ensure the safe and stable operation of the network system. This process includes not only the prevention of known threats but also early warning and response to unknown threats. The choice of defense strategy is based on the means of network attack suffered while ensuring network security and normal service and selecting the strategy with the highest profit value. The adoption of non-defensive measures in the face of different attack behaviours requires different defense costs. Reasonably utilizing the effects and impacts generated by non-defensive measures in the defense process can achieve better dynamic defense effects. This process of constantly seeking maximum benefits in the confrontation process is a multi-stage game. As shown in Table 1, the effects and impacts of different defense measures on specific attacks are presented.

	Block the source IP address.	*IP address jump*	*Disable the attacked port*
I P/ Port scanning password cracking.	Defense successful	Defense successful	Defense failure
	Almost no impact on service	Affects service	Cannot serve
DDoS	Defense successful	Defense successful	Defense successful
	Almost no impact on service	Cannot serve	Cannot serve

Rebound shell	Defense successful	Defense failure	Defense successful
	Almost no impact on service	Affects service	Almost no impact on service

Table 1: The effects and impacts of different defense measures on specific attacks.

Defense decisions can be transformed into Markov decision processes, where the state space is represented as S, the intelligent action set is represented as A, the corresponding utility function is represented as R, the environmental state transition function is represented as T, and the quadruple is represented as $<S,A,R,T>$. A single strategy is defined as $\pi : S \times A \to [0,1]$, and when the agent is in the S state, the state value function is shown in formula (3):

$$V^{\pi}(s) = E_{\pi}[\sum_{k=0}^{\infty} \gamma_k r_{t+k+1} |\ s_t = s] \tag{3}$$

In the formula, the expected value is represented as E, and the future time steps are represented as k.

Further, the state value function can be obtained from the Bellman equation as shown in (4):

$$V^{\pi}(s) = \sum_{a} \pi(s,a) \sum_{s'} P(s,a,s')[R(s,a,s') + \gamma V^{\pi}(s')] \tag{4}$$

The formula $P(s,a,s') \in [0,1]$ represents the possibility of state transition s', and the degree of importance that the agent places on benefits is represented as γ. The expected return is expressed as $R(s,a,s')$, and its expression is shown in (5):

$$R(s,a,s') = E^{\pi}\{r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + ... \big| s_t = s, a = a_t = \pi(s_t), s_{t+1} = s'\} \tag{5}$$

The state behaviour function is shown in (6):

$$Q^{\pi}(s,a) = \sum_{k=0}^{\infty} \gamma_k r_{t+k+1} |\ s_t = s, a_t = a \tag{6}$$

From the Bellman equation, we can obtain (7):

$$Q^{\pi}(s,a) = E_{\pi}\{r_{t+1} + \gamma Q^{\pi}(s',a') \big| s,a\} \tag{7}$$

The utility matrix composed of the utility sets of each atomic attack corresponding to each anti-bone strategy is shown in (8):

$$R = \begin{bmatrix} r_{1,1} & r_{1,2} & \cdots & r_{1,m} \\ r_{2,1} & r_{2,2} & \cdots & r_{2,m} \\ \vdots & \vdots & \ddots & \vdots \\ r_{n,1} & r_{n,2} & \cdots & r_{n,m} \end{bmatrix} \tag{8}$$

In the process of offensive and defensive games, defenders need to engage in decision-making. Therefore, this article uses the Q-learning algorithm to achieve the goal of learning experience for decision agents. Enable decisions in different states to achieve the maximum expected value in the future. The Q-value update formula is shown in (9):

$$Q(s_t,a_t) \leftarrow Q(s_t,a_t) + \alpha[r_{t+1} + \gamma \underset{a}{max} Q(s_{t+1},a) - Q(s_t,a_t)] \tag{9}$$

In the formula, the time node is represented as t, the expected reward is represented as $Q(s_t,a_t)$, the learning rate is denoted as a, and the decay rate is denoted as γ.

The long-term accumulation discount reward and maximum expression of the optimal strategy chosen by the intelligent agent under any state and time conditions are shown in (10):

$$V^*(s) = \max_{\pi} E_{\pi}[\sum_{k=0}^{\infty} \gamma_k r_{t+k}|\ s_t = s] \tag{10}$$

The equivalent representation of its learning objectives is shown in (11):

$$Q^*(s,a) = \max_{\pi} E_{\pi}[\sum_{k=0}^{\infty} \gamma_k r_{t+k}|\ s_t = s, a_t = a] \tag{11}$$

In the formula, the long-term cumulative discount reward is expressed as $Q^*(s,a)$.

