Comparative DQN-Improved Algorithms for Stochastic Games-based Automated Edge Intelligence-enabled IoT Malware Spread-Suppression Strategies

Yizhou Shen, Carlton Shepherd, Chuadhry Mujeeb Ahmed, Shui Yu, *Fellow, IEEE*, Tingting Li

***Abstract*—Massive volumes of malware spread incidents continue to occur frequently across the Internet of Things (IoT). Owing to its self-learning and adaptive capability, artificial intelligence (AI) can provide assistance for automatically converging to an optimal strategy. By merging AI into edge computing, we consider an edge intelligence-enabled IoT (EIIoT) environment and provide a stochastic learning strategy for suppressing the spread of IoT malware. In particular, we introduce stochastic game theory to symbolise the whole process of the confrontation between IoT malware and edge nodes. Built upon the theoretical framework to demonstrate the specific spread-suppression architecture, we apply the improved Deep Q-Network algorithms including DDQMS, D2QMS and D3QMS that can deduce the optimal EIIoT malware spread-suppression strategy with better performance. Through experiments, we investigate the influence of related parameters on learning strategy selection, recommending the optimal parameters setting of automated EIIoT malware spread-suppression. We also compare the performance of the proposed three DQN-improved algorithms.**

***Index Terms*—Malware spread-suppression, Edge computing, Artificial intelligence, Internet of Things, Stochastic games, Deep Q-Network**

I. Introduction

VARIOUS suppression techniques have been adopted to address the problem of malware propagation in Internet of Things (IoT) networks. Nevertheless, traditional suppression mechanisms with cumbersomely manual analysis or sophisticated model construction have been unable to catch up with the sharp evolution of IoT malware. Thus, aided by Artificial Intelligence (AI) [1], we aim to automatically suppress the spread of IoT malware, increasing response efficiency and decreasing the false alarm rate, which will thereby achieve autonomous monitoring, autonomous suppression and autonomous counterattack.

Considering the resource-constrained IoT devices, we focus on adopting the structure of edge intelligence that integrates AI into edge computing. Edge intelligence [2], [3] enables the deployment of machine learning algorithms at the edge of IoT networks in response to real-time tasks. In this manner, self-adaptive learning for complex IoT user behavior can be carried out to assist edge node agents in generating a balanced strategy for suppressing the spread of IoT malware.

Compared to the current malware spread solutions including whitelisting [4], patching framework [5] and adversarial defense [6], applying game theory [7]–[9] assists to establish mathematical models with interest conflict between IoT malware and edge nodes in the field of automated malware spread-suppression under edge intelligence-enabled IoT (EIIoT) [10], [11]. Generally, an actual EIIoT network system for malware suppression is dynamic and complicated. For example, when the infected IoT end devices attempt to spread malware, they cannot accurately gain access to the current EIIoT system states and the suppression strategy of edge nodes. Similarly, the edge nodes cannot know the spread target, spread time, and spread route of malware, rendering them unable to ensure the effectiveness of their strategies. Stochastic games, which are a type of multi-agent and multi-state dynamic incomplete games with state transition probability, can better reveal the state transition process between suppression and non-suppression and the interactions between IoT end devices and edge nodes. Thus, we recommend stochastic games to weigh the cost of different strategies and optimize the decisions of a multi-state multi-agent EIIoT malware spread-suppression system with the consideration of limited resources, so as to improve the efficiency of the IoT malware suppression rate.

The Deep Q-Network (DQN) algorithm [12], [13], novelly combining deep learning and reinforcement learning, achieves an end-to-end learning architecture from perception to the real action. It applies neural networks to approximate Q-learning, outperforming human players in Atari 2600 games [14]–[17]. Considering independent and identically distributed data, experience replay is introduced to the DQN algorithm, which stores the data obtained from the exploration environment and updates the deep neural network by random sampling. This store-sample method also breaks the data correlation. Besides, implementing a target network keeps the target Q-value unchanged during specific time steps, improving the algorithm stability. Nevertheless, due to the max operation and bootstrapping in reinforcement learning, the problem of overestimation of DQN algorithm cannot be ignored [18]–[20].

Given the above research, several significant questions come

Manuscript received 09 January 2024. *(Corresponding author: Carlton Shepherd, Tingting Li.)*

Y. Shen, C. Shepherd, and C.M. Ahmed are with the School of Computing, Newcastle University, Newcastle upon Tyne, NE1 7RU, U.K. (e-mails: {Y.Shen34, carlton.shepherd, mujeeb.ahmed}@newcastle.ac.uk).

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

T. Li is with School of Computer Science and Informatics, Cardiff University, Cardiff, CF24 4AG, U.K. (e-mail: lit29@cardiff.ac.uk).

out as follows:

1) Can stochastic games properly express the whole process of IoT malware spread confrontation under EIIoT?
2) Can the traditional DQN algorithm be successfully extended to more advanced algorithms that accelerate convergence to the optimal strategy?
3) Which parameters will effect on EIIoT malware spread-suppression strategy selection?

To address these questions, we propose a stochastic games-based malware spread-suppression (SGMSS) model, for which a stable Nash equilibrium exists, representing an optimal EIIoT malware spread-suppression strategy, through constant simulation and decision-making adaptation. Here, the edge nodes are trained as an intelligent agent to automatically and efficiently analyse IoT malware behaviour and generate learning strategies without manual intervention. In this case, the end users can be protected from malware spread by preventing the access to the core architecture in practice. Furthermore, we extend the DQN algorithm to DDQMS (Double DQN for Malware Spread), D2QMS (Dueling DQN for Malware Spread), and D3QMS (Dueling Double DQN for Malware Spread) algorithms to practically obtain Nash equilibrium, which address the problem of state space explosion and obtain approximate Q-values. Eventually, experimental simulations aim to seek the optimal algorithm and the optimal parameter-setting for EIIoT malware spread-suppression strategy selection.

The main contributions are summarised as follows:

1) We analyze the process of automated EIIoT malware spread-suppression based on stochastic games. Based on this, we construct a theoretical SGMSS model to express the interaction between IoT malware and edge nodes for understanding the internal characteristic of EIIoT malware spread-suppression.
2) We implement DDQMS, D2QMS and D3QMS based on the given EIIoT malware spread-suppression environment for the proposed game model. The value estimation and strategy selection of DDQMS utilise two independently trained neural networks, mitigating the overestimation of DQN in large action space tasks. For the D2QMS, an advantage network is added to express the difference between taking different actions. Moreover, D3QMS combines the D2QMS-based Q-network and DDQMS-based reward function, integrating the advantages of the above two DQN-improved algorithms and forming the third DQN-improved algorithm. These three algorithms can practically solve the optimal EIIoT malware spread-suppression strategy.
3) We compare the influence of related parameters including the learning rate and discount factor on EIIoT malware spread-suppression strategy selection based on DDQMS, D2QMS and D3QMS, as well as compare performances including defender cumulative reward, average episode lengths, defender average episode loss, and successful spread rate. The comparative simulation eventually obtained the optimal parameter setting and the optimal DQN-improved algorithm, providing a practical basis for the optimal strategy selection of EIIoT malware spread-suppression.

The rest of the paper is organised as follows. In Section II, we introduce EIIoT, recap suppression for the spread of IoT malware, and discuss the integrated application of DQN and stochastic games considering the spread of IoT malware. In Section III, we construct an SGMSS model, analysing the stable Nash equilibrium and theoretically providing a unique and optimal EIIoT malware spread-suppression strategy. In Section IV, we develop the DQN-improved algorithms to practically obtain the optimal EIIoT malware spread-suppression strategy for the game. In Section V, we numerically compare the influence of related parameters on the decision-making assisted by DQN-improved algorithms, and the performance among the DQN-improved algorithms, which is followed by a conclusion and the potential future work in Section VI.

II. Related Work

Here, we give an overview of EIIoT, automated suppression for the spread of IoT malware, and DQN-aided and stochastic games-oriented IoT security solutions.

