Context-aware Adaptive Route Mutation Scheme: A Reinforcement Learning Approach

Changqiao Xu, *Senior Member, IEEE,* Tao Zhang, *Student Member, IEEE,* Xiaohui Kuang, Zan Zhou, and Shui Yu, *Senior Member, IEEE*

***Abstract*—Moving Target Defense (MTD) is an emerging proactive defense technology, which can reduce the risk of vulnerabilities exploited by attacker. As a crucial component of MTD, route mutation (RM) faces a few fundamental problems defending against sophisticated Distributed Denial of Service (DDoS) attacks: 1) It's unable to make optimal mutation selection due to insufficient learning in attack behaviors. 2) Because network situation is time-varying, RM also lacks self-adaptation in mutation parameters. In this paper, we propose a context-aware Q-learning algorithm for RM (CQ-RM) that can learn attack strategies to optimize the selection of mutated routes. We firstly integrate four representative attack strategies into a unified mathematical model and formalize multiple network constraints. Then, taking above network constraints into considerations, we model RM process as a Markov decision process (MDP). To look for the optimal policy of MDP, we develop a context estimation mechanism and further propose the CQ-RM scheme, which can adjust learning rate and mutation period adaptively. Correspondingly, the optimal convergence of CQ-RM is proved theoretically. Finally, extensive experimental results highlight the effectiveness of our method compared to representative solutions.**

***Index Terms*—Moving Target Defense, Distributed Denial-of-Service (DDoS) attack, Route Mutation, Reinforcement Learning, Context Awareness.**

I. Introduction

NETWORK security is essential for service availability and Quality of Service (QoS) against increasing common network threats, such as Distributed Denial of Service (DDoS) and eavesdropping. For example, as per the attack report in [1], many Internet services were forced to interrupt under DDoS attacks in 2018, which caused $\$221,836,80$ in downtime damage. Static network defense technologies always need to detect attack behaviors [2] [3] [4] or preserve privacy [5] [6], but these technologies are easy to be disrupted. For instance, current DDoS mitigation techniques [7] [8] try to filter out attack traffic, but attack behaviors [9] are going more stealthy, so that these techniques can not distinguish between benign and malicious traffic. As a result, static network defense technologies are incapable to resist the premeditated cyber attack to a great extent.

To cope with this serious inherent weakness, Moving Target Defense (MTD) technology [10] was proposed as a game-changing innovation, which can modify network properties to reduce the success probability of cyber attacks. Since routing is a crucial asset to protect [11][12][13][14], many route mutation (RM) techniques have been came out to implement MTD in recent years [15]-[21]. RM changes routing periodically to avoid the necessary of passing through some compromised links or nodes. Early researches [15][16][17][18] just consider purely random route mutation, which will avoid attacks to a certain extent, but still limited. In [9], Kang *et al.* show that the skewed distribution of flows result in the emergence of critical links or nodes, which are easy to be compromised by reconnaissance, eavesdropping and DDoS attacks. Based on this conclusion, some researches [19] [20] [21] have focused on recognizing attack strategies to optimize mutation selections, but these approaches cannot response fast to the changes of attack strategies. In fact, RM aims to establish dynamics into static network by reconfiguring the routing to keep unpredictability. The objective of this dynamics is to invalidate attacker's prior knowledge, which will thwart the attack chain. Thus, RM has become an emerging MTD solution to defend against network threats.

Although considerable researches have been devoted to RM [15]-[21], rather less attention has been paid to withstand sophisticated DDoS attack, *e.g.*, attack strategy is dynamic. Frequent changes of attack strategy will cause two big challenges that need to be studied and addressed in depth for RM techniques. Firstly, and most importantly, the blindness of mutated routing selection gives rise to inadequate attack avoidance due to insufficiency self-learning in attack strategies, which adversely impact the benefits from RM. Secondly, when facing dynamic attack strategies, RM can't adjust mutation parameters (*e.g.*, mutation period) adaptively to reduce network overheads because current methods are unable to characterize and analyze the real-time network situation. Consequently, how to mitigate sophisticated DDoS attack to improve the effectiveness and feasibility of RM scheme leaves much to be desired.

Manuscript received October 3, 2020; revised February 18, 2021; accepted March 5, 2021. This work is supported by the National Natural Science Foundation of China (NSFC) under Grant Nos.61871048 and 61872253; by the National Key R&D Program of China (2018YFE0205502); by the BUPT Excellent Ph.D. Students Foundation CX2020123. Yu's work is partially supported by Australia ARC DP200101374. *(Corresponding author: Changqiao Xu.)*

C. Xu, T. Zhang and Z. Zhou are with the State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China (e-mail: cqxu@bupt.edu.cn; zhangtao17@bupt.edu.cn; leonzhou@bupt.edu.cn).

X. Kuang is with State Key Laboratory of Science and Technology on Information System Security. He is also a guest professor with Beijing University of Posts and Telecommunications, Beijing 100876, China (e-mail: xhkuang@bupt.edu.cn).

S. Yu is with the School of Computer Science, University of Technology Sydney, NSW 2007, Australia (e-mail: shui.yu@uts.edu.au).

To address the problems above, we propose a **C**ontext-aware **Q**-learning method for network **R**oute **M**utation (**CQ-RM**), which is a significant extension of our previous work [22]. The objective of CQ-RM is to learn attack strategies iteratively with adjusting learning rate and mutation period adaptively. Distinguished from existing solutions, we apply Q-learning [23] into RM innovatively. Q-learning is a model-free reinforcement learning (RL) algorithm that can evaluate the state-action value effectively. RL has been used in theoretical analysis of attack behaviors [24] [25], enhance security [26] and data privacy [27]. In this paper, we attempt to design an extended Q-learning algorithm for RM scheme to defend against dynamic attack strategies. To the best of our knowledge, this is the first work to employ an extended Q-learning in solving RM issues.

To sum up, the main contributions of this paper are summarized as follows:

1) Considering practical conditions, we model multiple network requirements. Based on Satisfiability Modulo Theory (SMT), above requirements are all transformed into network constraints to guarantee the feasibility of mutated routes. We also integrate four representative attack strategies into a unified mathematical model.
2) Taking above constraints into considerations, we model RM process as a Markov decision process (MDP) with multiple constraints. Thereinto, we define the state as current flows' distribution and the action as selecting a mutated route. Then, how to select the optimal mutated route will be transformed into looking for the optimal policy of MDP. In particular, the malicious behaviors of attacker becomes as a part of the environment.
3) For solving above optimization problem, we develop an intelligent CQ-RM innovatively. A context estimation mechanism is designed to characterize and analyze network situation accurately. Differently form directly applying Q-learning algorithm, CQ-RM is capable of adjusting mutation period and learning rate adaptively to reduce network overheads and accelerate the convergence of learning.
4) To demonstrate the effectiveness, we analyze the complexity and the optimal convergence of CQ-RM theoretically. Then, we carry out extensive simulations and the results show that CQ-RM provides significant improvements in multiple aspects including defense, mutation overhead, network and convergence performance with respect to state-of-the-art approaches.

The rest of this paper is organized as follows: Section II explains the related work. System architecture, mutation space formalization and threat model are introduced in Section III. In Section IV, an optimization problem is stated in detail. Next in Section V, to solve the optimization problem, we propose a CQ-RM scheme, which can adjust mutation period and learning rate adaptively. Furthermore, the theoretical analysis of CQ-RM including complexity and convergence is given in Section VI. Section VII shows extensive experimental evaluation about CQ-RM. Lastly, section VIII concludes this paper.

II. Related Work

RM is one of the most important research directions in MTD technologies. In the early researches, mutated routing selection is purely random. For example, Duan *et al.* [16] firstly present a RM framework called random route mutation (RRM), which adopts SMT to acquire appropriate mutated routes. Based on the previous RRM, the work by Gillani *et al.* [17] formulates both network performance and crossfire attack into SMT constraints. The work by Aseeri *et al.* [18] proposes a bidirectional multipath routing algorithm. By the negotiation between source and destination, routes are mutated randomly during multi-flow transmission. Other researches [28][29][30] employ software defined network (SDN) to improve performance in implementation. However, these approaches have the deterministic routing selection strategy and don't consider the importance of different nodes or links to optimize mutation behaviors. Over the recent years, researches have focused on this problem and tried to propose some solutions. The work by Zhang *et al.* [20] proposes a mutation selection strategy that uses hypothesis test to learn malicious reconnaissance strategies. The work by Duan *et al.* [21] presents a proactive RM technique to invalidate attacker's knowledge about critical links in the network. These approaches can learn the adversary strategy to some extent, but the ability of learning is still limited and cannot response fast to the changes of attack strategies.

Some researches consider different attacks such as crossfire attack [31]. Aydeger *et al.* [31] present an SDN-based MTD for proactive and reactive defense simultaneously against crossfire attack. There also exist other researches, for instance, Rauf *et al.* [32] present a decentralize RM scheme without reconfiguring infrastructure devices and Tan *et al.* [33] propose an area-dividing RRM that can decrease convergence time caused by link changes. All of these approaches don't analyze the network situation to reduce network overhead, which will cause the feasibility of RM to be limited.

On the other hand, many researchers have been inspired to apply RL into solving security problems. For example, Yousefi *et al.* [24] propose an RL-based attack graph analysis for discovering the potential attack path. Trejo *et al.* [25] investigate the stackelberg security game. Then, they formulate an adaptive actor-critic framework, and finally find the Nash equilibrium of game. Xiao *et al.* [26] apply deep Q-network (DQN) into a mobile crowdsensing system to derive the optimal policy against sensing attacks. Xiao *et al.* [27] use RL to provide secure offloading for edge nodes against jamming attacks. In [34], R. Elderman *et al.* use different RL algorithms to examine their effectiveness of learning opponents in an incomplete information markov game.

From all these RL-based approaches, we can know the basic idea of RL is to learn optimal strategy iteratively by maximizing cumulative reward value from the environment. Therefore, RL methods are more focused on the strategy of learning to solve problems. Based on these characteristics, RL is considered as a good way to address the main problems in current RM researches. This is because it can optimize the mutated routing selection by learning attack strategies and be

efficient towards multiple kinds of attacks. Meanwhile, the optimal mutated routing selection can reduce mutation overhead effectively. Nevertheless, to the best of our knowledge, the study of combining RM technology with RL is still in a blank stage. Hence, this work is the first attempt to apply RL into solving RM issues.

III. Preliminary

In this section, we introduce system architecture, and formalize the RM space and threat model.

A. System Architecture

Fig.1 shows the framework of our proposed CQ-RM scheme, which is deployed in the centralized network, e.g., SDN. Centralized controller, where the defender is located, has the ability of computing and completing route generation by sending flow tables to routers, which carry out the next mutated route. The protected substrate network consists of many nodes (switchers or routers), which is connected to end-hosts. Each flow is from an end-host to any another end-host and lasts for a certain period of time. Without loss of generality, we assume that there are multiple flows transmitted concurrently in the network. In practical, centralized controller collects and stores traffic information from routers by remotely installing monitoring softwares, such as Cacti [35]. The network topology won't change violently because changing topology is generally expensive and slow [36]. Substrate network can be abstracted as a directed graph $\mathbb{G} = (\mathbb{V}, \mathbb{E})$, in which $\mathbb{V}$ is node set and $\mathbb{E}$ is edge set. An attacker initially compromises or resides in some of the nodes. Whereafter, it can uses the nodes as step stones and further conducts malicious actions like reconnaissance or DDoS attack.