The optimal value functions under the optimal strategy are shown in (12) and (13) respectively:

$$V^*(s) = \max_{\pi} V_{\pi}(s) \tag{12}$$

$$Q^*(s,a) = \max_{\pi} Q_{\pi}(s,a) \tag{13}$$

Based on this optimal strategy, its solution only requires solving the optimal value function. The recursive relationship of the value function is shown in formulas (14) and (15):

$$V_{\pi}(s) = E_{\pi}[r_{t+1} + \gamma V_{\pi}(s_{t+1})\big|s_t = s] \tag{14}$$

$$Q_{\pi}(s,a) = E_{\pi}[r_{t+1} + \gamma Q_{\pi}(s_{t+1},a_{t+1})\big|s_t = s, a_t = a] \tag{15}$$

4 EXPERIMENTAL RESULTS

In order to test the application performance of the network security defense model based on the reinforcement learning algorithm, this paper constructs the target intranet according to the optimal attack path scheme of the model. The network security defense effect is verified by simulating the internal structure of the network. The network topology diagram of the optimal attack path scheme experiment is shown in Figure 1.

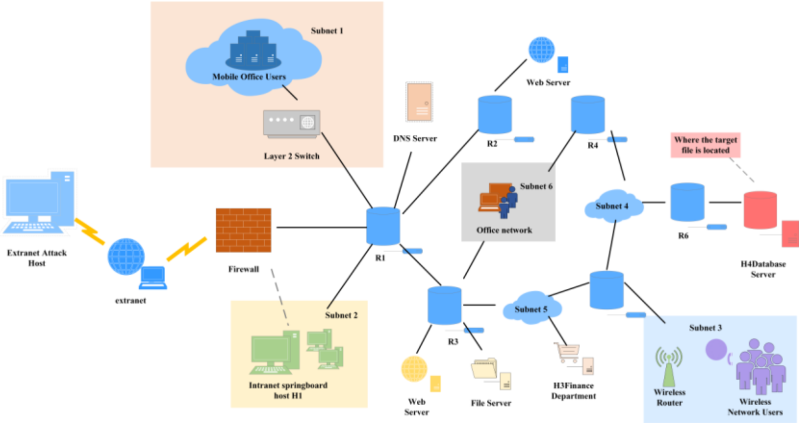

Figure 1: Schematic diagram of optimal attack path scheme experimental network topology.

In this experiment, virtual machines were mainly used to simulate network attacks. Although the network configuration of virtual machines is more convenient in this state, there are certain differences in the network attacks they receive compared to the actual network environment. Virtual machines are limited by host resources and it is difficult to simulate large-scale and complex network attack situations. In addition, compared with the security devices and firewalls in actual networks, there are also gaps in the corresponding devices in virtual experiments. Therefore, in the experimental process, this article will try its best to simulate the real environment, narrow the gap between the two, and ensure the accuracy and reliability of the experiment as much as possible.

In order to test the efficiency of the model application in this article, two other models were selected for efficiency comparison experiments based on relevant literature. The results are shown in Figure 2. The results in Figure 2(a) are the time consumption results of three models under the same conditions and the same number of iterations. The results showed that all three models would experience significant time consumption before the 5th iteration, but as the number of iterations increased, the slope of the time consumption curve gradually decreased, indicating that the consumption rate remained relatively stable. Compared with the other two models, the time consumption of this model is significantly lower, and the smoothness of the time curve is higher. This indicates that the time consumption generated by the model in this article is relatively stable during each iteration, and the minimum Q-value method has played a good role in excluding non-critical paths. Figure 2(b) shows the convergence state of the Q value under the optimal path condition. The results show that Model A has the fastest convergence speed among the three models, but the convergence value is relatively high, indicating that it may be in a local optimal state. Similarly, although Model B has a good convergence speed, its final convergence value is relatively high, and it may have fallen into a local optimal state. The convergence speed of this model is relatively fast, and the final convergence value is significantly lower than other models. The comprehensive experimental results show that this article exhibits the best time consumption performance and convergence stability among the three models, with stronger practicality.