Edge intelligence, deployed at edge nodes, allows rapid access to large amounts of real-time data generated by IoT end devices, which is commonly beneficial for AI model training and reasoning. Nkenyereye et al. [21] suggested a containerized edge intelligence framework for mobile wearable IoT devices to provide intelligent inference services of AI models, in order to achieve dynamic instantiation. Considering the resource limitation of Industrial IoT, Tang et al. [22] designed a multi-exit-based federated edge learning approach, deploying computational intelligence and cooperative training under edge-enabled Industrial IoT, which improves not only data privacy but also bandwidth allocation. Xu et al. [23] presented a smart contract-based edge intelligence architecture to handle the trust and security issues of personalized model learning in IoT networks. They then proved that the established architecture achieves better model accuracy. Xiao et al. [24] explored a high-efficient AI-based task scheduling and offloading scheme for edge-assisted dependent IoT applications under dynamic IoT networks, realizing low latency and reducing energy consumption. Ke et al. [25] merged edge computing into an intelligent parking surveillance system utilizing an enhanced single shot multibox detector taking system flexibility and reliability into account.

As the number of IoT end devices invaded by malware increases, the researchers attempt to achieve automated detection, suppression and control of malware spread utilizing AI technologies including DQN algorithms. Reh et al. [26] described a DDQN-based botnet detector to detect the whole lifecycle of botnets, which can dynamically adapt to the constantly changing IoT environments. It demonstrates that the proposed detector has strong generalizability and resilience to self-respond to malware attack. Shen et al. [27] put forward a differential game-based malware spread-patch framework based on a hybrid patches-distribution approach for Industrial IoT to control IoT malware dissemination. Furthermore, they developed a novel DDQN-based algorithm to seek the optimal

malware control self-learning strategy, which demonstrates the effectiveness and superiority of the proposed method. Zhang et al. [28] recommended a novel hybrid representation learning method to label and cluster android malware via retaining heterogeneous information from various sources, which can effectively recognize and classify security threats through continuous learning.

Intelligent game countermeasure technology is indispensable in IoT systems, which applies DQN variants and game theory to settle sequential decision problems. Benaddi et al. [29] modelled stochastic games to improve the decision-making by MDP and analysing IDS behaviour. They then developed a DQN-based IDS algorithm to obtain the Nash equilibrium and advance the detection rate and accuracy, guaranteeing IoT system security against cyber-attacks. Li et al. [30] advised a non-cooperative game framework to drive the transmission strategy with the consideration of an intelligent reflecting surface (IRS). They next established a DQN-based power allocation algorithm to train the base station to predict attack behaviour and suppress intelligent attackers, so as to enhance the system security. Liu et al. [31] proposed a distributed reflection denial of service (DRDoS) attack-defense framework based on POMDP-aided stochastic games and a recurrent-based DQN algorithm, dynamically converging to the optimal suppression strategy in the context of partial rationality and incomplete information. Due to the characteristic of data storage, the novel DRQN (Deep Recurrent Q-Network) applied in RNN (Recurrent Neural Network) is more suitable for such scenario, attaining the optimal POMDP-assisted attack-suppression strategy. Dunstatter et al. [32] suggested a DNQN (Deep Nash Q-Network) method to derive the optimal attack-suppression strategy based on Markov Games considering the large scale of action and state space. It can monitor and record abnormal behaviour under intrusion prevention and detection systems, effectively fulfilling attack alert and suppression. Zhang et al. [33] constructed an advanced persistent threat rivalry evolutionary game, in which the useful information tends to be left during defenders' strategy-selection and always be exploited by intelligent and rational attackers, causing the information leakage. They further sought out the Nash equilibrium based on two DQN-based learning mechanisms to guarantee that the optimal suppression strategy adjustment timing can be specified by defenders and the least information can be learned by attackers.

As shown in Table I, we provide a comparative table to explain the difference between our work and others in terms of network scenario, game theory, solution, advantage, and limitation to critically analyze the existing state-of-the-art solutions. From the analyses above, it is a novel and feasible idea for exploring the decision-making problem of automated EIIoT malware spread-suppression with the combination of game theory and DQN, which can be a crucial and promising research direction. Compared to the related work above, we concentrate on stochastic games-based and DQN algorithm-aided automated EIIoT malware spread-suppression. Note that the agents in the given environment cannot directly observe the current state, while the state distribution can be derived from the global and partial observations of the model. Besides, there

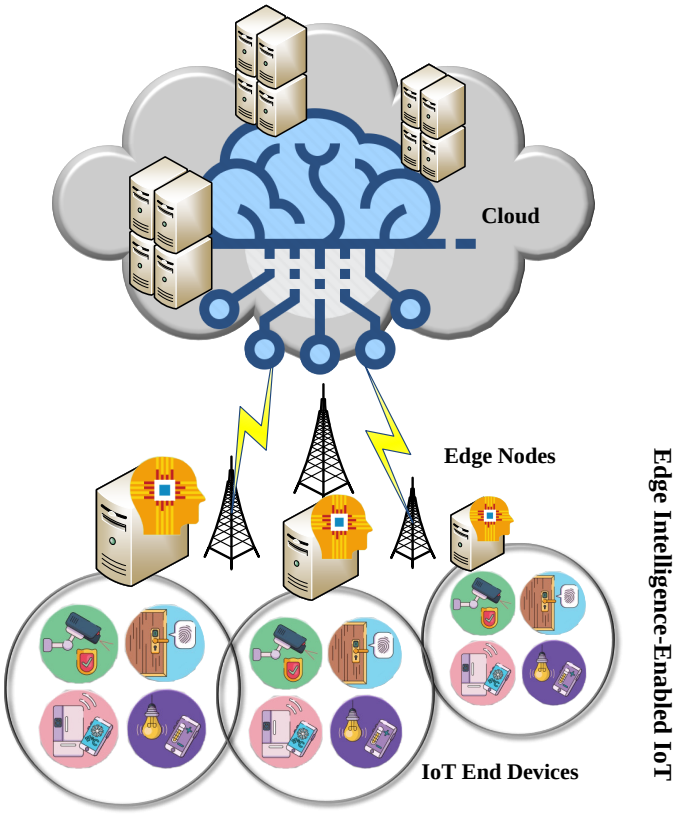

Fig. 1. Framework of edge intelligence-enabled IoT.

are several parameters such as the learning rate and discount factor that can affect the final learning strategy selection. Nevertheless, the existing works do not fully cogitate on these two decision parameters while training the optimal malware spread-suppression strategy. To remedy these deficiencies, we implement DDQMS, D2QMS and D3QMS to solve the optimal strategy of the proposed stochastic games, laying a solid foundation for the practical application of automated EI-IoT malware spread-suppression decisions based on stochastic games.

III. Theoretical Stochastic Games-based EIIoT Malware Spread-Suppression Model

Here, we consider an IoT framework utilizing edge intelligence as shown in Fig. 1 that provides multi-level resource support and performance optimization of IoT end devices based on its operating mechanism and network structure. Edge intelligence relies on the distributed features of edge computing, decentralizing the self-adaptive learning and intelligent decision-making process of AI. It effectively addresses the problem that merely deploying AI in the cloud center probably causes time and resource over-consuming.