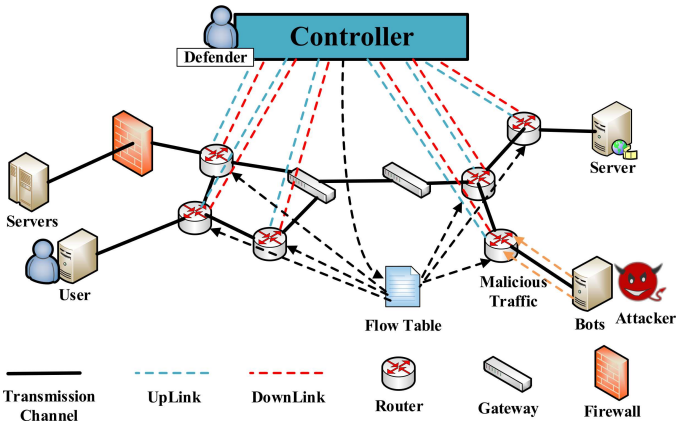

Fig. 1. Framework of CQ-RM scheme.

B. Route Mutation Space Formalization

Considering mutated routes should satisfy multiple network constraints, we formalize these constraints based on SMT [37]. Suppose the network includes n nodes denoted as $v_1, \ldots, v_n$ and m edges denoted as $e_1, \ldots, e_m$. The incoming edge set of node $v_i (1 \leq i \leq n)$ is defined as I_i, the outgoing edge set of node v_i is defined as O_i, and the set of incoming and outing neighbor nodes of v_i is defined as $\mathcal{L}_i$. We define the flow set as $\mathcal{F}$ and use boolean variable b_i^f to denote whether node v_i is selected into flow f ($f \in \mathcal{F}$), and if so, b_i equals 1, otherwise b_i equals 0. Hence, the route formalization of flow f is $B^f = \left\{b_1^f, b_2^f, \cdots, b_n^f\right\}$. Based on the practical conditions, the flow f from source S to destination D should satisfy multiple constraints, and they can be formalized with SMT as following:

1) Reachability Constraints: A valid mutated route should be reachable from source S to destination D, which can be formalized as:

$$\sum_{e_{ji} \in O_j, v_i \in \mathcal{L}_j} b_i^f = \sum_{e_{ij} \in I_j, v_i \in \mathcal{L}_j} b_i^f, \tag{1}$$

$$\sum_{e_{Si} \in O_S, v_i \in \mathcal{L}_S} b_i^f = 1, \sum_{e_{iS} \in I_S, v_i \in \mathcal{L}_S} b_i^f = 0, \tag{2}$$

$$\sum_{e_{iD} \in I_D, v_i \in \mathcal{L}_D} b_i^f = 1, \sum_{e_{Di} \in O_D, v_i \in \mathcal{L}_D} b_i^f = 0, \tag{3}$$

$$b_i^f \in \{0, 1\}, \forall v_i \text{ except } S \text{ and } D, \forall f \in \mathcal{F}. \tag{4}$$

First equation indicates that any node (except source and destination) must have identical number of outgoing and incoming edges in the flow. The second and third equations indicate that source and destination must be S and D. The last equation denotes the value range of variable b_i^f.

2) Middle-ware Device Constraints: An application flow might require some specialized services from the substrate (e.g., the IPSec, etc.). Mostly, only a handful of nodes can provide such requested feature. Suppose these nodes are $b_{i_1}^f, \cdots, b_{i_k}^f$, the SMT formalization is:

$$\left(b_{i_1}^f = 1\right) \vee \cdots \vee \left(b_{i_k}^f = 1\right), v_{i_k} \in \mathbb{V}, \forall f \in \mathcal{F}, \tag{5}$$

which indicates the flow f must pass through one of these middle-ware devices.

3) Capacity Constraints: Mutated routes cannot pass through those switch nodes that have no resources for usage costs. This constraint is formalized as:

$$\sum_{k=0}^{\mathcal{M}} b_i^{f_k} c_i^{f_k} \leq \mathbb{C}_i^{th}, \forall v_i \in \mathbb{V}, f_k \in \mathcal{F}, \tag{6}$$

where $\mathcal{M}$ is the number of flows passing through node v_i, $c_i^{f_k}$ denotes the cost of using the resource at switch node v_i in flow f_k and $\mathbb{C}_i^{th}$ is the maximum threshold of cost at node v_i. The important characteristic on switch nodes is that the costs of resource usage inflate with the increase on the workloads of resources. Thereinto, $c_i^{f_k}$ can be described as:

$$c_i^{f_k} = \mathbb{C}_i' \left(\xi_c^{1 - \frac{\mathbb{C}_i'(k)}{\mathbb{C}_i'}} - 1\right), \forall v_i \in \mathbb{V}, f_k \in \mathcal{F}, \tag{7}$$

where $\mathbb{C}_i'(k)$ is the number of available entries in the routing table at node v_i when the k-th flow arrives. In addition, ξ_c is a parameter that is usually set to $2n$, n is the total number of nodes [38].

4) Quality of Service Constraints: Mutated routes should satisfy some required QoS, such as transmission delays or max number of hops. The delay SMT formalization is described as:

$$\sum_{v_i \in \mathbb{V}} b_i^f \mathcal{D}_f + \sum_{v_i \in \mathbb{V}} \sum_{v_j \in \mathcal{L}_i} b_i^f b_j^f \mathcal{D}_t \leq \varphi_{del}^{th}, \forall f \in \mathcal{F}, \quad (8)$$

where $\mathcal{D}_f$ is node forward delay and $\mathcal{D}_t$ is link transmission delay. In addition, φ_{del}^{th} is the threshold of delay. The hop SMT formalization is described as:

$$\sum_{v_i \in \mathbb{V}} b_i^f \leq \varphi_{hop}^{th}, \forall f \in \mathcal{F}. \quad (9)$$

It means total hop is smaller than threshold φ_{hop}^{th}.

C. Threat Model

For an attacker, it can discover the network topology by reconnaissance in advance. High-volume malicious traffic is usually easy to be filtered while low-volume malicious traffic is adopted widely to cause packets dropping [39]. Therefore, selecting multiple nodes to send low-volume traffic is more likely to obstruct a flow persistently. If nodes in the route are attacked successfully, both attacker and defender know results. In addition, the attacker's resources are limited, it's impossible to compromise all nodes simultaneously. The maximum ability of attacker is supposed to attack ρ nodes. In order to maximize the effectiveness, the attacker needs to adopt appropriate strategies. As shown in Fig.2, we investigate four widely used attack strategies.

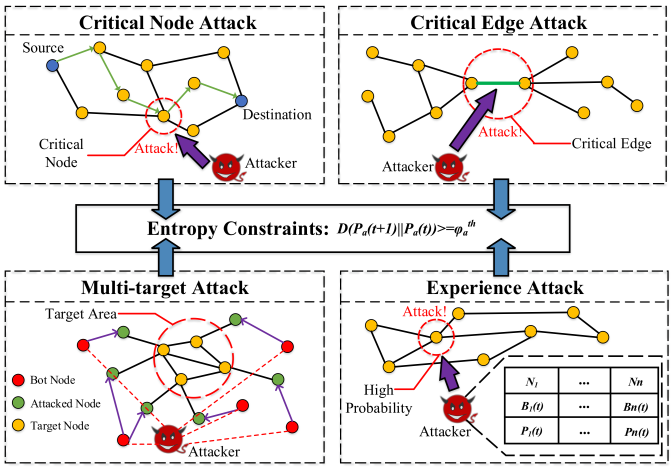

Fig. 2. Four attack strategies and mixed strategy with entropy constraints

(1)**Critical Node Attack**: An attacker is more likely to select nodes with high degree to launch DDoS attacks [40]. Let $D(i)(1 \leq i \leq n)$ be the degree of node v_i. Let $\mathbb{P}_{node}(i)$ be the probability of selecting node v_i to attack, and it can be written as:

$$\mathbb{P}_{node}(i) = \frac{\lambda_{node}(i) D(i)}{\sum_{i=1}^{n} \lambda_{node}(i) D(i)}, \quad (10)$$

where $\lambda_{node}(i)$ is the coefficient of node v_i. The objective of coefficient λ_{node} is to amplify the selection probability of high-degree nodes to avoid randomization. Sorting all nodes in decreasing order of degree, if node v_i is one of ρ highest degree nodes, $\lambda_{node}(i)$ will be a large constant $(\lambda_{node}(i) > 1)$. Otherwise, $\lambda_{node}(i)$ will be 1.

(2)**Critical Edge Attack**: Let $B(i,j)$ be the weight of edge connecting nodes v_i and v_j, where $B(i,j) = D(i) \times D(j)$. We describe the probability of selecting edge e_{ij} to attack as follows:

$$\mathbb{P}_{edge}(i,j) = \frac{\lambda_{edge}(i,j) D(i) D(j)}{\sum_{i=1}^{n} \sum_{j \in \mathcal{L}_i} \lambda_{edge}(i,j) D(i) D(j)}, \quad (11)$$

where $\lambda_{edge}(i,j)$ is the coefficient of edge e_{ij}. The objective of coefficient λ_{edge} is also to amplify the selection probability of high-degree edges to avoid randomization. Sorting all edges in decreasing order of weight, if edge e_{ij} is one of 0.5ρ highest weight edges, $\lambda_{edge}(i,j)$ will be a large constant $(\lambda_{edge}(i,j) > 1)$. Otherwise, $\lambda_{edge}(i,j)$ will be 1.

(3)**Multi-target Attack**: This strategy is abstracted from crossfire attack [9]. Let χ be the target node set and $|\chi|$ be the cardinality of the set. The attacker firstly selects a random node $v_i (1 \leq i \leq n)$ to make $\chi = \{v_i\}$. If $|\mathcal{L}_\chi| < \rho$, all neighbor nodes will be added into χ and iterating above process until $|\mathcal{L}_\chi| \approx \rho$. Then, the attacker sends malicious traffic to nodes in the set $\mathcal{L}_\chi$ to throttle shared edges, which will cause the denial to service of nodes in χ indirectly.

(4)**Experience Attack**: An attacker can determine the priority of targets based on its knowledge background. The probability of selecting node v_i to attack is shown as follows:

$$\mathbb{P}_{exp}(i, t_0) = \frac{\sum_{t=1}^{t_0} \mathcal{B}_i(t)}{\sum_{i=1}^{n} \sum_{t=1}^{t_0} \mathcal{B}_i(t)}, \ t_0 \in \mathbb{N}^+, \quad (12)$$

where $\mathcal{B}_i(t)$ denotes whether node v_i is compromised by the attacker at time slot t, and if so, $\mathcal{B}_i(t) = 1$, otherwise 0.

Considering above four strategies, we formulate a unified mathematical model to select attack strategy dynamically. We define the strategy space as $\Sigma_a = \{\sigma_1, \sigma_2, \sigma_3, \sigma_4\}$ and the probability distribution as $P_a = \{p_1, p_2, p_3, p_4\}$, which should satisfy:

$$0 \leq p_x \leq 1, \forall p_x \in P_a \text{ and } \sum_{p_x \in P_a} p_x = 1. \quad (13)$$

Relative entropy describes the similarity of two probability distributions and is defined as:

$$\mathbb{D}(P_a(t+1) \| P_a(t)) = -\sum_{p_x \in P_a} p_x(t+1) \log \frac{p_x(t+1)}{p_x(t)}, \quad (14)$$

where $P_a(t)$ is the probability distribution at time slot t. To ensure the unpredictability of attack strategies, probability distribution also needs to change dynamically, and it can be formalized as:

$$\mathbb{D}(P_a(t+1) \| P_a(t)) \geq \varphi_a^{th}, \quad (15)$$

where φ_a^{th} is the threshold.