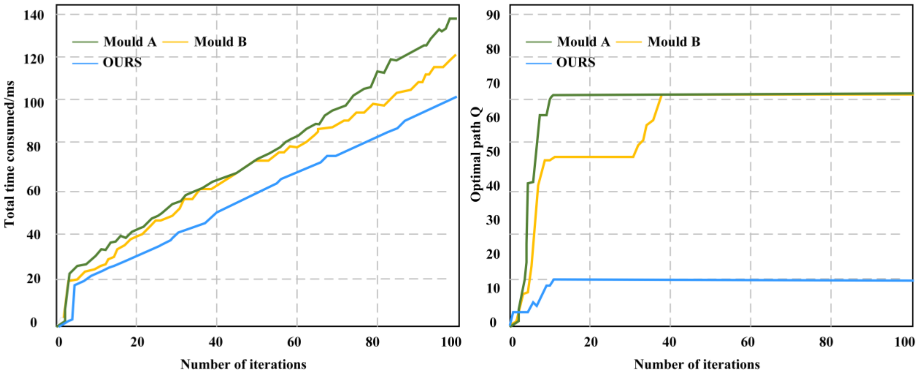

Figure 2: Experimental results comparing the efficiency of three models.

In order to further verify the efficiency of the model, this paper simulated the topology conditions of a large network during the experiment and tested the running efficiency of the model in it. The results are shown in Figure 3. In this experiment, starting with a sequence of 100 nodes, each group of 100 nodes gradually increases, and the node size, edge size, and time consumption for finding the optimal attack path of the model are statistically analyzed. The results show that the model proposed in this paper can effectively reduce the total time required to determine the optimal attack path in large-scale network attack environments, and as the path deepens, its time consumption does not increase exponentially. This indicates that compared with previous methods for finding the optimal attack path, the model proposed in this paper has certain advantages and exhibits better application performance.

The model in this article adopts a multi-agent reinforcement learning guidance form in terms of defense strategy distribution. In order to test the performance of the model, this article compares it with the results of single-agent reinforcement learning, as shown in Figure 4. The results show that the distribution of network defense strategies in the context of single-agent reinforcement learning is uneven and limited, making it difficult to adapt to the diverse range of network attack methods.

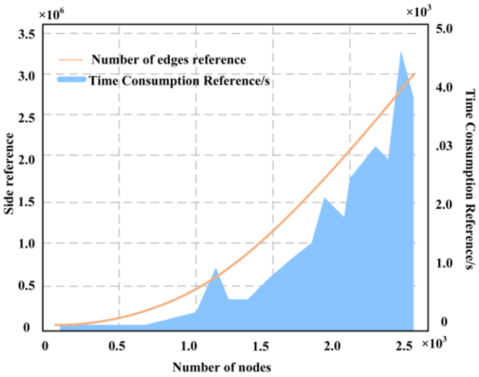

Figure 3: Efficiency experiment statistical results.

In the case of multi-agent cooperation in this article, the distribution of strategies is relatively more uniform, with a wider range, and can take timely response measures and adjust decisions in complex attack environments. This indicates that the model is influenced by the attack methods during the game with the attacker, allowing defense strategies and means to be adjusted in a timely manner.

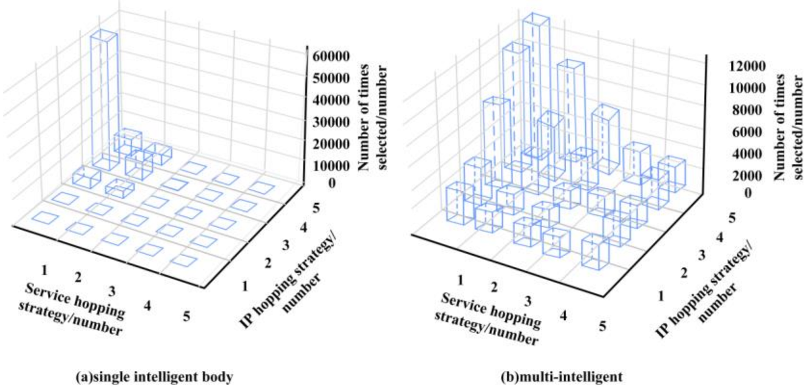

Figure 4: Comparison of reinforcement learning effects under different numbers of intelligent agents.

As shown in Figures 5 and 6, the importance ratios of availability and other indicators are 1:1, respectively—the distribution results of model defense strategies under different attack intensities. From the results in the figure, it can be seen that when the model's strategy tends towards a defensive state, it will block and cut off the network attack chain in a timely manner by increasing the frequency of IP address hopping. When the strategy of the model tends towards a normal service state, it will reduce or avoid negative impacts on the service by lowering the frequency of IP address hopping.

After the virtual network attack experiment mentioned above, the model in this article found that there are three attack paths with serious harm, as shown in Figure 7. Under such attacks, there is a high possibility that internal hosts may not be able to defend against network attacks fully, and further comprehensive security checks are needed. This also indicates that the model in this article can discover security vulnerabilities and weaknesses in the internal network through the self-immune dynamic attack generation module and the attack defense game model, helping the internal network to improve its security defense as soon as possible.