We further build a theoretical stochastic games-based malware spread-suppression model in EIIoT, in which the IoT malware follows a random spread policy and the corresponding edge nodes are trained to select an optimal suppression strategy via DQN-improved algorithms including DDQMS, D2QMS and D3QMS. Note that the EIIoT states consistently

TABLE I
COMPARATIVE TABLE EXPLAINING THE DIFFERENCE BETWEEN OUR WORK AND OTHERS

Paper	Network Scenario	Game Theory	Solution	Advantage	Limitation
Reh et al. [26]	Malware botnet detection	None	DDQN-based malware botnet detector	Enhance the IDS generalizability	Lack of researching on the infected system itself
Shen et al. [27]	IIoT malware spread-patch	Differential game	DDQN	Introduce novel spread control parameters to probe the optimal spread-patch strategies under optimization theory	Delay of patches distribution by the central computer
Zhang et al. [28]	Android malware clustering	None	Hybrid representation learning	Maintain heterogeneous information for effective Android malware clustering	Lack of researching on false alarms caused by the incomplete whitelist
Benaddi et al. [29]	Intrusion detection	Stochastic game	DQN-based IDS	Classify attacks to optimize the solution for maximum IDS reward	Lack of extending the model to deal with large-scale cyber attacks
Li et al. [30]	IRS-assised wireless communication	Non-cooperative game	DQN-based power allocation	Research the secrecy rate in IRS-assisted wireless communication networks under active eavesdroppers	Lack of extending the model to deal with large-scale fading channels with path loss
Liu et al. [31]	DRDoS attack-defense	Stochastic game	DRQN	Analyze attack-defense behaviors and address game equilibria under conditions of partial rationality and incomplete information	Lack of diversity in attack and defense
Dunstatter et al. [32]	Cyber alert allocation	Markov Game	DNQN	Extend to a much larger state space and perform loss-less compression of prohibitively large state and action spaces	Approximately obtain the equilibrium point
Zhang et al. [33]	Advanced persistent threats	Evolutionary game	DQN-based learning mechanism	Find out the best timing of strategy adjustment and appropriately allocate the resource	Lack of researching on cooperation among attackers or defenders
Our paper	EIIoT malware spread-suppression	Stochastic game	DDQMS, D2QMS, D3QMS	Propose and compare three DQN-improved algorithms as well as explore crucial parameter-setting	Lack of researching on DRL agents against non-DRL agents on strategy selection

change according to the EIIoT malware spread-suppression strategies.

Definition 1. The theoretical stochastic games-based malware spread-suppression (SGMSS) model for EIIoT is denoted by a six-tuple $SGMSS = \langle X, S, T, \xi(S_m \ |S_n), \gamma, \rho \rangle$. Here,

- $X = \{X_A,\ X_D\}$ represents the participant set, where X_A represents the attacker *IoT malware* and X_D represents the defender *edge nodes*;
- $S = \{S_1,\ S_2, ..., S_N\}$ represents the state space set;
- $T = \{T_A \times T_D\}$ represents the strategy space set, where $T_A = \left\{T_A^1,\ T_A^2,\ ...,\ T_A^K\right\}$ represents the spread strategy set of IoT malware based on the random attack policy, and $T_D = \left\{T_D^1,\ T_D^2,\ ...,\ T_D^L\right\}$ represents the suppression strategy set of edge nodes, trained via DQN-improved algorithms;
- $\xi(S_m|S_n) \mapsto [0,\ 1]$ represents the state transition probability from state S_n to state S_m;
- $\gamma \mapsto [0,\ 1]$ represents the discount factor;
- ρ represents the max value for an spread-suppression attribute, which can be regarded as a constraint.

In the proposed SGMSS, we receive the corresponding spread strategy $T_A^k,\ k \in 1,\ 2,\ ...,\ K$ of the IoT malware, and suppression strategy $T_D^l,\ l \in 1,\ 2,\ ...,\ L$ of the edge nodes, as well as maximize their expect reward functions

$$R_A = \mathbb{E}\left[r \mid T_A^k,\ S_n,\ S_m\right] \tag{1}$$

and

$$R_D = \mathbb{E}\left[r \mid T_D^l,\ S_n,\ S_m\right], \tag{2}$$

respectively. Herein, we have state transition probability

$$\xi(S_m|S_n) = \mathbb{P}\left[S_m \mid S_n,\ (T_A^k,\ T_D^l)\right] \tag{3}$$

moving from the state $S_n,\ n \in 1,\ 2,\ ...,\ N$ to state $S_m,\ m \in 1,\ 2,\ ...,\ M$ with the EIIoT malware spread-suppression strategy $(T_A^k,\ T_D^l)$. This EIIoT malware spread-suppression process will continue to evolve until converging to a stable Nash equilibrium.

Theorem 1: The game SGMSS has the optimal EIIoT malware spread-suppression strategy.

Proof: According to [34], the cost criteria $\mathcal{C}(t,\ s)$ with any EIIoT malware spread-suppression strategy $t \in T$ and initial state distribution $s \in S$ can be denoted as

$$\mathcal{C}(t,\ s) = (1-\gamma)E_t^s \sum_{z=1}^{\infty} \gamma^{z-1} C(X,\ T_A^k,\ T_D^l), \tag{4}$$

in which $k \in \{1,\ 2,\ ...,\ K\}, l \in \{1,\ 2,\ ...,\ L\}, \gamma \in [0,\ 1]$ expresses the reward discount factor, E_t^s expresses the expectation under the EIIoT malware spread-suppression strategy $(T_A^k,\ T_D^l)$, $C(\cdot)$ expresses the cost function under the EIIoT malware spread-suppression strategy $(T_A^k,\ T_D^l)$ with participants IoT malware X_A and edge nodes X_D. Then, we obtain

$$\mathcal{C}(t,\ s) \le \rho. \tag{5}$$

An EIIoT malware spread-suppression strategy $t \in T$ is a Nash equilibrium if any spread strategy t_A of IoT malware satisfies

$$\mathcal{C}_{min}(t,\ s) \le \mathcal{C}_{min}((t_A, t_D^*),\ s), \tag{6}$$

in which $\mathcal{C}_{min}(\cdot)$ expresses the minimal cost function, and t_D^* is the optimal suppression strategy of edge nodes against the spread strategy t_A of IoT malware. Obviously, there is

$$\mathcal{C}((t_A, t_D^*),\ s) < \rho,\ \forall s. \tag{7}$$

Thus, the Strong Slater condition [34] is satisfied and there exists a Nash equilibrium meaning the optimal EIIoT malware spread-suppression strategy. This completes the proof. ■

Theorem 2: The optimal EIIoT malware spread-suppression strategy for the game SGMSS is unique.

Proof: According to value iteration [35], a Bellman optimality backup operator $\mathcal{B}$ [36] is introduced. We obtain

$$\mathcal{BV}(s) := \max_{a \in A} \sum_{\tilde{s} \in S} \xi(\tilde{s}|s,a)[r(s,a,\tilde{s}) + \gamma \mathcal{V}(\tilde{s})], \forall s \in S, \tag{8}$$

where $\mathcal{V}(s)$ expresses the state value function with the current state s, next state $\tilde{s}$, and action a. For $\forall s$, we have

$$|\mathcal{BV}_1(s) - \mathcal{BV}_2(s)| = \|\mathcal{BV}_1(s) - \mathcal{BV}_2(s)\|_\infty, \tag{9}$$

where $\mathcal{V}_1(s)$ and $\mathcal{V}_2(s)$ expresses two different state value functions, and $\|\cdot\|_\infty$ expresses the infinity-norm. Based on the Chebyshev distance [37], we have

$$\|\mathcal{BV}_1(s) - \mathcal{BV}_2(s)\|_\infty = \max_s |\mathcal{BV}_1(s) - \mathcal{BV}_2(s)|. \tag{10}$$