IV. Problem Statement

We assume that the time is divided into equal slots, whose length is ΔT. Hence, time can be slotted with time index $t \in \{0, 1, 2, \cdots\}$. In this section, we model the process of RM as a MDP containing the basic elements of state set, action set,

state transition probability, and reward function. Then, how to select the optimal mutated route is transformed into looking for the optimal policy of MDP.

(1)**State Set**: the network state is denoted as a multi-dimensional vector $S(t) = \{s_1, \cdots, s_n\}$, where s_i $(1 \leq i \leq n)$ is the number of flows passed through node v_i at time slot t. Assuming the quantified number of flow distributions is L, the state set can be expressed as $\mathbb{S} = \{S_1, S_2, \cdots, S_L\}$. When L becomes very large, the convergence of RL will be slow. This problem has been solved in our other work [41].

(2)**Action Set**: selecting a mutated route can be considered as an action, which is denoted as a multi-dimensional vector $A^f(t) = \left\{b_1^f, \cdots, b_n^f\right\}$, where $b_i^f = 1$ denotes node v_i is selected into the route of flow f at time slot t, otherwise $b_i^f(t) = 0$. It can be seen that the total number of possible mutated action is 2^n, which will grow exponentially with the number of network nodes. However, most actions in mentioned space are illegal because they don't satisfy network constraints formalized in Section III.B. Supposing the number of feasible mutated routes is H, action set can be described as $\mathbb{A}^f = \left\{A_1^f, A_2^f, \cdots, A_H^f\right\}$.

(3)**State Transition Probability**: the state transit between consecutive time slots is decided by the state transition probabilities. According to the Markov property, the network state $S(t+1)$ depends only on current state $S(t)$ and the selected action $A^f(t)$, thus we denote the state transition probability as $Pr\left(S(t+1)|S(t), A^f(t)\right)$.

(4)**Reward Function**: to evaluate the effectiveness of actions, a reward function needs to be designed. For simplicity and without loss of generality, we define the reward, influenced by attack behaviors, as follows:

$$\mathcal{R}(t) = \begin{cases} -\xi_r \mathcal{N}(t), & \text{if route is attacked successfully} \\ \mathcal{C}, & \text{if route avoids being attacked} \end{cases} \tag{16}$$

where ξ_r is a coefficient, $\mathcal{N}(t)$ is the number of nodes compromised by the attacker at time slot t and $\mathcal{C}$ is a constant. If current route is attacked successfully, $\mathcal{R}(t)$ will be a negative value linearly related to $\mathcal{N}(t)$. On the contrary, if current route avoids attacks, $\mathcal{R}(t)$ will be a positive constant.

Based on the perspective of defender, its objective is to select the optimal mutated route, which can be transformed into solving the problem of maximizing expected cumulative reward obtained from the environment. To this end, we formulate the optimization problem for RM as follows:

$$\mathbf{P1}: \max_{\pi} E_{\pi}\left[\sum_{k=0}^{\infty} \gamma^k \mathcal{R}(t+k) \middle| S(t) \in \mathbb{S}\right] \tag{17}$$

$$s.t. \quad (1)-(9), \ and \ (13), \tag{18}$$

where π is the policy of selecting mutated routes, E is expectation operator and γ is a discount factor between 0 and 1.

Theoretically, above optimization problem can be solved by the traditional dynamic programming method [42]. However, it is hard to mathematically trace the state transition probability of RM process, which must be used for computing in this scenario. Fortunately, RL has shown the capability to learn the optimal policy, enabling it a good fit for MDP. In this problem, the attacker becomes as a part of the environment. Meanwhile, considering that the defender cannot know attacked targets in advance, P1 is considered as an online problem with multiple constraints. Therefore, to solve P1, we develop a context-aware Q-learning algorithm, which can adjust learning rate and mutation period adaptively.

V. Context-aware Reinforcement Learning-based Route Mutation Scheme

Usually, programmable centralized controller has limited computing and communication resources. However, solving mutated routes that satisfy SMT constraints formalized in Section III.B is NP-complete [16], and real-time computing for mutated routes will consume lots of resources. In this paper, considering aforementioned problem, all feasible mutated routes are pre-calculated by Z3 solver [49] in the centralized controller, staged in router configurations in advance and activated on demand. Then, P1 can be transformed into a new problem that selects feasible mutated routes to maximize the objective function. Fig.3 shows the flow chart of CQ-RM scheme. There exist two cycles, which are learning and awareness cycle respectively. At each time slot, the defender selects a feasible mutated action, then decided by attacker's behaviors, reward and state transition will be returned back to defender for iterative learning, which is called the learning cycle. Context estimation mechanism collects information from the defender and environment, then it outputs threat value to help adjust learning rate and mutation period, which is called the awareness cycle.

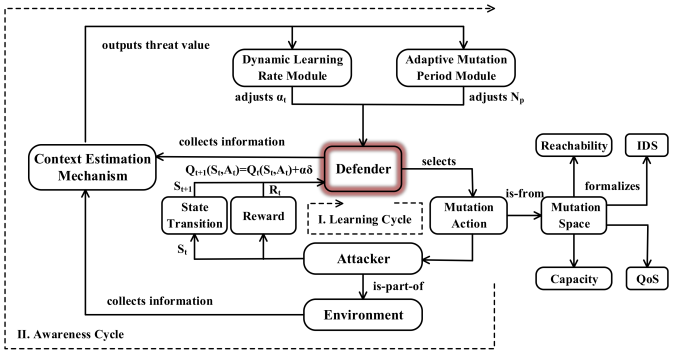

Fig. 3. Flow chart of CQ-RM scheme

In conclusion, CQ-RM makes important contributions to the following two stages:

- Accurately estimates the reliability of context in real time, which is illustrated in Section V.A.
- To solve the optimization problem P1, an extended Q-learning algorithm, which contains dynamic learning rate and adaptive mutation period modules, is proposed in Section V.B.

CQ-RM is also general enough to consider new attack strategies by only updating threat model without changing any other parts.

A. Context Estimation Mechanism

The objective of context estimation mechanism is to analyze the reliability of context, which depends on whether selected routes are compromised by attacker. In this paper, context can be regarded as current network situation. We formulate the context as four-tuple $\left\langle C_t^a, G_t^a, C_t^d, G_t^d \right\rangle$, where C_t^a is attack cost, G_t^a is attack profit, C_t^d is defense cost and G_t^d is defense profit. In DDoS attack architecture, command-and-control server issues commands to zombie machines to send malicious traffic. If attacked nodes are not in the route, malicious traffic sent to these nodes can be regarded as attack cost. If reducing the network system throughput successfully, the attacker will have corresponding attack profit. On the other hand, mutating routes periodically also has defense cost, i.e., loss of system throughput. If routes are protected successfully, the defender will have defense profit. We give following definition:

Definition 1. *We define context matrix Φ as:*

$$\Phi_{T\times I} \triangleq \begin{cases} \phi_{t,i} = -g_{t,i}^a, & \textit{if } v_i \textit{ in the route is attacked}, \\ \phi_{t,i} = g_{t,i}^d, & \textit{if } v_i \textit{ in the route avoids attacks}, \\ \phi_{t,i} = 0, & \textit{otherwize}, \end{cases} \tag{19}$$

where T denotes time slots, I denotes nodes, $g_{t,i}^a \in G_t^a$ denotes the attack profit at node v_i and $g_{t,i}^d \in G_t^d$ denotes the defense profit at node v_i.

Context matrix is the log of attack-defense confrontations. In context matrix, each row represents the confrontation result at one time slot. If node v_i in the route is attacked successfully at time slot t, $g_{t,i}^a$ will equal to the loss of system throughput. On the contrary, if node v_i in the route avoids attacks at time slot t, $g_{t,i}^d$ will equal to the real-time system throughput. Otherwise, if node v_i is not in the route at time slot t, there is no attack profit or defense profit, i.e, $\phi_{t,i}$ will be zero. Based on the context matrix, we calculate the context value, whose definition is shown as follows:

Definition 2. *The context value Ω is described as:*

$$\Omega(T) \triangleq \frac{1}{T}\sum_{t=1}^{T}\left(\sum_{i=1}^{n}\phi_{t,i} + \sum_{i=1}^{n}\widehat{c}_{t,i}^a - C_t^d\right), \tag{20}$$

where $\widehat{c}_{t,i}^a$ is the estimated attack cost at node v_i at time slot t from the perspective of defender.

Existing intrusion detection method [43] can detect the malicious traffic with high accuracy rate. The relationship between estimated attack cost and true attack cost is $\widehat{c}_{t,i}^a = \xi_a c_{t,i}^a$ $(0 < \xi_a < 1, c_{t,i}^a \in C_t^a)$, where ξ_a is the accuracy rate. It's worth to be noted that only estimated attack cost $\widehat{c}_{t,i}^a$ is known for the defender while actual attack cost $c_{t,i}^a$ and accuracy rate ξ_a are both unknown. If node v_i is attacked at time slot t, but is not in the route, $\widehat{c}_{t,i}^a$ will equal to the volume of detected malicious traffic. In addition, the mutation cost C_t^d is formalized as:

$$C_t^d = \xi_d \times \sum_{i=1}^{n} |b_i^f(t) - b_i^f(t-1)|, t \geq 2, \tag{21}$$

where ξ_d is the cost coefficient. The definition of context value is to calculate the sum of all profits and costs, then take the mean value of sum on time slots. To improve the accuracy of context awareness without being influenced by incomplete information, we define threat value as the negative derivative of context value, which is shown as follows:

Definition 3. *The threat value $\mathcal{K}$ at time slot t_0 is defined as:*

$$\mathcal{K}(t_0) \triangleq -\Omega'(t_0) = -\lim_{\Delta t \to 0}\frac{\Omega(t_0 + \Delta t) - \Omega(t_0)}{\Delta t}. \tag{22}$$

Threat value is to utilize the trend of context value to represent the reliability of context. It is easy to understand that when $\mathcal{K} > 0$, the reliability is low because the profit of attacker becomes larger in the attack-defense confrontation. Otherwise, when $\mathcal{K} < 0$, the reliability is high because the profit of defender becomes larger in the attack-defense confrontation.

Algorithm 1 Context Awareness Algorithm

Input: $\{\widehat{c}_{t,1}^a, \cdots, \widehat{c}_{t,n}^a\}$, $\{\phi_{t,1}, \cdots, \phi_{t,n}\}$, C_t^d.
Output: Threat value $\mathcal{K}$.
1: Update the context matrix $\Phi_{T\times I}$ via formula (20).
2: Calculate context value $\Omega(t)$ via formula (21).
3: Calculate threat value $\mathcal{K}$ via formula (22).
4: **return** $\mathcal{K}$.