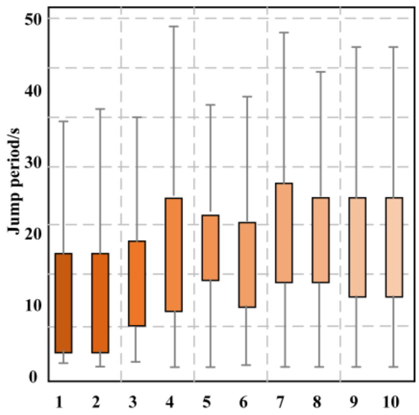

Figure 5: The distribution of Q-learning guided defense strategies under different attack intensities in multi-agent joint states.

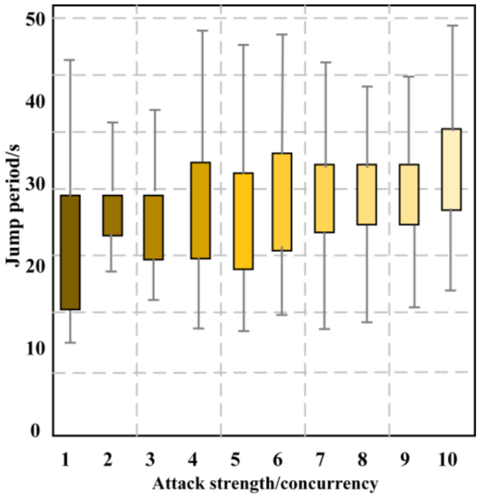

Figure 6: Minmax-Q learning guides the distribution of defense strategies under different attack intensities.

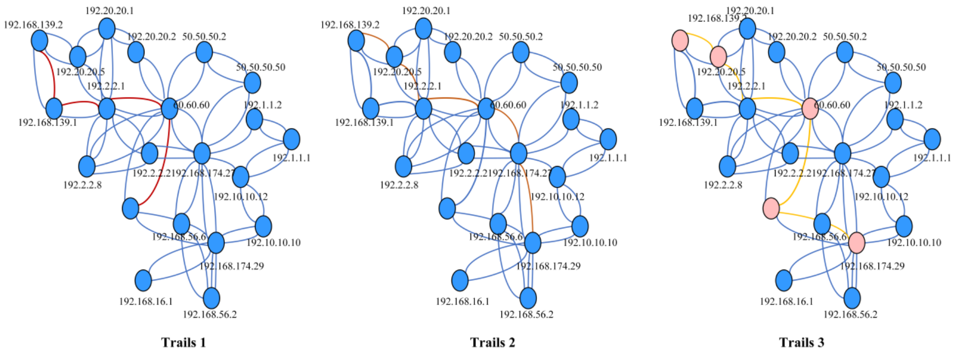

Figure 7: Virtual network attack result chart.

5 CONCLUSIONS

Network security defense is one of the most important contents in the development of cyberspace in the Internet era. In the past, network security defense mainly used static defense methods to resist known network attacks. However, with the improvement of network attack methods and technologies, traditional network security defense methods can no longer adapt to new types of network attacks. Therefore, this article combines reinforcement learning algorithms to construct a network security defense model and implements dynamic attack generation based on the concept of self-immunity. It can discover defense weaknesses in the internal network by simulating network attacks. In addition, this article also combines the attack defense game model to achieve strategy optimization of the model and improves the applicability and adaptability of network defense strategies through virtual network attack experiments. The experimental results show that the model proposed in this paper can quickly determine the optimal attack path in virtual network attacks, and has good stability performance, which can adapt to large-scale network attack environments. The multi-agent joint strategy and Minmax-Q value strategy adopted by the model can adjust defense strategies in a timely manner under different attack intensities, avoiding resource waste and effectively achieving the goal of defending against network attacks. At the same time, it can also adjust defense strategies in a timely manner in daily services, ensuring network security without having a negative impact on services. In addition, through virtual network attack experiments, this model can effectively identify security defense weaknesses in the intranet, clearly mark attack paths with greater harm, and improve the overall defense capability of the intranet in a targeted manner.