According to Eq. (8), we have

$$|\mathcal{B}\mathcal{V}_1(s)| = \max_{a_1 \in A} \sum_{\tilde{s} \in S} \xi(\tilde{s}|s, a_1)[r(s, a_1, \tilde{s}) + \gamma \mathcal{V}_1(\tilde{s})], \quad (11)$$

and

$$|\mathcal{B}\mathcal{V}_2(s)| = \max_{a_2 \in A} \sum_{\tilde{s} \in S} \xi(\tilde{s}|s, a_2)[r(s, a_2, \tilde{s}) + \gamma \mathcal{V}_2(\tilde{s})], \quad (12)$$

respectively. Then,

$$\begin{aligned}
&\max_s |\mathcal{B}\mathcal{V}_1(s) - \mathcal{B}\mathcal{V}_2(s)| \\
&\leq \gamma \max_s \left\{ \max_{a \in A} \left| \sum_{\tilde{s} \in S} \xi(\tilde{s}|s, a) \left[\mathcal{V}_1(\tilde{s}) - \mathcal{V}_2(\tilde{s}) \right] \right| \right\} \\
&\leq \gamma \max_s \left\{ \max_{a \in A} \left[\sum_{\tilde{s} \in S} \xi(\tilde{s}|s, a) \left[\mathcal{V}_1(\tilde{s}) - \mathcal{V}_2(\tilde{s}) \right] \right] \right\} \\
&\leq \gamma \max_s \left\{ \max_{\tilde{s}, a \in A} |\mathcal{V}_1(\tilde{s}) - \mathcal{V}_2(\tilde{s})| \right\} \\
&\leq \gamma \left\| \mathcal{V}_1(s) - \mathcal{V}_2(s) \right\|_\infty .
\end{aligned} \quad (13)$$

According to Eqs. (10) and (13), we have

$$\left\| \mathcal{B}\mathcal{V}_1(s) - \mathcal{B}\mathcal{V}_2(s) \right\|_\infty \leq \gamma \left\| \mathcal{V}_1(s) - \mathcal{V}_2(s) \right\|_\infty . \quad (14)$$

Hereto, we have proved that $\mathcal{B}$ belongs to a contraction mapping. We then prove the uniqueness by contradiction. Assume $\mathcal{B}$ has two optimal EIIoT malware spread-suppression strategies $t_1 \in T$ and $t_2 \in T$ such that $t_1 \neq t_2$. Then there must be

$$\left\| t_1 - t_2 \right\|_\infty > 0, \quad (15)$$

and

$$\left\| \mathcal{B}t_1 - \mathcal{B}t_2 \right\|_\infty = \left\| t_1 - t_2 \right\|_\infty . \quad (16)$$

Here, the optimal strategies t_1 and t_2 can be derived from the above two state value functions $\mathcal{V}_1(s)$ and $\mathcal{V}_2(s)$. Thus, Eq. (14) can be rewritten as

$$\left\| \mathcal{B}t_1 - \mathcal{B}t_2 \right\|_\infty \leq \gamma \left\| t_1 - t_2 \right\|_\infty . \quad (17)$$

Due to the contraction mapping of $\mathcal{B}$ and according to Eq. (17), we have

$$\left\| \mathcal{B}t_1 - \mathcal{B}t_2 \right\|_\infty \leq \gamma \left\| t_1 - t_2 \right\|_\infty < \left\| t_1 - t_2 \right\|_\infty . \quad (18)$$

Thus, the hypothesis is not valid. The state value function satisfying the Bellman optimal equation is unique and the state value function derived through value iteration must be optimal. As a corollary, the unique optimal EIIoT malware spread-suppression strategy for the game SGMSS can be obtained. This completes the proof. ■

Heretofore, Theorem 1 demonstrates that the Nash equilibrium can be reached in the proposed game SGMSS, meaning that there exists an optimal EIIoT malware spread-suppression strategy. In Theorem 2, a Bellman optimality backup operator is introduced to perform value iteration for verifying the uniqueness of the optimal EIIoT malware spread-suppression strategy by contradiction. Specifically, once the Nash equilibrium is achieved, both IoT end devices and edge nodes hold their own optimal EIIoT malware spread-suppression strategies, which form a fixed strategy set. In this strategy set, none of IoT end devices and edge nodes is willing to change their strategies, otherwise they cannot remain their maximal rewards. Overall, Theorems 1 and 2 indicate that the strategies of the IoT end devices and edge nodes strike a balance.

IV. DQN-Improved Algorithms for Practical Optimal EIIoT Malware Spread-Suppression Strategies

Here, we propose the DQN-improved algorithms consisting of DDQMS, D2QMS and D3QMS. DDQMS aims to decrease the overestimation of Q-value by decomposing action selection and value estimation. D2QMS adds an advantage network, separating the last layer of the neural network into two parts, which is an improvement to the DQN network structure. The novel D3QMS algorithm, combining DDQMS and D2QMS, absorbs the advantages of both algorithms. In this manner, we solve the problem of the difficulty in achieving game parameters and the Nash equilibrium of the given SGMSS in practice.

A traditional DQN algorithm always results in overestimation of Q-values. The optimization goal of the traditional DQN algorithm is

$$y = r + \gamma Q(\tilde{s}, \arg\max_{\tilde{a}} Q(\tilde{s}, \tilde{a}; \omega); \omega), \quad (19)$$

where the selection of actions depends on the target network ω. The optimal action

$$a^* = \arg\max_{\tilde{a}} Q(\tilde{s}, \tilde{a}; \omega) \quad (20)$$

is first selected under the state $\tilde{s}$, and then the corresponding value

$$Q = Q(\tilde{s}, a^*; \omega) \quad (21)$$

is calculated. In this case, the maximum value of all actions estimated by the neural network is obtained each time while calculating via the same Q-network. Considering that the value estimated by the neural network probably produces positive or negative errors at certain times, the positive errors will accumulate under the updating mode of traditional DQN. To be specific, assume Q-values of all actions under the state $\tilde{s}$ equal to 0, i.e.

$$Q(\tilde{s}, a_i) = 0, \ \forall i, \quad (22)$$

the target value should be

$$y = r + 0 = r. \quad (23)$$

Nevertheless, there exists a positive error in the estimation of an action $\acute{a}$ due to the error of neural network, i.e.

$$Q(\tilde{s}, \acute{a}) > 0. \quad (24)$$

Thus, the current target value becomes

$$y = r + \gamma \max Q > r, \quad (25)$$

leading to the overestimation. As a corollary, we improve the traditional DQN to DDQMS, D2QMS and D3QMS algorithms as well as compare the performance as follows.

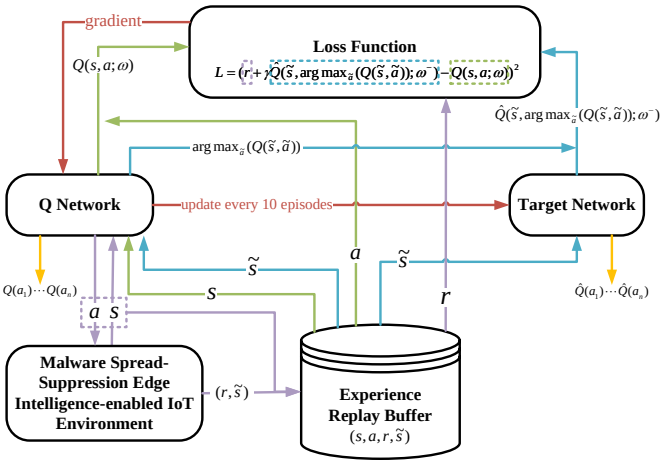

Fig. 2. DDQMS structure.

A. DDQMS

DDQMS integrating double Q-learning and DQN is proposed to investigate EIIoT malware spread-suppression based on the Double DQN algorithm developed by Van Hasselt et al. [38]. The training process of DDQMS is similar to that of tridtional DQN algorithm. Specifically, in the proposed DDQMS, the Q-network ω is utilised to select an action with the maximal value and the target network ω^- is applied to calculate the value of the selected action. Thus, the optimization goal of the DDQMS algorithm is

$$y = r + \gamma Q(\tilde{s}, \arg\max_{\tilde{a}} Q(\tilde{s},\tilde{a};\omega);\omega^-). \tag{26}$$

By virtue of DDQMS, an optimal EIIoT malware spread-suppression strategy can be attained through continuous learning, which separates action selection from strategy evaluation, addressing the problem of overestimating the action value function in the traditional DQN training process. The specific structure of DDQMS for automatically suppressing the spread of IoT malware is shown in Fig. 2. We then practically develop the DDQMS algorithm to obtain the optimal EIIoT malware spread-suppression strategy for the game SGMSS as in Algorithm 1.