The pseudo code of context estimation mechanism is shown as **Algorithm 1**. The inputs of context awareness algorithm are estimated attack cost, attack profit, defense profit and defense cost. First of all, as context matrix only records information of previous time slots, we update it to contain the latest information (line 1). Then, we calculate context value and threat value (line 2-3). Finally, The output is the threat value of current time slot (line 4). In general, our context estimation mechanism is an effective method to assess reliability of context and will be applied in next section.

B. Extended Q-learning Algorithm for Optimization Problem

We now use RL to solve the optimization problem of mutated routing. The objective of RL is to learn the optimal strategy for accomplishing goal by maximizing cumulative reward obtained from the environment. Among many of RL algorithms, Q-learning is a model-free algorithm that can evaluate the state-action value (Q-value) effectively. According to Bellman equation, update process after state transition is shown as following:

$$\delta = \mathcal{R}(t) + \gamma \max_{A'} Q_t(S(t+1), A') - Q_t(S(t), A(t)), \tag{23}$$

$$Q_{t+1}(S(t), A(t)) = Q_t(S(t), A(t)) + \alpha\delta, \tag{24}$$

where δ is TD-error and Q_t is Q-value at the time slot t.

In addition, Q-learning also has the dilemma between selecting the action that has highest state-action value (exploitation) and selecting other actions to update Q-values (exploration). There exist three distinguished exploration strategies called ε-greedy, softmax [44] and UCB-1 [45] respectively. In this paper, we adopt ε-greedy that selects random action with probability ε and the optimal Q-value action with probability $1 - \varepsilon$.

Differently from applying Q-learning directly, we propose an extended Q-learning algorithm for RM to adjust the learning rate and mutation period adaptively. Extended Q-learning algorithm mainly have two extra modules, which are introduced next.

(1)Dynamic Learning Rate Module: To accelerate the convergence rate of Q-learning, learning rate is adjusted by context estimation mechanism. Update process after state transition can be rewritten as following:

$$\begin{aligned} Q_{t+1}\left(S(t),A(t)\right) &= Q_t\left(S(t),A(t)\right) + \alpha_t\left(\mathcal{K}\right)\delta \\ &= \left(1-\alpha_t\left(\mathcal{K}\right)\right) Q_t\left(S(t),A(t)\right) \\ &+ \alpha_t\left(\mathcal{K}\right)\left[\mathcal{R}(t) + \gamma \max_{A'} Q_t\left(S(t),A'\right)\right], \end{aligned} \tag{25}$$

where $\alpha_t\left(\mathcal{K}\right)$ is the dynamic learning rate depended on threat value $\mathcal{K}$. Then, we use sigmoid function to define dynamic learning rate function:

$$\alpha_t\left(\mathcal{K}\right) \triangleq 1 \Big/ \left(1+e^{-\mathcal{K}}\right)\tau\ , \tag{26}$$

where τ is the time duration that consists of multiple time slots. We defined it as $\tau = \left\lceil \frac{t}{\xi_\tau} \right\rceil (\xi_\tau \in \mathbb{N}^+)$.

The learning rate α determines the rate at which new information overrides the old one. When most nodes in the route have been attacked, α should be closer to 1. It indicates that the defender will focus more on new information. When most nodes in the route avoid attack, α should be closer to 0. It indicates that the defender will focus more on old information.

(2)Adaptive Mutation Period Module: Mutation period is the key feature of impacting the defense performance. Short mutation period will lead to high network overheads while long mutation period will decrease the effectiveness of RM. Therefore, determining the length of mutation period is a trade-off between defense performance and network overheads. We propose an adaptive mutation period based on the context estimation mechanism. This module can adjust the length of mutation period under different threat values in order to reduce network overheads and keep defense performance simultaneously. The principle of adaptive mutation period module is described as following:

$$N_p = \begin{cases} \tau_m, & \text{when } \mathcal{K} > \varphi_l^{th}, \\ \mathcal{C}_l\tau_m, & \text{when } \mathcal{K} \le \varphi_l^{th} \text{ and } \mathcal{K} > \varphi_h^{th}, \\ \mathcal{C}_h\tau_m, & \text{when } \mathcal{K} \le \varphi_h^{th}, \end{cases} \tag{27}$$

where parameter τ_m is the basic mutation interval. $\mathcal{C}_l$ and $\mathcal{C}_h$ are both constants with $\mathcal{C}_l \le \mathcal{C}_h$. φ_l^{th} and φ_h^{th} are low and high thresholds respectively. $\mathcal{K}$ is the threat value calculated by formula (22). When threat value $\mathcal{K}$ is high, the mutation period becomes short. On the contrary, when threat value $\mathcal{K}$ is low, the mutation period becomes long. Our focus here is to design an adaptive mutation period module, which aims to reduce network overheads. Multiple parameters in this adaption rule can be set according to experiment data in practice.

The pseudo code of extended Q-learning for RM is shown as **Algorithm 2**. Discount factor, greedy factor and output matrix Q are initialized (line 1-2). In addition, mutation period is initialized with one time slot, and the value of mutation flag is same as mutation period at beginning (line 3). Then, the defender explores different network states without knowledge background. The exploration starts from a random network state until after T time slots, which is called an episode (line 4-5). On each time slot, if mutation flag equals 0, the defender will use greedy policy to select an action (line 7-15). Then, the defender executes the selected action and observes the reward $\mathcal{R}(t)$ (line 16-18). Reward is decided by attack behaviors formulated in threat model. The learning rate and mutation period are adjusted by two modules specified as before (line 19-21). Mutation flag is reset to a new mutation period (line 22). Finally, the matrix Q will be updated (line 23). We should notice that the maximization problems in line 14 and 23 are easy to be solved by only querying the Q-table to find the max Q-value.

Algorithm 2 Extended Q-learning Algorithm for Route Mutation

Output: Mutated route selection policy (Matrix Q).

1: Set parameters of discounted factor γ, greedy factor ε.
2: Initialize Q as $0s$.
3: Initialize mutation period $N_p = 1$ and mutation flag $F_p = N_p$.
4: **for** $episode\ k = 1, 2, \cdots, K$ **do**
5: Select a random initial state ($S(t) \in \mathbb{S}$).
6: **for** $time\ slot\ t = 1, 2, \cdots, T$ **do**
7: $F_p = F_p - 1$.
8: **if** $F_p == 0$ **then**
9: Choose a random probability p.
10: Select one mutated route from RM space as,
11: **if** $p \le \varepsilon$ **then**
12: Randomly select an action $A^f(t)$.
13: **else**
14: $A^f(t) = \arg\max_{A^f} Q(S(t), A')$.
15: **end if**
16: Execute mutated action $A^f(t)$
17: Observe outcome reward $\mathcal{R}(t)$.
18: Obtain next state $S(t+1)$.
19: Update threat value $\mathcal{K}$ via Algorithm 1.
20: Adjust learning rate α_t via (26).
21: Adjust mutation period N_P via (27).
22: Reset $F_p = N_p$.
23: Update Q:

$$\begin{aligned} Q_{t+1}\left(S(t),A^f(t)\right) &= \left(1-\alpha_t\left(\mathcal{K}\right)\right) Q_t\left(S(t),A^f(t)\right) \\ &+ \alpha_t\left(\mathcal{K}\right)\left[\mathcal{R}(t) + \gamma \max_{A'} Q_t\left(S(t+1),A'\right)\right]. \end{aligned}$$

24: **end if**
25: **end for**
26: **end for**
27: **return** Q

VI. Complexity and Convergence Analysis

In this section, we theoretically investigate two properties, which are complexity and convergence respectively.

A. Complexity Analysis

Assuming that $|S|$ is the number of network states, $|A|$ is the number of mutation actions, $|T|$ is the number of time slots, n is the number of nodes and $|K|$ is the number of episodes. The space complexity of Algorithm 1 is $O(n|T|)$ and that of Algorithm 2 is $O\left(|S||A|\right)$. The time complexity of Algorithm 1 is $O(|T|)$. Considering that the Algorithm 2 is a kind of value iteration algorithm, its max time complexity is $O\left(|K||T|(|A|+|T|)\right)$.

B. Convergence of Threat Value

From the global perspective, just for proving the convergence of threat value, we model the interactions between attacker and defender as a finitely repeated game. In the time horizon, we assume the number of attacks is finite but sufficient. Attack strategies and RL exploration strategies are supposed to be public. In addition, interaction history, utility functions and the type of game are also common knowledge.

Definition 4. *(Repeated RM Game): The repeated RM game is defined as a tuple $< \mathcal{N}, \mathcal{A}, \mathcal{D}, \Sigma, U >$, where $\mathcal{N} = \{\mathcal{N}_a, \mathcal{N}_d\}$ represents attacker and defender respectively, $\mathcal{A}$ represents attacker's action set, $\mathcal{D}$ represents defender's action set, $\Sigma = \{\Sigma_a, \Sigma_d\}$ represents the mixed-strategy set of attacker and defender, $U = \{U_a, U_d\}$ represents the utility function of attacker and defender, which can be described as:*

$$U_a = \frac{1}{T}\sum_{t=0}^{T} u_a^t\left(\sigma_a\right) = \frac{1}{T}\sum_{t=0}^{T}\left(\sum_{i=1}^{n}\left(g_{t,i}^a - c_{t,i}^a\right)\right), \sigma_a \in \Sigma_a, \tag{28}$$

$$U_d = \frac{1}{T}\sum_{t=0}^{T} u_d^t\left(\sigma_d\right) = \frac{1}{T}\sum_{t=0}^{T}\left(\sum_{i=1}^{n}\left(r_{t,i}^d\right) - C_t^d\right), \sigma_d \in \Sigma_d, \tag{29}$$

where T is the number of stages and $\{u_k^t\left(\sigma_k\right)\}$ is the one-shot utility of player k with mixed-strategy σ_k at time slot t.

Because the game is strictly competitive, it's impossible to cooperate between attacker and defender. As per [46], the existence of fixed point satisfies the Kakutani's theorem, there exists a mixed strategy equilibrium σ^* in the one-shot game. Therefore, there also exists at least one subgame perfect Nash equilibrium in the repeated game [47]. Let t^* be the equilibrium beginning time. When arriving the equilibrium σ^*, one-shot utility values are constants that defined as $u_a^{t^*}\left(\sigma^*\right)$ and $u_d^{t^*}\left(\sigma^*\right)$. Then, we can have:

$$\begin{aligned} u_d^{t^*}\left(\sigma^*\right) - u_a^{t^*}\left(\sigma^*\right) &= \sum_{i=1}^{n}\left(g_{t^*,i}^d - g_{t^*,i}^a + c_{t^*,i}^a\right) - C_{t^*}^d \\ &= \sum_{i=1}^{n}\phi_{t^*,i} + \sum_{i=1}^{n} c_{t^*,i}^a - C_{t^*}^d. \end{aligned} \tag{30}$$

Based on the relationship between estimated attack cost and true attack cost, we know $\sum_{i=1}^{n}\widehat{c}_{t^*,i}^a$ is also constant. We define equilibrium context value Θ_0 as:

$$\Theta_0 = \sum_{i=1}^{n}\phi_{t^*,i} + \sum_{i=1}^{n}\widehat{c}_{t^*,i}^a - C_{t^*}^d, \tag{31}$$

where Θ_0 is still constant. Then, by the Definition 3, we have:

$$\mathcal{K}(t^*) = -\lim_{\Delta t\to 0}\frac{\Delta t}{t^*}\frac{\Theta_0}{\Delta t} = -\frac{\Theta_0}{t^*}. \tag{32}$$

Finally, $\mathcal{K}$ will converge to $-\Theta_0/t^*$ from time slot t^*.