Ying Li, https://orcid.org/0009-0000-4846-7054
Lijuan Yang, https://orcid.org/0000-0003-1288-1805
Haiyan Liu, https://orcid.org/0009-0000-9890-2819

REFERENCES

[1] Bashendy, M.; Tantawy, A.; Erradi, A.: Intrusion response systems for cyber-physical systems: A comprehensive survey, Computers & Security, 124(1), 2023, 102984. https://doi.org/10.1016/j.cose.2022.102984

[2] Chen, T.; Liu, J.; Xiang, Y.; Niu, W.; Tong, E.; Han, Z.: Adversarial attack and defense in reinforcement learning-from AI security view, Cybersecurity, 2(1), 2019, 1-22. https://doi.org/10.1186/s42400-019-0027-x

[3] Dixit, A.; Mani, A.; Bansal, R.: An adaptive mutation strategy for differential evolution algorithm based on particle swarm optimization, Evolutionary Intelligence, 15(1), 2022, 1571-1585. https://doi.org/10.1007/s12065-021-00568-z

[4] Goel, A.; Goel, A.-K.; Kumar, A.: The role of artificial neural network and machine learning in utilizing spatial information, Spatial Information Research, 31(3), 2023, 275-285. https://doi.org/10.1007/s41324-022-00494-x

[5] Hernandez, S.-A.; Sanchez, P.-G.; Toscano, M.-L.-K.; Perez, M.-H.; Olivares, M.-J.; Portillo, P.-J.; García, V.-L.-J.: ReinforSec: an automatic generator of synthetic malware samples and denial-of-service attacks through reinforcement learning, Sensors, 23(3), 2023, 1231. https://doi.org/10.3390/s23031231

[6] Hussien, A.-G.; Amin, M.: A self-adaptive Harris Hawks optimization algorithm with opposition-based learning and chaotic local search strategy for global optimization and feature selection, International Journal of Machine Learning and Cybernetics, 13(2), 2022, 309-336. https://doi.org/10.1007/s13042-021-01326-4

[7] Huynh, T.-N.; Do, D.-T.-T.; Lee, J.: Q-Learning-based parameter control in differential evolution for structural optimization, Applied Soft Computing, 107(1), 2021, 107464. https://doi.org/10.1016/j.asoc.2021.107464

[8] Jarah, B.-A.-F.; Jarrah, M.-A.-A.; Almomani, S.-N.; AlJarrah, E.; Al-Rashdan, M.: The effect of reliable data transfer and efficient computer network features in Jordanian banks accounting information systems performance based on hardware and software, database and number of

hosts, International Journal of Data and Network Science, 7(1), 2022, 357-362. https://doi.org/10.5267/j.ijdns.2022.9.012

[9] Kim, J.; Kim, J.; Kim, H.; Shim, M.; Choi, E.: CNN-based network intrusion detection against denial-of-service attacks, Electronics, 9(6), 2020, 916. https://doi.org/10.3390/electronics9060916

[10] Lv, Z.; Han, Y.; Singh, A.-K.; Manogaran, G.; Lv, H.: Trustworthiness in industrial IoT systems based on artificial intelligence, IEEE Transactions on Industrial Informatics, 17(2), 2020, 1496-1504. https://doi.org/10.1109/TII.2020.2994747

[11] Muradova, A.-A.; Khaytbaev, A.-F.: Analysis of the reliability of the components of a multiservice communication network based on the theory of fuzzy sets, Telkomnika (Telecommunication Computing Electronics and Control), 19(5), 2021, 1715-1723. http://doi.org/10.12928/telkomnika.v19i5.19854

[12] Nguyen, G.; Dlugolinsky, S.; Bobák, M.; Tran, V.; López G, Á.; Heredia, I.; Hluchý, L.: Machine learning and deep learning frameworks and libraries for large-scale data mining: a survey, Artificial Intelligence Review, 52(1), 2019, 77-124. https://doi.org/10.1007/s10462-018-09679-z

[13] Nguyen, T.-T.; Reddi, V.-J.: Deep reinforcement learning for cyber security, IEEE Transactions on Neural Networks and Learning Systems, 34(8), 2021, 3779-3795. https://doi.org/10.1109/TNNLS.2021.3121870

[14] Shah, H.; Kakkad, V.; Patel, R.; Doshi, N.: A survey on game theoretic approaches for privacy preservation in data mining and network security, Procedia Computer Science, 155(1), 2019, 686-691. https://doi.org/10.1016/j.procs.2019.08.098

[15] Tubishat, M.; Idris, N.; Shuib, L.; Abushariah, M.-A.; Mirjalili, S.: Improved Salp Swarm Algorithm based on opposition based learning and novel local search algorithm for feature selection, Expert Systems with Applications, 145(1), 2020, 113122. https://doi.org/10.1016/j.eswa.2019.113122