B. D2QMS

D2QMS is an advanced to investigate EIIoT malware spread-suppression strategy based on the Dueling DQN algorithm developed by Wang et al. [39], which slightly modifies the network structure to separate values from actions. Specifically, the state value can be predicted separately and is no longer completely dependent on the action value. In this case, our model can attain not only a certain state value but also different action values under that state. Thus, it can learn the state and action independently but closely and process more flexibly in the EIIoT malware spread-suppression environment. Combining the idea of advantage learning, D2QMS splits the abstract characteristics into two branches, symbolized as value function $V(s)$ and advantage function $A(s,a)$. Here, $V(s)$ represents the evaluation of the state s, and $A(s,a)$ represents the advantage of an action a in comparison with the average

Algorithm 1: DDQMS algorithm to obtain the optimal EIIoT malware spread-suppression strategy for the game SGMSS

Initialize the input experience replay buffer Σ, Q-network ω, target network ω^- copying of ω, total episode number τ, and episode number $e = 0$;
while $e \leq \tau$ **do**
  Take an action a and select a state s in the EIIoT malware spread-suppression environment, and obtain $Q(s,a;\omega)$;
  Add transition tuple $(s,a,r,\tilde{s})$ to Σ;
  Sample a random batch from $Unif(\Sigma)$;
  Construct target values;
  $a^{max}(\tilde{s};\omega) \leftarrow \arg\max_{\tilde{a}} Q(\tilde{s},\tilde{a};\omega)$;
  if $\tilde{s}$ *is a terminal* **then**
    $y \leftarrow r$
  end
  $y \leftarrow r + \gamma Q(\tilde{s}, \arg\max_{\tilde{a}} Q(\tilde{s},\tilde{a};\omega);\omega^-)$;
  $L \leftarrow \|y - Q(s,a;\omega)\|^2$;
  Move to the next state $\tilde{s}$ and next action $\tilde{a}$;
  $e \leftarrow e + 1$;
  Decay learning rate per episode;
end
RETURN trained results;

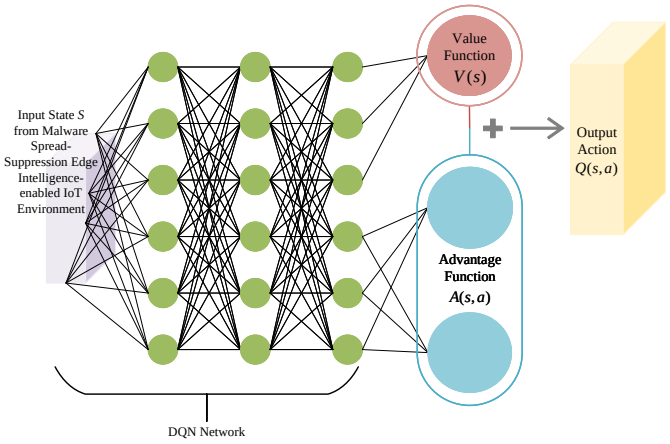

Fig. 3. D2QMS structure.

value of all actions under state s, which can be utilised to evaluate each action in the current state. In the end, two branches are combined via aggregation to obtain the Q-value $Q(s,a)$, represented by

$$Q(s,a) = V(s) + A(s,a). \tag{27}$$

The specific structure of D2QMS for automatically suppressing the spread of IoT malware is shown in Fig. 3. We then practically develop the D2QMS algorithm to obtain the optimal EIIoT malware spread-suppression strategy for the game SGMSS as in Algorithm 2.

Algorithm 2: D2QMS algorithm to obtain the optimal EIIoT malware spread-suppression strategy for the game SGMSS

Initialize the input experience replay buffer Σ, Q-network ω, total episode number τ, episode number $e = 0$, value function $V(s)$, and advantage function $A(s, a)$;
$Q(s, a; \omega) \leftarrow V(s) + A(s, a)$;
while $e \leq \tau$ **do**
  Take an action a and select a state s in the EIIoT malware spread-suppression environment;
  Add transition tuple $(s, a, r, \tilde{s})$ to Σ;
  Sample a random batch from $Unif(\Sigma)$;
  Construct target values;
  $a^{max}(\tilde{s}; \omega) \leftarrow \arg\max_{\tilde{a}} Q(\tilde{s}, \tilde{a}; \omega)$;
  if *$\tilde{s}$ is a terminal* **then**
    $y \leftarrow r$
  end
  $y \leftarrow r + \gamma Q(\tilde{s}, \arg\max_{\tilde{a}} Q(\tilde{s}, \tilde{a}; \omega); \omega)$;
  $L \leftarrow \|y - Q(s, a; \omega)\|^2$;
  Move to the next state $\tilde{s}$ and next action $\tilde{a}$;
  $e \leftarrow e + 1$;
  Decay learning rate per episode;
end
RETURN trained results;

Algorithm 3: D3QMS algorithm to obtain the optimal EIIoT malware spread-suppression strategy for the game SGMSS

Initialize the input experience replay buffer Σ, Q-network ω, total episode number τ, episode number $e = 0$, value function $V(s)$, and advantage function $A(s, a)$;
$Q(s, a; \omega) \leftarrow V(s) + A(s, a)$;
while $e \leq \tau$ **do**
  Take an action a and select a state s in the EIIoT malware spread-suppression environment;
  Add transition tuple $(s, a, r, \tilde{s})$ to Σ;
  Sample a random batch from $Unif(\Sigma)$;
  Construct target values;
  $a^{max}(\tilde{s}; \omega) \leftarrow \arg\max_{\tilde{a}} Q(\tilde{s}, \tilde{a}; \omega)$;
  if *$\tilde{s}$ is a terminal* **then**
    $y \leftarrow r$
  end
  $y \leftarrow r + \gamma Q(\tilde{s}, \arg\max_{\tilde{a}} Q(\tilde{s}, \tilde{a}; \omega); \omega^-)$;
  $L \leftarrow \|y - Q(s, a; \omega)\|^2$;
  Move to the next state $\tilde{s}$ and next action $\tilde{a}$;
  $e \leftarrow e + 1$;
  Decay learning rate per episode;
end
RETURN trained results;

C. D3QMS

From the above analysis, we understand that DDQMS and D2QMS both have better performance in searching for the optimal EIIoT malware spread-suppression strategy, respectively. Herein, we present a novel D3QMS algorithm integrating the strengths of both DDQMS and D2QMS. In D3QMS, the Q-network and target network are both implemented to obtain the Q-values, eliminating the overestimation of Q-values. Besides, an advantage function is introduced, which can learn the difference between different actions, especially in an environment with a large action space. The structure of D3QMS for automatically suppressing the spread of IoT malware is shown in Fig. 4. We then practically develop the D3QMS algorithm to obtain the optimal EIIoT malware spread-suppression strategy for the game SGMSS as in Algorithm 3.

V. Experimental Results and Evaluation

Here, we utilize Python to conduct experimental simulations based on idsgame (https://github.com/Limmen/gym-idsgame) [40]. With the consideration of indirect observation of attackers' behaviour, idsgame models stochastic games with POMDP between attackers and defenders from the defenders' perspective, which evolves itself into an optimal strategy without human intervention [41]–[43]. We implement an EIIoT malware spread-suppression environment, in which the edge nodes are trained via DQN-improved algorithms, and the IoT malware follows a random spread policy. In the given idsgame environment, we add the proposed DQN-improved algorithms including DDQMS, D2QMS and D3QMS to derive the optimal EIIoT malware spread-suppression strategy, as well as explore the influence of the related parameters on decision-making process and compare the performance among these three algorithms.