C. Optimal Convergence of CQ-RM

Watkins and Dayan [23] have proved that Q-learning algorithm must converge to the optimal state-action value Q^*. To show the optimal convergence of CQ-RM, we adopt the convergence theorem of stochastic sequence from [48].

Lemma 1. *The random process P_t taking values in $\mathbb{R}^n$ and defined as:*

$$P_{t+1}(S) = \left(1-\alpha_t\right)P_t(S) + \alpha_t G_t(S), \tag{33}$$

converges to zero with probability 1 under the following assumptions:

1) *The state and action spaces are finite,*
2) $0 \le \alpha_t \le 1, \sum_{t=0}^{\infty}\alpha_t = \infty, \sum_{t=0}^{\infty}\alpha_t^2 < \infty$,
3) $\mathrm{E}\left[G_t(S)\,|F_t\right] \le \gamma||P_t||_\infty$, *where* $\gamma \in (0,1)$,
4) $\mathrm{var}\left[G_t(S)|F_t\right] \le C\left(1+||P_t||_\infty^2\right)$, *where* $C > 0$.

Here, F_t stands for the history at time slot t. The notation $\|\cdot\|_\infty$ refers to sup-norm. We define optimal Q-value as $Q^*\left(S(t), A(t)\right)$ and subtract it from both sides of formula (27). Then, $P_t(S,A) = Q_t(S,A) - Q^*(S,A)$, $G_t(S,A) = \mathcal{R}_t + \gamma\max_{A'} Q_t\left(S(t+1), A'\right) - Q^*\left(S(t), A(t)\right)$. We have the same formulation as (33).

In CQ-RM, network states and actions are both finite. When repeated game reaches the equilibrium, threat value will converge to a constant described as $\mathcal{K}^*$. The value of sigmoid function $1/\left(1+e^{-K^*}\right)$ is also a constant described as Ψ. We define $\tau^* = \left\lceil\frac{t^*}{\xi_\tau}\right\rceil$ to be the equilibrium beginning time duration. Now we have:

$$\begin{aligned} \sum_{t=0}^{\infty}\alpha_t(\mathcal{K}) = &\frac{1}{\left(1+e^{-\mathcal{K}_1}\right)} + \frac{1}{2\left(1+e^{-\mathcal{K}_2}\right)} + \cdots + \\ &\frac{1}{\tau^*\left(1+e^{-\mathcal{K}^*}\right)} + \frac{1}{\left(\tau^*+1\right)\left(1+e^{-\mathcal{K}^*}\right)} + \cdots. \end{aligned} \tag{34}$$

The sum of series items before τ^* is a constant defined as Θ_1. We rewrite (34) as:

$$\sum_{t=0}^{\infty}\alpha_t(\mathcal{K}) = \Theta_1 - \Theta_2 + \Psi\sum_{\tau=1}^{\infty}\frac{1}{\tau}, \tag{35}$$

where $\Theta_2 = \left(1+\frac{1}{2}+\cdots+\frac{1}{\tau^*-1}\right)\Big/\left(1+e^{-\mathcal{K}^*}\right)$ and $\Psi = 1/\left(1+e^{-K^*}\right)$. Since Θ_1, Θ_2 and Ψ are all constants, we now just need to prove the divergency of harmonic series $\sum_{\tau=1}^{\infty}1/\tau$. We assume that the harmonic series converges, so $\lim_{n\to\infty}(S_{2n} - S_n) = \lim_{n\to\infty}(\sum_{\tau=1}^{2n}1/\tau - \sum_{\tau=1}^{n}1/\tau) = 0$. On the contrary, we also know $S_{2n} - S_n = 1/(\tau+1) + \cdots + 1/2\tau > 1/2$. It is obvious that above two formulas contradict each other. Therefore, the assumption is not true, *i.e.*, $\sum_{t=0}^{\infty}\alpha_t(\mathcal{K}) = \Theta_1 - \Theta_2 + \Psi\sum_{\tau=1}^{\infty}1/\tau = \infty$.

Similar to above proof process, we have:

$$\sum_{t=0}^{\infty} \alpha_t^2(\mathcal{K}) = \Theta_3 - \Theta_4 + \Psi^2 \sum_{\tau=1}^{\infty} \frac{1}{\tau^2}, \tag{36}$$

where Θ_3 is the sum of series items before τ^* and $\Theta_4 = \left(1 + \frac{1}{4} + \cdots + \frac{1}{(\tau^*-1)^2}\right) \Big/ \left(1 + e^{-\mathcal{K}^*}\right)^2$. In fact, $\sum_{t=0}^{\infty} {\alpha_t}^2 = \sum_{\tau=1}^{\infty} 1/(\tau^2) = \pi^2/6$. Then, $\sum_{t=0}^{\infty} \alpha_t^2(\mathcal{K}) = \Theta_3 - \Theta_4 + \Psi^2\pi/6$, i.e., $\sum_{t=0}^{\infty} \alpha_t^2 < \infty$. Therefore, the assumption (2) is satisfied.

Next, we prove that the random process satisfies assumptions (3) and (4) curtly:

$$\begin{aligned} &E\left[G_t(S,A)|F_t\right] \\ &= \sum_{S' \in \mathbb{S}} P_{S,S'}^a \left[\mathcal{R}(S,A) + \gamma \max_{A' \in \mathbb{A}} Q_t\left(S,A'\right) - Q^*(S,A)\right] \\ &\leq \gamma ||Q_t - Q_t^*||_\infty = \gamma \left\|P_t\right\|_\infty, \end{aligned} \tag{37}$$

where we have the condition that $\sum_{S' \in \mathbb{S}} P_{S,S'}^a = 1$.

$$\begin{aligned} &\mathrm{Var}\left[G_t(S,A)|F_t\right] \\ &= \mathrm{Var}\left[\mathcal{R}(S,A) + \gamma \max_{A' \in \mathbb{A}} Q_t\left(S(t+1),A'\right)|F_t\right] \\ &\leq C\left(1 + ||P_t||_\infty^2\right), \end{aligned} \tag{38}$$

where $\mathcal{R}(S,A)$ is bounded, $0 < \gamma < 1$ and C is a constant. More details on the proof of assumptions (3) and (4) can be seen in [49]. So $P_t = Q_t - Q^*$ will converge to zero, *i.e.*, Q_t will be the optimal Q-value when learning process converges.

VII. Experimental Evaluation

To show the effectiveness of our proposed **Algorithm 1** and **Algorithm 2**, we conduct a series of simulations and compare CQ-RM with latest solutions called I-RRM [21] and Two-way Multi-path [18] respectively. In our experiment, SMT constraints problem is solved by Z3 Solver [49], which is a latest theorem prover from Microsoft Research and can solve tens of thousands of constraints and millions of variables [50]. Meanwhile, we utilize Python to build CQ-RM scheme that combines Z3 solver. Considering that CQ-RM focuses to protect cyber-physical systems in real-world complex network environment, we use BRITE [51] to generate the experimental network topology that satisfies Waxman model [52] with parameter $\alpha = 0.2$ and $\beta = 0.15$. As shown in Fig.4, the number of nodes is set to 100 and red lines are examples of feasible mutated routes from node 0 to node 50. We have used a machine with Core i7-8750H, 2.2 GHz processor and 16GB RAM to run all simulation-based experiments. The main simulation parameters are given in Table I. We analyze the performance of proposed CQ-RM scheme from following five aspects:

A. Defense Performance

Attack success rate is one of the most important parameters to measure defense performance. We carry out 2.3×10^5 time slots of experiments and calculate the attack success rate on each time slot against different attack strategies while I-RRM, Two-way Multipath and CQ-RM are deployed respectively.

TABLE I
Simulation Parameters

Description	Value
Number of network nodes	$N = 100$
Parameters of Waxman model	$\alpha = 0.2, \beta = 0.15$
Discount factor	$\gamma = 0.9$
Number of IPS	$k = 20$
The max capacity for node v_i	$\mathbb{C}_i' = 50$ [38]
Parameter ξ_c	$\xi_c = 200$ [38]
Node forward delay	$\mathcal{D}_f = 0.5ms$ [53]
Link transmission delay	$\mathcal{D}_t = 10ms$ [53]
Link bandwidth	$10Mbps$
Total hop threshold	φ_{hop}^{th}=15
The coefficient of reward	$\xi_r = 1$
The accuracy rate	$\xi_a = 0.8$
The coefficient of time duration	$\xi_3 = 1 \times 10^4$

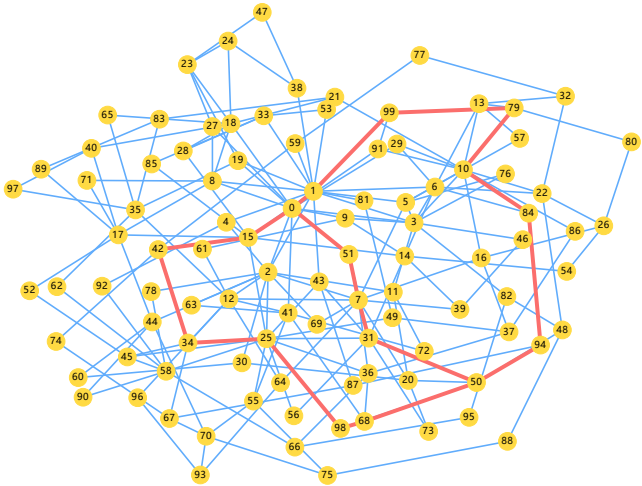

Fig. 4. Network topology with $N = 100$, where red lines are examples of feasible mutated routes from node 0 to node 50.

As shown in Fig.5, attack success rate in I-RRM does not change significantly over time slot. Experience attack has the highest success rate, reaching about 25%. Other attack strategies have the success rates of about 23.5%, 19.5% and 18.5% respectively. This is because experience attack has most knowledge background, which can increase attack success rate significantly. Compared with I-RRM, Two-way Multipath can reduce attack success rate to a certain extent shown in Fig.6. Experience attack also has the highest success rate, reaching about 21%. Other attack strategies have the success rates of about 16.5%, 15.25% and 15% respectively. The results in Fig.7 show that attack success rate in CQ-RM drops significantly with time slot until converges. Results indicate that our proposed algorithm can learn attack strategy and further avoid being attacked. Therein, the success rate of experience attack gradually decreases from 27% to 15%. At the beginning, success rate in CQ-RM is slightly higher than that in I-RRM but eventually decreases by 12%. This is due to random route selection has a little defense capability but is still limited. The success rate of edge attack drops from 23% to 5% with a maximum decrease of 18%. Other attacks decrease from 21% to 7% and from 20% to 10% respectively.

Particularly, the decline of success rate of mixed attack is obviously smaller than that of other attack strategies because mixed attack dynamically changes attack strategies at each

time slot. As shown in Fig.8, we compare attack success rates when three RM schemes deployed respectively. It is obvious that CQ-RM can reduce attack success rate by about 10% compared with I-RRM and about 5% compared with Multi-path. Meanwhile, the difference of attack success rate of mixed attack between three RM schemes is the least. However, CQ-RM can still minimize the success rate of mixed attack. In conclusion, experimental results prove that our method can greatly reduce the success rate of various attack strategies and is better than I-RRM and Two-way Multipath in defense performance.