For this experiment, we set replay memory size and batch size as 10,000 and 32, respectively. The FNN (Fuzzy Neural Network) model consists of one input layer, two hidden layers, and one output layer. We apply ReLU (Rectified Linear Unit) as the hidden activation type, Huber() as the loss function calculator, and Adam as the loss function optimiser. We perform simulations using 16,000 episodes and plot every 400 episodes.

A. Learning Rate Influence on the EIIoT Malware Spread-Suppression Strategy Selection

Here, we explore the influence of the learning rate on the EIIoT malware spread-suppression strategy selection. Generally, when the learning rate is relatively small, the training convergence becomes slower and it requires much more episodes to reach the locally optimal result. Conversely, when the learning rate is relatively high, the training probably merely reaches the locally suboptimal result. Thus, according to [44], [45] and our model, we change the learning rate as $\alpha = 4e - 4$, $\alpha = 4e - 3$, and $\alpha = 4e - 2$, as well as set the initial discount factor $\gamma = 0.99$ and exploration rate $\varepsilon = 0.1$. We then compare this parameter on the EIIoT malware spread-suppression strategy selection in terms of the successful spread rate based on DDQMS, D2QMS and D3QMS.

We describe the successful spread rate obtained by the edge nodes applying DDQMS, D2QMS and D3QMS with

This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2024.3381281

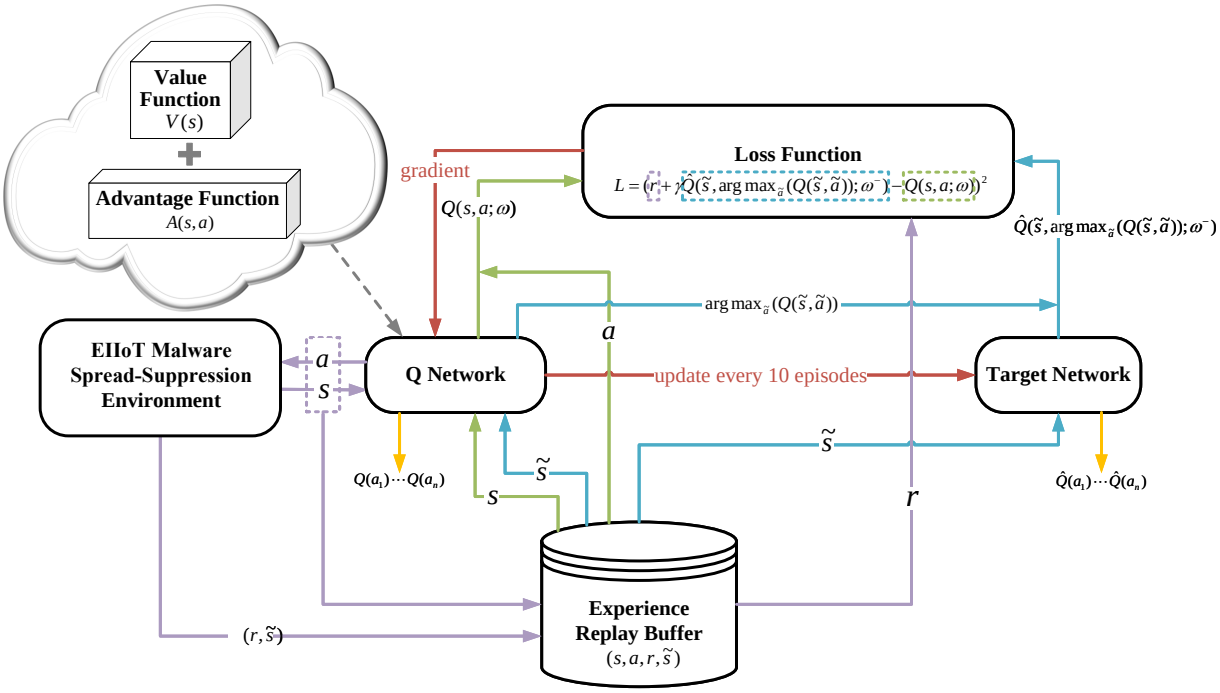

Fig. 4. D3QMS structure.

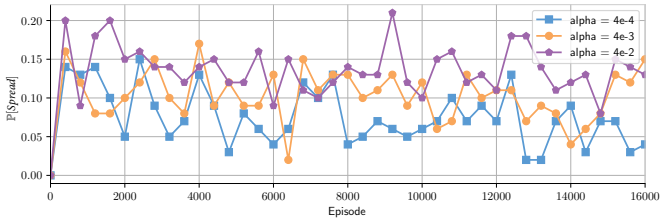

Fig. 5. Influence of learning rate α on the successful spread rate based on DDQMS.

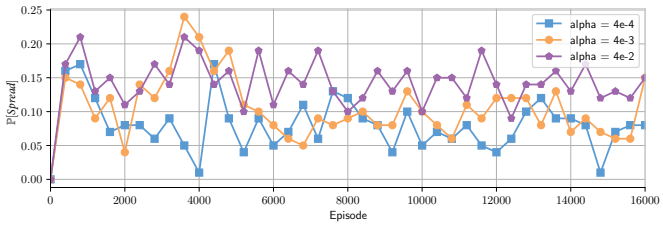

Fig. 6. Influence of learning rate α on the successful spread rate based on D2QMS.

learning rates $\alpha = 4e-4$, $\alpha = 4e-3$, and $\alpha = 4e-2$, respectively. As can be seen from Figs. 5–7, although there are irregular deviations, the probability of successful suppression is all lower than 0.2 in these three cases, elucidating that all the DDQMS-, D2QMS-, and D3QMS-trained edge nodes possess the splendid capability to suppress the spread of IoT malware. It is notable that the successful spread rate received by the D2QMS-trained edge nodes with $\alpha = 4e-3$ presents

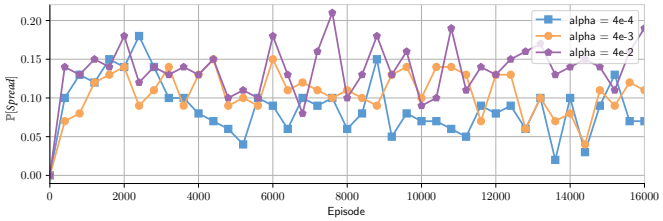

Fig. 7. Influence of learning rate α on the successful spread rate based on D3QMS.

a catabatic trend as shown in Fig. 6, fluctuating between 0.10 and 0.25 for the first 6,000 episodes and stabilising around 0.10 for the last 10,000 episodes. Moreover, the D2QMS-trained edge nodes with $\alpha = 4e-4$ overwhelm the edge nodes with $\alpha = 4e-3$ and $\alpha = 4e-2$ with a probability of 63%, which is 25 out of 40 times. In addition, as shown in Figs. 5 and 7, the edge nodes trained by DDQMS and D3QMS with $\alpha = 4e-4$ transcend the other two cases in 70% of cases, which is both 28 out of 40 times. Consequently, the edge nodes based on DDQMS, D2QMS and D3QMS with $\alpha = 4e-4$ outperform on the successful spread rate than those of $\alpha = 4e-3$ and $\alpha = 4e-2$. As a corollary, $\alpha = 4e-4$ can be identified as the optimal learning rate for the DDQMS, D2QMS and D3QMS algorithms.

B. Discount Factor Influence on the EIIoT Malware Spread-Suppression Strategy Selection

Here, we explore the influence of the discount factor on the EIIoT malware spread-suppression strategy selection. In order to perform well in a long term, we are required to consider

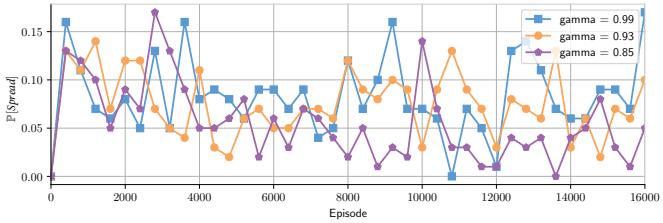

Fig. 8. Influence of discount factor γ on the successful spread rate based on DDQMS.