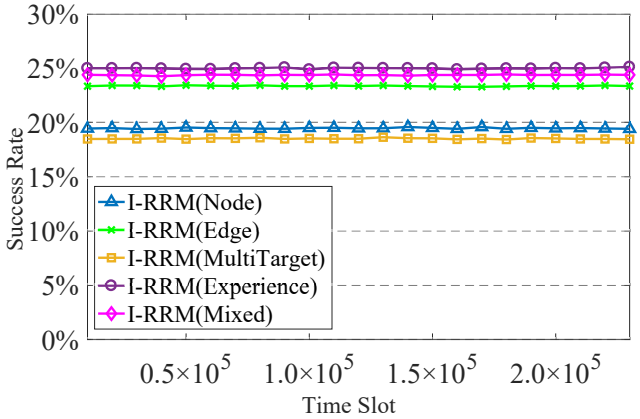

Fig. 5. Defense performance comparison of five attack strategies when I-RRM is deployed.

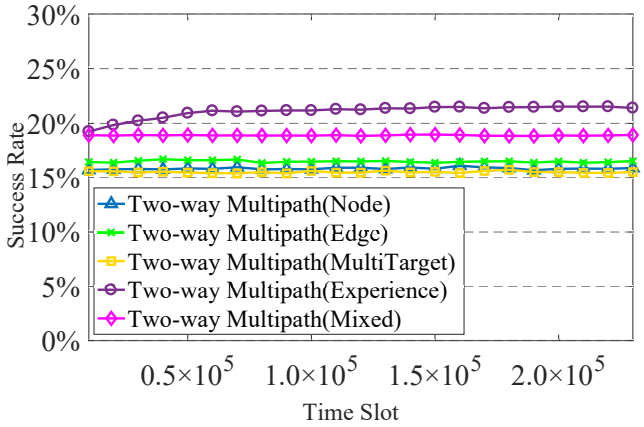

Fig. 6. Defense performance comparison of five attack strategies when Two-way Multipath is deployed.

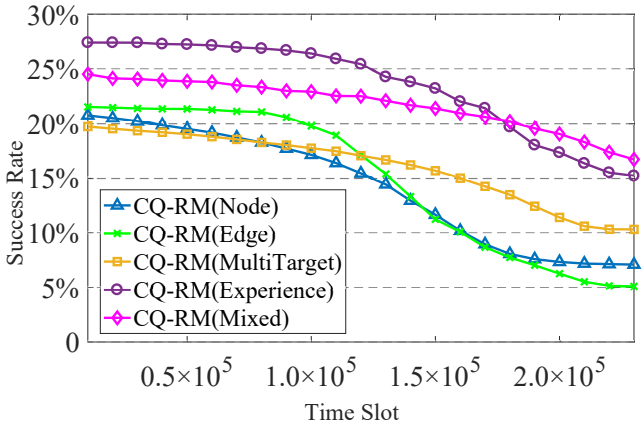

Fig. 7. Defense performance comparison of five attack strategies when CQ-RM is deployed.

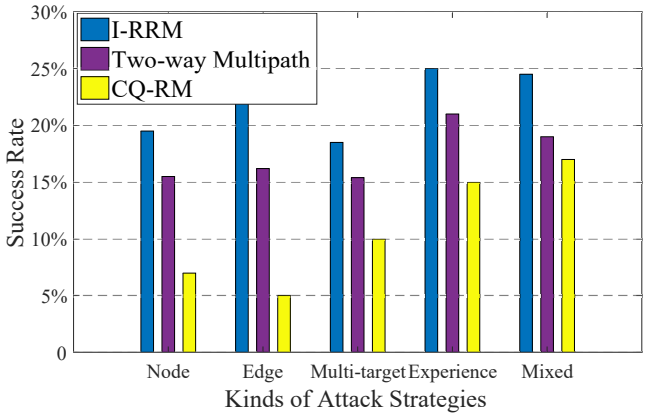

Fig. 8. Success rate comparison of five attack strategies when three RM schemes are deployed respectively.

B. Context-aware Analysis

As specified in Section V.A, context value is the sum of all profits and costs in the attack-defense confrontation process. As shown in Fig.9, red dashed line illustrates that the sum of profits and costs reaches zero. Comparison with red dashed line can reflect who has an advantage in the attack-defense confrontation indirectly.

We calculate the context value with CQ-RM deployed against different attack strategies. Now we analyze the trend of context value to indicate the accuracy of context estimation mechanism. According to results in Fig.9, context values of attack strategies are all drop firstly and rise subsequently. It is caused by the decline of attack success rate so that the defender have more profit in the confrontation gradually. Context value of node attack only drops a little and then increases rapidly because the route avoids attacks largely in CQ-RM. On the other hand, context value under mixed attack declines slowly for a long time because mixed attack is hard to defense. The defender needs to take more time slots to learn how to avoid attacks. Similar results can be concluded in other attack strategies. Based on these analysis, it is reasonable to define the threat value with negative derivative of context value. The threat value represents the accurate awareness of network situation.

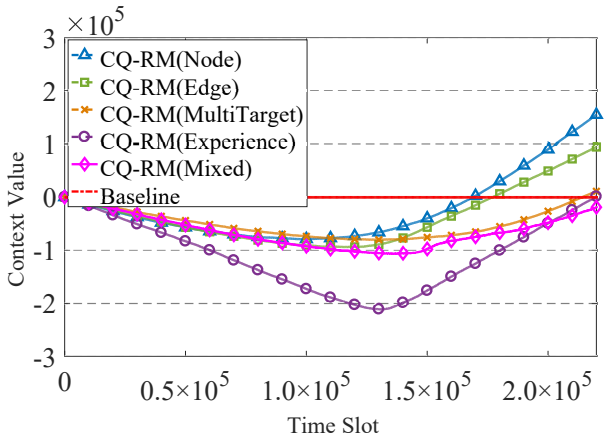

Fig. 9. The context value of five attack strategies, where the red dashed line indicates that the sum of profit and cost is zero.

C. Mutation Overhead Performance

The cost of RM mainly includes network and management overhead. Therefore, the added network performance overhead is a key factor that leads to poor availability for RM scheme. Considering that mutation at each time slot will cause a large number of resource consumption, our proposed adaptive mutation period module can reduce resource consumption in learning process.

As shown in Fig.10, we set $[\mathcal{C}_1, \mathcal{C}_2]$ pairs as $[1, 1]$, $[2, 4]$ and $[3, 5]$ respectively. $[1, 1]$ means that mutation period does not change adaptively in learning process. The results show that adaptive mutation period module will not reduce the defense performance of CQ-RM significantly while it will affect the convergence time of CQ-RM slightly. The reason is context estimation mechanism can guarantee non-mutation happens in the relatively secure environment. Furthermore, we can know that the number of mutation reduce substantially shown in Fig.11, which means the mutation overhead can be reduced to a great extent. The numbers of non-mutation under five attack strategies are $2 \times 10^5, 1.75 \times 10^5, 1.7 \times 10^5, 1.25 \times 10^5$ and 1.5×10^5 respectively. Results show that the decrease of mutation number is most under node attack while the decrease of mutation number is least in the experience attack. It is because the experience attack is depended on hit history so that defender must take mutation actions more times to invalid attacker's knowledge background.

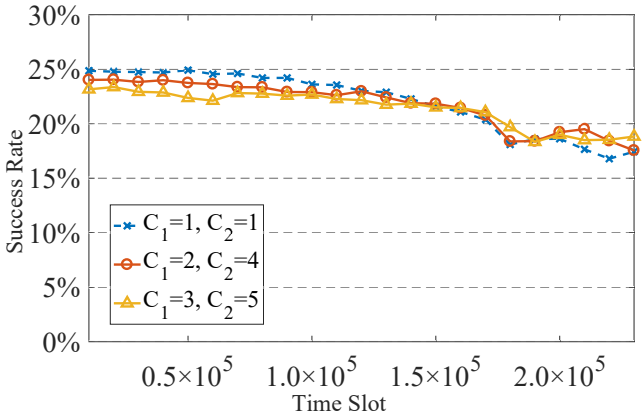

Fig. 10. Defense performance of mixed attack deploying adaptive mutation period module with $[\mathcal{C}_1, \mathcal{C}_2] = [1, 1], [2, 4], [3, 5]$, where blue dashed line is the success rate with consistent mutation period.

D. Network Performance

We consider two metrics called delay and mutation distance to measure the network performance in the RM scheme. Delay is one of the most important metrics for QoS. For simplicity, we assume that hop count is proportional to network latency in the relatively homogeneous network. This assumption has been widely used in existing literature[54]. Mutation distance is also positively correlated with the required additional network overhead caused by RM.

As shown in Fig.14, delays under other attack strategies will eventually decline about 46% except increasing by 30% under the edge attack. The reason is when attacker selects the critical edges to launch DDoS attack, defender should select the route

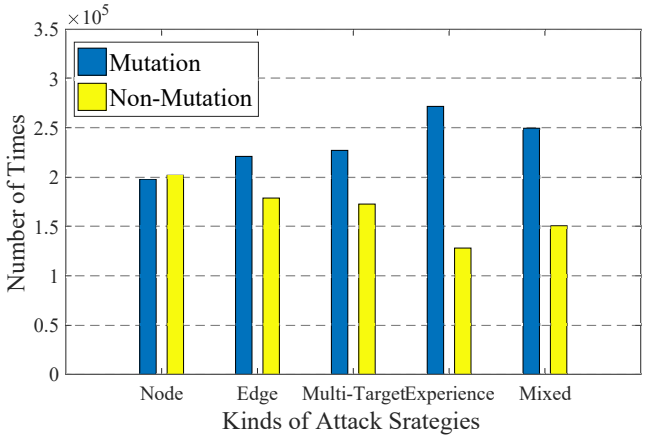

Fig. 11. Deploying the adaptive mutation period module, the number of mutation compares with the number of non-mutation.

that bypasses those critical edges, which will cause more delay. As a whole, CQ-RM will not have a strong impact on delay. The results in Fig.15 show that mutation distance decreases gradually from about 3 at the beginning until converge to about 2. It means that mutation distance in CQ-RM under all attack strategies drops by about 43%. This is because mutation period will become longer with the decrease of attack success rate in RL process. Hence, it can be explained that CQ-RM helps reduce network overhead and increase the feasibility of RM scheme.

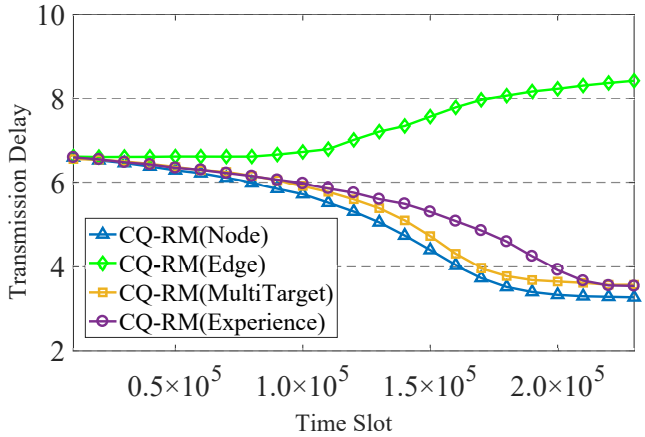

Fig. 14. Transmission delay comparison of four attack strategies when CQ-RM is deployed.

E. Convergence Performance

Fitness [55] is regarded as the cumulative and incremental update value of partial history. Fitness can be described as $F(h_k)$, where $h_k = \{S(0), A(0), S(1), \cdots\}$ is the state-action history generated by the RL process in episode k and F is a kind of evaluation measure for the history. In practice, fitness can be approximated as $F(h_k) \approx \sum_{t=1}^{|h_k|} \delta(t)$, where $|h_k|$ is the number of time slots in history h_k and $\delta(t)$ is calculated by (23). We compare dynamic learning rate with constant learning rate that set as 0.9. As shown in Fig.16, the fitness of dynamic learning rate rises faster than the fitness of constant learning rate at beginning. However, after about 0.75×10^5 time slots, the fitness of constant learning rate exceeds the fitness of dynamic learning rate and the difference

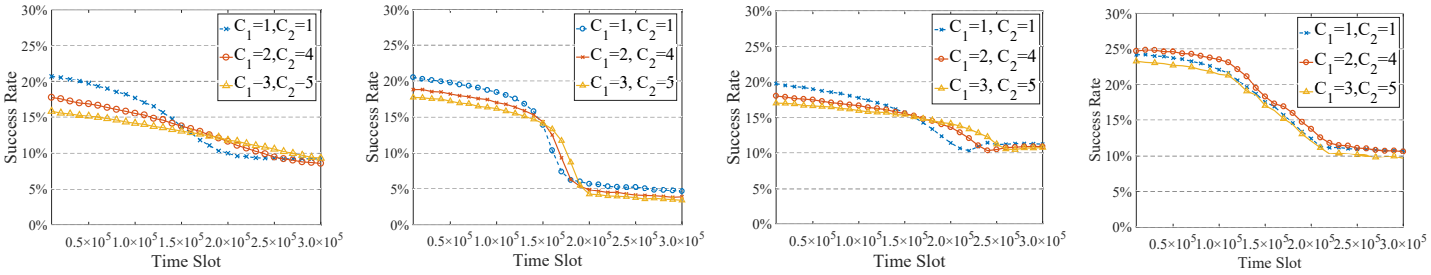

Fig. 12. Defense performance of four attacks deploying the adaptive mutation period module with $[C_1, C_2] = [1, 1], [2, 4], [3, 5]$, where blue dashed line is the success rate with consistent mutation period. (a) Node attack. (b) Edge attack. (c) Multi-Target attack. (d) Experience attack.

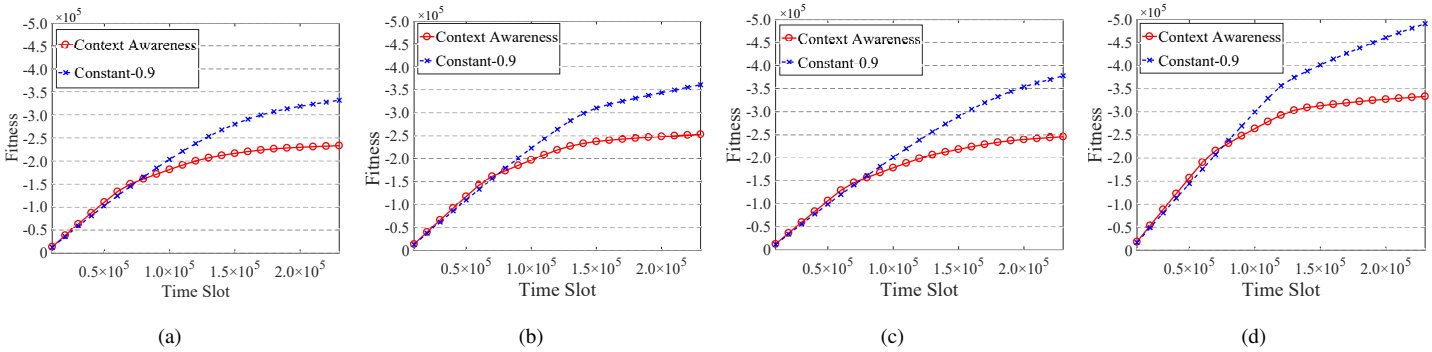

Fig. 13. Convergence performance comparison between dynamic learning rate and constant learning rate. (a) Node attack. (b) Edge attack. (c) Multi-Target attack. (d) Experience attack .

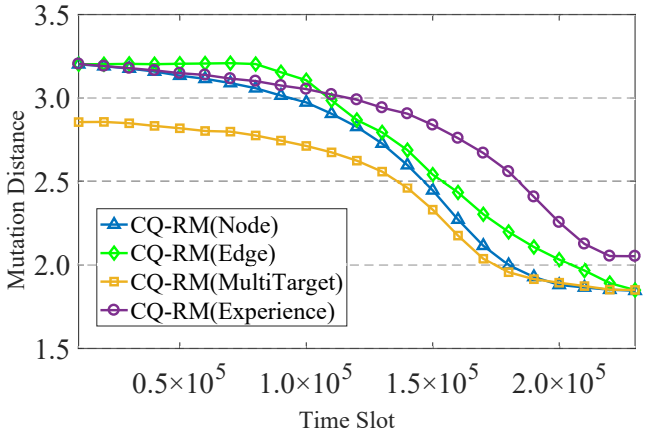

Fig. 15. Mutation distance comparison of four attack strategies when CQ-RM is deployed.

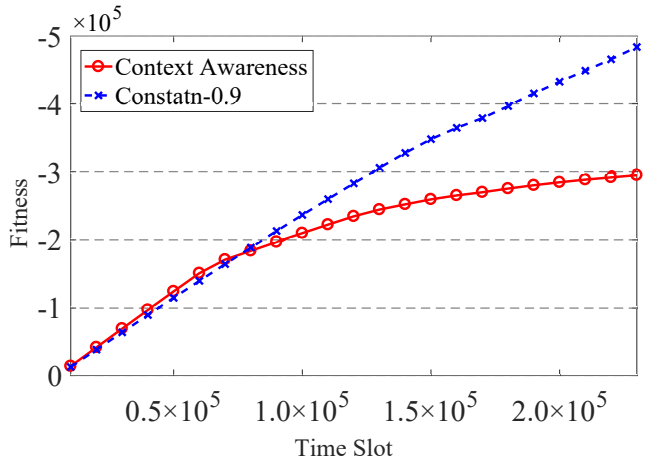

Fig. 16. Convergence performance comparison between dynamic learning rate and constant learning rate.

of fitness becomes larger from that moment. At last, the fitness of dynamic learning rate has converged while the fitness of constant learning rate still increases. When attack success rate converges, dynamic learning rate is close to zero so that the fitness of it converges more quickly. It illustrates that dynamic learning rate can accelerate the convergence of RL. The similar experimental results are also shown in Fig.13(a), Fig.13(b), Fig.13(c) and Fig.13(d).

VIII. Conclusion and Future Work

In this work, we have proposed the CQ-RM scheme to increase the defense performance significantly. Firstly, we formalize network constraints based on SMT and integrate four representative attack strategies into a unified mathematical model. Considering the requirements of adaptiveness, we develop a context estimation mechanism to characterize and analyze the network situation accurately. Based on this mechanism, we design adaptive mutation period and dynamic learning rate module. Then, we further propose a novel extended Q-learning algorithm that can adjust mutation period and learning rate adaptively. Furthermore, the complexity analysis, the convergence of threat value and the optimal convergence of CQ-RM are proved theoretically. Finally, we conduct series of simulation experiments to confirm the feasibility and

effectiveness of our method.

In future work, we will investigate how to deploy our CQ-RM scheme in distributed SDN controller. Besides, we will also discuss the learning of optimal mutation period to improve the effectiveness of CQ-RM particularly.

REFERENCES

[1] Arbor Networks, "Worldwide Infrastructure Security Report: Special Report," pp. 3-4, 2019. [Online]. Available: https://www.netscout.com/report/.

[2] S. Yu, Y. Tian, S. Guo, and D. O. Wu, "Can We Beat DDoS Attacks in Clouds?," *IEEE Transactions on Parallel and Distributed Systems*, 2014, 25(9): 2245-2254.

[3] S. Yu, S. Guo, and I. Stojmenovic, "Fool Me If You Can: Mimicking Attacks and Anti-attacks in Cyberspace," *IEEE Transactions on Computers*, 2015, 64(1): 139-151.

[4] S. Yu, W. Zhou, W. Jia, S. Guo, Y. Xiang, and F. Tang, "Discriminating DDoS Attacks from Flash Crowds Using Flow Correlation Coefficient," *IEEE Transactions on Parallel and Distributed Systems*, 2012, 23(6): 1073-1080.

[5] M. Wang, C. Xu, X. Chen, H. Hao, L. Zhong and S. Yu, "Differential Privacy Oriented Distributed Online Learning for Mobile Social Video Prefetching," *IEEE Transactions on Multimedia*, 2019, 21(3): 636-651.

[6] C. Xu, L. Zhu, Y. Liu, J. Guan and S. Yu, "DP-LTOD: Differential Privacy Latent Trajectory Community Discovering Services over Location-Based Social Networks," *IEEE Transactions on Services Computing*, early access.

[7] N. Ravi and S. M. Shalinie, "Learning-Driven Detection and Mitigation of DDoS Attack in IoT via SDN-Cloud Architecture," *IEEE Internet of Things Journal*, 2020, 7(4): 3559-3570.

[8] L. Zhu, X. Tang, M. Shen, *et al.*, "Privacy-Preserving DDoS Attack Detection Using Cross-Domain Traffic in Software Defined Networks," *IEEE Journal on Selected Areas in Communications*, 2018, 36(3): 628-643.

[9] M. S. Kang, S. B. Lee and V. D. Gligor, "The Crossfire Attack," in *Proc. IEEE Symposium on Security and Privacy*, Berkeley, 2013: 127-141.

[10] J. Cho, *et al.*, "Toward Proactive, Adaptive Defense: A Survey on Moving Target Defense," *IEEE Communications Surveys & Tutorials*, 2020, 22(1): 709-745.

[11] G. Han, J. Jiang, M. Guizani and J. J. P. C. Rodrigues, "Green Routing Protocols for Wireless Multimedia Sensor Networks," *IEEE Wireless Communications*, 2016, 23(6): 140-146.

[12] J. V. V. Sobral, J. J. P. C. Rodrigues, R. A. L. Rablo, K. Saleem and S. A. Kozlov, "Improving the Performance of LOADng Routing Protocol in Mobile IoT Scenarios," *IEEE Access*, 2019, 7: 107032-107046.

[13] J. V. V. Sobral, J. J. P. C. Rodrigues, R. A. L. Rablo, *et al.*, "Routing protocols for low power and lossy networks in internet of things applications," *Sensors*, 2019, 19(9): 2144.

[14] Y. Xu, J. Liu, Y. Shen, J. Liu, X. Jiang and T. Taleb, "Incentive Jamming-Based Secure Routing in Decentralized Internet of Things," *IEEE Internet of Things Journal*, 2021: 8(4): 3000-3013.

[15] J. Liu, H. Zhang, Z. Guo, "A Defense Mechanism of Random Routing Mutation in SDN," *IEICE Transactions on Information and Systems*, 2017, 100(5): 1046-1054.

[16] Q. Duan, E. Al-Shaer and H. Jafarian, "Efficient Random Route Mutation considering flow and network constraints," in *Proc. IEEE Conference on Communications and Network Security (CNS)*, National Harbor, MD, 2013: 260-268.