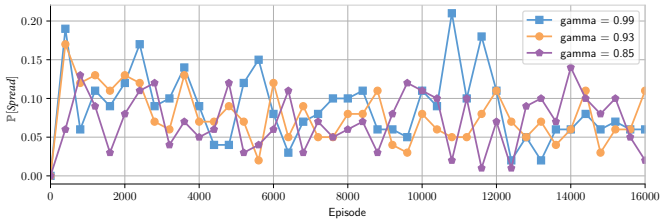

Fig. 9. Influence of discount factor γ on the successful spread rate based on D2QMS.

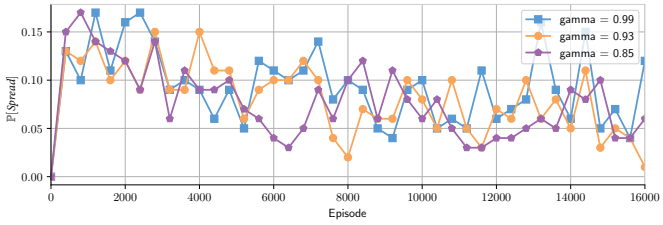

Fig. 10. Influence of discount factor γ on the successful spread rate based on D3QMS.

not only the immediate rewards but also the future rewards. We therefore introduce a discount factor. Nevertheless, when the discount factor is relatively small, the obtained strategies become too short-sighted, which are based entirely on the immediate rewards. To balance the present and future results, the discount factor is usually set close to $\gamma = 1$ [46]–[49], meaning that the future rewards are taken into account while calculating the values generated by current actions. Thus, according to the expert experience and our model, we change the discount factor as $\gamma = 0.99$, $\gamma = 0.93$, and $\gamma = 0.85$, as well as set the initial learning rate as $\alpha = 4e-4$ and exploration rate $\varepsilon = 0.1$. We then compare this parameter on the EIIoT malware spread-suppression strategy selection in terms of the successful spread rate based on DDQMS, D2QMS and D3QMS.

We describe the successful spread rate obtained by the edge nodes applying DDQMS, D2QMS and D3QMS with discount factors $\gamma = 0.99$, $\gamma = 0.93$, and $\gamma = 0.85$, respectively. As can be seen from Figs. 8–10, despite the existence of variable oscillation, the successful spread rate is almost lower than 0.15 in all these three cases. Noticeably, in Figs. 8 and 10, the DDQMS-, and D3QMS-trained edge nodes with $\gamma = 0.85$ show a tendency to drop in spite of the persistent volatility.

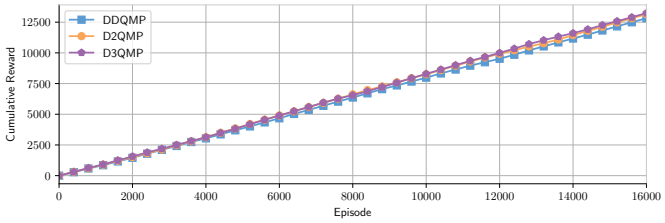

Fig. 11. Comparison of defender cumulative reward among DDQMS, D2QMS and D3QMS.

To be specific, in Fig. 8, the successful spread rate of the edge nodes utilizing the DDQMS algorithm with $\gamma = 0.85$ ranges from 0.05 to 0.15 in the first 4,000 episodes, which follows concentrating between 0.00 and 0.05 to the end and wins 21 out of 40 times. Similarly, the D2QMS-, and D3QMS-trained edge nodes with $\gamma = 0.85$ respectively precede the edge nodes with $\gamma = 0.99$ and $\gamma = 0.93$ in 55% and 53% of cases. These illustrate that the edge nodes trained via DDQMS, D2QMS and D3QMS can learn a better qualified suppression strategy via unceasing learning. Consequently, the edge nodes based on DDQMS, D2QMS and D3QMS with $\gamma = 0.85$ outperform on the successful spread rate than those of $\gamma = 0.99$ and $\gamma = 0.93$. As a corollary, $\gamma = 0.85$ can be identified as the optimal discount factor for the DDQMS, D2QMS and D3QMS algorithms.

C. Comparison of the proposed DQN-Improved Algorithms

Here, we compare the performance among DQN-improved algorithms including DDQMS, D2QMS and D3QMS in terms of the defender cumulative reward, average episode lengths, average episode loss, and successful spread rate, respectively. To guarantee consistent performance, we set the initial parameters: learning rate $\alpha = 4e-4$, discount factor $\gamma = 0.99$, and exploration rate $\varepsilon = 0.1$.

We describe the cumulative reward obtained by the edge nodes applying DDQMS, D2QMS and D3QMS algorithms. It shows an upward trend as shown in Fig. 11, meaning that the edge nodes in these three cases are incessantly learning to achieve an optimal EIIoT malware spread-suppression strategy, maximising their rewards. It is noticeable that the cumulative reward of the edge nodes trained by D3QMS acquires more cumulative reward than those of DDQMS and D2QMS after 4,000 episodes. Generally, the aim is to maximize the cumulative reward. The more cumulative reward the edge nodes obtain, the faster they reach the optimal EIIoT malware spread-suppression strategy. Thus, despite the tiny advantage, the edge nodes with D3QMS not only spend less time to attain the optimal EIIoT malware spread-suppression strategy, but they also address the overestimation as expected. Consequently, D3QMS outperforms on the defender cumulative reward than those of DDQMS and D2QMS.

We describe the average episode lengths obtained by the edge nodes applying DDQMS, D2QMS and D3QMS algorithms. As can be seen from Fig. 12, there is little effect on average episode lengths based on different algorithms.

This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2024.3381281

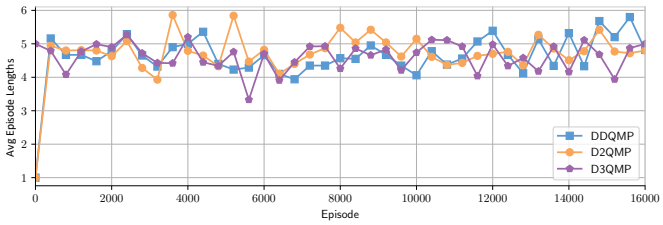

Fig. 12. Comparison of average episode lengths among DDQMS, D2QMS and D3QMS.

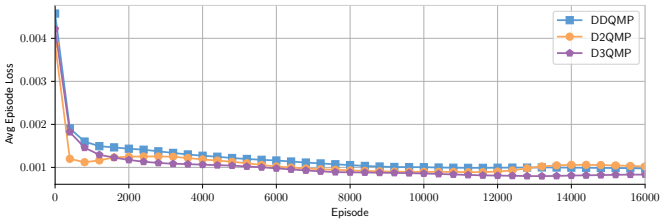

Fig. 13. Comparison of defender average episode loss among DDQMS, D2QMS and D3QMS.

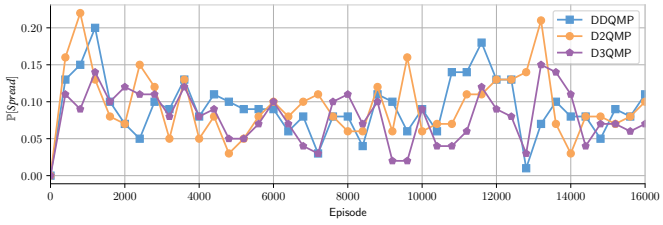

Fig. 14. Comparison of the successful spread rate among DDQMS, D2QMS and D3QMS.

All these three lines are almost rising and falling between approximately 4 steps and 5 steps. Note that the D3QMS-trained edge nodes remain less average episode lengths in more than one-third cases. Consequently, D3QMS has little advantage in average episode lengths than those of DDQMS and D2QMS.