[17] F. Gillani, E. Al-Shaer, S. Lo, Q. Duan, *et al.*, "Agile virtualized infrastructure to proactively defend against cyber attacks," in *Proc. IEEE International Conference on Computer Communications (INFOCOM)*, Kowloon, 2015: 729-737.

[18] A. Aseeri, N. Netjinda, R. Hewett, "Alleviating Eavesdropping Attacks in Software-Defined Networking Data Plane," in *Proc. ACM Annual Conference on Cyber and Information Security Research (CISRC)*, NY, 2017.

[19] Z. Zhou, C. Xu, X. Kuang, T. Zhang and L. Sun, "An Efficient and Agile Spatio-Temporal Route Mutation Moving Target Defense Mechanism," in *Proc. IEEE International Conference on Communications (ICC)*, Shanghai, 2019: 1-6.

[20] H. Zhang, C. Lei, D. Chang and Y. Yang, "Network moving target defense technique based on collaborative mutation," *Computers & Security*, 2017: 70: 51-71.

[21] Q. Duan, E. Al-Shaer, S. Chatterjee, *et al.*, "Proactive routing mutation against stealthy Distributed Denial of Service attacks: metrics, modeling, and analysis," *The Journal of Defense Modeling and Simulation*, 2018, 15(2): 219-230.

[22] T. Zhang, X. Kuang, Z. Zhou, H. Gao, C. Xu, "An Intelligent Route Mutation Mechanism against Mixed Attack based on Security Awareness," in *Proc. IEEE Global Communications Conference (GLOBECOM)*, Waikoloa, HI, USA, 2019: 1-6.

[23] Watkins C J C H, Dayan P, "Technical Note: Q-Learning," *Machine Learning*, 1992, 8(3-4): 279-292.

[24] M. Yousefi, N. Mtetwa, Y. Zhang and H. Tianfield, "A Reinforcement Learning Approach for Attack Graph Analysis," in *Proc. IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, NY, 2018: 212-217.

[25] K. K. Trejo, J. B. Clempner, A. S. Poznyak, "Adapting attackers and defenders patrolling strategies: A reinforcement learning approach for Stackelberg security games, " *Journal of Computer and System Sciences*, 2018: 35-54.

[26] L. Xiao, Y. Li, G. Han, H. Dai and H. V. Poor, "A Secure Mobile Crowdsensing Game With Deep Reinforcement Learning," *IEEE Transactions on Information Forensics and Security*, 2018, 13(1): 35-47.

[27] L. Xiao, X. Wan, C. Dai, X. Du, X. Chen and M. Guizani, "Security in Mobile Edge Caching with Reinforcement Learning," *IEEE Wireless Communications*, 2018, 25(3): 116-122.

[28] L. Zhang, Q. Wei, K. Gu, *et al.*, "Path hopping based SDN network defense technology," in *Proc. International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery*, 2016: 2058-2063.

[29] S. Yoon, J. Cho, D. S. Kim, *et al.*, "Attack Graph-Based Moving Target Defense in Software-Defined Networks," *IEEE Transactions on Network and Service Management*, 2020, 17(3): 1653-1668.

[30] D. P. Sharma, S. Y. Enoch, J. Cho, *et al.*, "Dynamic Security Metrics for Software-Defined Network-based Moving Target Defense," *Journal of Network and Computer Applications*, 2020, 170: 102805.

[31] A. Aydeger, N. Saputro, K. Akkaya and M. Rahman, "Mitigating Crossfire Attacks Using SDN-Based Moving Target Defense," in *Proc. IEEE Conference on Local Computer Networks (LCN)*, Dubai, 2016: 627-630.

[32] U. Rauf, F. Gillani, E. Al-Shaer, *et al.*, "Formal Approach for Resilient Reachability based on End-System Route Agility, " in *Proc. ACM Workshop on Moving Target Defense (MTD)*, 2016: 117-127.

[33] H. Tan, C. Tang, C. Zhang, *et al.*, "Area-Dividing Route Mutation in Moving Target Defense Based on SDN, " in *Proc. Network and System Security*, 2017: 565-574.

[34] R. Elderman, J. J. Pater L., S. Thie A., M. Drugan M. and M. Wiering M., "Adversarial Reinforcement Learning in a Cyber Security Simulation," in *Proc. International Conference on Agents and Artificial Intelligence*, 2017: 559-566.

[35] T. Urban, Cacti 0.8 beginner's guide, Packt Publishing Ltd, 2011.

[36] C. Liaskos and S. Ioannidis, "Network Topology Effects on the Detectability of Crossfire Attacks," *IEEE Transactions on Information Forensics and Security*, 2018, 13(7): 1682-1695.

[37] M. Franzle, C. Herde, T. Teige, *et al.*, "Efficient solving of large non-linear arithmetic constraint systems with complex boolean structure," *Journal on Satisfiability, Boolean Modeling and Computation*, 2007, 1: 209-236.

[38] M. Huang, W. Liang, Z. Xu, *et al.*, "Dynamic routing for network throughput maximization in software-defined networks," in *Proc. IEEE International Conference on Computer Communications (INFOCOM)*, 2016: 978-986.

[39] M. Shtern, R. Sandel, M. Litoiu, C. Bachalo and V. Theodorou, "Towards Mitigation of Low and Slow Application DDoS Attacks," in *Proc. IEEE International Conference on Cloud Engineering*, Boston, 2014, pp. 604-609.

[40] J. Wu, H. Deng, Y. Tan, *et al.*, "Vulnerability of complex networks under intentional attack with incomplete information," *Journal of Physics A: Mathematical and Theoretical*, 2007, 40(11): 2665.

[41] T. Zhang, B. Zhang, X. Kuang, Y. Wang, S. Yang, C. Xu, "DQ-RM: Deep Reinforcement Learning-based Route Mutation Scheme for Multimedia Services," in *Proc. 2020 International Conference on Wireless Communications and Mobile Computing (IWCMC)*, Cyprus, 2020: 291-296.

[42] D. P. Bertsekas, *Dynamic Programming and Optimal Control*, 3rd ed., vol. I/II. Belmont, MA, USA: Athena Scientific, 2011.

[43] C. Xu, J. Shen and X. Du, "A Method of Few-Shot Network Intrusion Detection Based on Meta-Learning Framework," *IEEE Transactions on Information Forensics and Security*, 2020: 15: 3540-3552.

[44] S M Hung , S N Givigi, "A Q-learning approach to flocking with UAVs in a stochastic environment," *IEEE Transactions on Cybernetics*, 2016, 47(1): 186-197.
[45] A. Garivier, E. Moulines, "On upper-confidence bound policies for switching bandit problems," *Algorithmic Learning Theory*, 2001: 174-188.
[46] I. L. Glicksberg, *et al*., "A Further Generalization of the Kakutani Fixed Point Theorem, with Application to Nash Equilibrium Points," *Proceedings of the American Mathematical Society*, 1952, 3(1): 170-174.
[47] P. Benoit, V. Krishna, "Finitely repeated games," *Econometrica*, 1985.
[48] T. Jaakkola, M. I. Jordan, *et al*., "On the convergence of stochastic iterative dynamic programming algorithms," *Neural Computation*, 1994, 6: 1185-1201.
[49] L. D. Moura, and N. Bjørner, "Z3: An Efficient SMT Solver," *Tools and Algorithms for the Construction and Analysis of Systems*, Berlin, Heidelberg, 2008: 337-340.
[50] L. D. Moura, and N. Bjørner, "Satisfiability Modulo Theories: Introduction and Applications," *Communications of the ACM*, 2011, 54(9): 69-77.
[51] M. Alberto, *et al*., "Brite: A flexible generator of internet topologies," *Technical report*, Boston, MA, USA, 2000.
[52] B M Waxman, "Routing of multipoint connections," *IEEE Journal on Selected Areas in Communications*, 1988, 6(9): 1617-1622.
[53] M. Wang, C. Xu, X. Chen, H. Hao, L. Zhong and D. O. Wu, "Design of Multipath Transmission Control for Information-Centric Internet of Things: A Distributed Stochastic Optimization Framework," *IEEE Internet of Things Journal*, 2019, 6(6): 9475-9488.
[54] B. Peng, A. H. Kemp, and S. Boussakta, "Qos routing with bandwidth and hop-count consideration: A performance perspective," *Journal of Communications*, 2006, 1(2): 1-11.
[55] S. Singh, R. L. Lewis, A. G. Barto and J. Sorg, "Intrinsically Motivated Reinforcement Learning: An Evolutionary Perspective," *IEEE Transactions on Autonomous Mental Development*, 2010, 2(2): 70-82.

Changqiao Xu received the Ph.D. degree from the Institute of Software, Chinese Academy of Sciences (ISCAS) in Jan. 2009. He was an Assistant Research Fellow and R&D Project Manager in ISCAS from 2002 to 2007. He was a researcher at Athlone Institute of Technology and Joint Training PhD at Dublin City University, Ireland during 2007-2009. He joined Beijing University of Posts and Telecommunications (BUPT), Beijing, China, in Dec. 2009. Currently, he is a Professor with the State Key Laboratory of Networking and Switching Technology, and Director of the Network Architecture Research Center at BUPT. His research interests include Future Internet Technology, Network Security and Mobile Communications. He has edited two books and published over 200 technical papers in prestigious international journals and conferences, including IEEE Comm. Magazine, IEEE/ACM ToN, IEEE TMC, INFOCOM etc. He has served a number of international conferences and workshops as a Co-Chair and TPC member. He is currently serving as the Editor-in-Chief of Transactions on Emerging Telecommunications Technologies (Wiley). He is Senior member of IEEE.

Tao Zhang received his B.S. degree in Internet of Things Engineering from Beijing University of Posts and Telecommunications, Beijing, China, and Queen Mary University of London, London, UK in 2018. He is currently working toward a Ph.D in School of Computer Science, Beijing University of Posts and Telecommunications, Beijing, China. His research interests include network security, moving target defense, and reinforcement learning. He is Student member of IEEE.

Xiaohui Kuang received the Ph.D. degree from the School of Computer, National University of defense technology, Changsha, China, in 2003. He is currently a research fellow and professor with National Key Laboratory of Science and Technology on Information System Security, Beijing, China. He is also a guest professor with Beijing University of Posts and Telecommunications. His research interest includes network and information security, wireless network.

Zan Zhou received his B.S. degree in communication engineering from Shanghai University, Shanghai, China, in 2016. He is currently working toward a Ph.D in School of Computer Science, BUPT, Beijing, China. His research interests include network security, moving target defense, and artificial intelligence.

Shui Yu is a Professor of School of Computer Science, University of Technology Sydney, Australia. Dr Yu's research interest includes Big Data, Security and Privacy, Networking, and Mathematical Modelling. He has published three monographs and edited two books, more than 350 technical papers, including top journals and top conferences, such as IEEE TPDS, TC, TIFS, TMC, TKDE, TETC, ToN, and INFOCOM. His h-index is 52. Dr Yu initiated the research field of networking for big data in 2013, and his research outputs have been widely adopted by industrial systems. He is currently serving a number of prestigious editorial boards, including IEEE Communications Surveys and Tutorials (Area Editor), IEEE Communications Magazine, and IEEE Internet of Things Journal. He is a Senior Member of IEEE, a member of AAAS and ACM, and a Distinguished Lecturer of IEEE Communications Society.