We describe the defender average episode loss caused by the edge nodes applying DDQMS, D2QMS and D3QMS algorithms. It shows a downward trend as shown in Fig. 13. The first 400 episodes witness a plummet in all three algorithms, with the number shrinking from 0.0045 to 0.0020, 0.0039 to 0.0012, and 0.0042 to 0.0018, respectively. From then until the 16,000th episode, there is a smooth fall for the D3QMS-trained edge nodes to 0.0008 in 12,000 episodes, which remains stable at 0.0008 in the last 4,000 episodes. It is apparent that the D3QMS-trained edge nodes catch up with the DDQMS- and D2QMS-trained edge nodes in 1,600 episodes with a lower defender average episode loss of 0.0012. Consequently, D3QMS outperforms on the defender average episode loss than those of DDQMS and D2QMS.

We describe the successful spread rate obtained by the edge nodes applying DDQMS, D2QMS and D3QMS algorithms. As can be seen from Fig. 14, although there exists oscillation erratically, all these three algorithms effectively suppress the spread of IoT malware with a successful spread rate of lower than 0.15. Particularly, the successful spread rate of the D3QMS-trained edge nodes surpasses the DDQMS- and D2QMS-trained ones nearly approaching a half, which fluctuates between around 0.03 and 0.10. Nevertheless, the successful spread rates of DDQMS- and D2QMS-trained edge nodes undulate almost from 0.05 to 0.15, even exceeding 0.20 in several cases. Consequently, D3QMS outperforms on the successful spread rate than those of DDQMS and D2QMS.

In general, although there is little advantage on average episode lengths, the D3QMS-trained edge nodes hold better performance with regard to the defender cumulative reward, average episode loss, and successful spread rate. As a corollary, the D3QMS algorithm gains mastery over DDQMS and D2QMS algorithms. This is because D3QMS integrates the strengths of DDQMS and D2QMS. The former focuses on settling over-estimation of Q-values and the latter has an advantage network, which are beneficial for the edge nodes to evaluate the potential value of the chosen EIIoT malware spread-suppression strategy.

VI. Conclusion

In this paper, we have theoretically and practically proposed a stochastic games-oriented model and DQN-improved algorithms to attain the optimal learning strategy for automatically suppressing the spread of IoT malware under EIIoT, respectively. In our scheme, a spread-suppression constraint ρ is introduced to theoretically demonstrate the existence of stable Nash equilibrium. Further, a Bellman optimality backup operator $\mathcal{B}$ is brought to theoretically attest the optimal and unique EIIoT malware spread-suppression strategy for the presented game model. Moreover, we have implemented the DQN-improved algorithms including DDQMS, D2QMS and D3QMS to practically capture the optimal EIIoT malware spread-suppression strategy. In addition, we have explored the effect of related parameters on the EIIoT malware spread-suppression learning strategy selection, as well as assessed the difference among three DQN-improved algorithms. The relevant experimental simulations certify that the edge nodes using DQN-improved algorithms with learning rate $\alpha = 4e{-}4$ and discount factor $\gamma = 0.85$ are superior in automatically suppressing the spread of IoT malware. Additionally, D3QMS-trained edge nodes display a better performance than DDQMS- and D2QMS-trained ones in terms of the defender cumulative reward, average episode lengths, average episode loss, and successful spread rate.

For future work, we will focus on other parameter adjustments, such as buffer size, batch size, and the network itself, to thoroughly investigate EIIoT malware spread-suppression strategy selection. Moreover, probing the performance of deep reinforcement learning (DRL) agents against non-DRL agents on EIIoT malware spread-suppression strategy selection is another direction with great promise.

References

[1] P. Zhang, N. Chen, S. Shen, S. Yu, N. Kumar, and C. H. Hsu, "AI-enabled space-air-ground integrated networks:

Management and optimization," *IEEE Netw.*, 2023, early Access, http://dx.doi.org/10.1109/MNET.131.2200477.

[2] P. McEnroe, S. Wang, and M. Liyanage, "A survey on the convergence of edge computing and AI for UAVs: Opportunities and challenges," *IEEE Internet Things J.*, vol. 9, no. 17, pp. 15 435–15 459, Sept. 2022.

[3] J. Mills, J. Hu, and G. Min, "Communication-efficient federated learning for wireless edge intelligence in IoT," *IEEE Internet Things J.*, vol. 7, no. 7, pp. 5986–5994, Jul. 2020.

[4] T. S. Gopal, M. Meerolla, G. Jyostna, P. R. L. Eswari, and E. Magesh, "Mitigating mirai malware spreading in IoT environment," in *Proc. 2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, Bangalore, India, 2018, pp. 2226–2230.

[5] U. J. M. N. Aman and B. Sikdar, "IoT-proctor: A secure and lightweight device patching framework for mitigating malware spread in IoT networks," *IEEE Syst. J.*, vol. 16, no. 3, pp. 3468–3479, Apr. 2022.

[6] R. Yumlembam, B. Issac, S. M. Jacob, and L. Yang, "IoT-based android malware detection using graph neural network with adversarial defense," *IEEE Internet Things J.*, vol. 10, no. 10, pp. 8432–8444, Jul. 2022.

[7] Y. Shen, S. Shen, Z. Wu, H. Zhou, and S. Yu, "Signaling game-based availability assessment for edge computing-assisted IoT systems with malware dissemination," *J. Inf. Secur. Appl.*, vol. 66, May 2022, art. no. 103140.

[8] S. Shen, X. Wu, P. Sun, H. Zhou, Z. Wu, and S. Yu, "Optimal privacy preservation strategies with signaling Q-learning for edge-computing-based IoT resource grant systems," *Expert Syst. Appl.*, vol. 225, Sept. 2023, art. no. 120192.

[9] S. Shen, C. Cai, Z. Li, Y. Shen, G. Wu, and S. Yu, "Deep Q-network-based heuristic intrusion detection against edge-based SIoT zero-day attacks," *Appl. Soft. Comput.*, Jan. 2024, art. no. 111080.

[10] G. Wu, L. Xie, H. Zhang, J. Wang, S. Shen, and S. Yu, "STSIR: An individual-group game-based model for disclosing virus spread in Social Internet of Things," *J. Netw. Comput. Appl.*, vol. 214, May 2023, art. no. 103608.

[11] Y. Shen, S. Shen, Q. Li, H. Zhou, Z. Wu, and Y. Qu, "Evolutionary privacy-preserving learning strategies for edge-based IoT data sharing schemes," *Digit. Commun. Netw.*, vol. 9, no. 4, pp. 906–919, Aug. 2023.

[12] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing atari with deep reinforcement learning," in *Proc. NIPS Deep Learning Workshop 2013*, Harrahs and Harveys, Lake Tahoe, 2013, https://doi.org/10.48550/arXiv.1312.5602.

[13] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, Feb. 2015.

[14] S. Elfwing, E. Uchibe, and K. Doya, "Sigmoid-weighted linear units for neural network function approximation in reinforcement learning," *Neural Netw.*, vol. 170, no. SI, pp. 3–11, Nov. 2018.

[15] Y. Liang, M. C. Machado, E. Talvitie, and M. Bowling, "State of the art control of atari games using shallow reinforcement learning," in *Proc. AAMAS'16: The 2016 International Conference on Autonomous Agents and Multiagent Systems*, Singapore, 2016, pp. 485–493.

[16] K. Ni, D. Yu, and Y. Liu, "Attention-based deep Q-network in complex systems," in *Proc. ICNIP 2019: Neural Information Processing*, Sydney, Australia, 2019, pp. 323–332.

[17] H. Oh and T. Kaneko, "Deep recurrent Q-network with truncated history," in *Proc. 2018 Conference on Technologies and Applications of Artificial Intelligence (TAAI)*, Taichung, Taiwan, 2018, pp. 34–39.

[18] L. Zeng, Q. Liu, S. Shen, and X. Liu, "Improved double deep Q network-based task scheduling algorithm in edge computing for makespan optimization," *Tsinghua Sci. Technol.*, vol. 29, no. 3, pp. 806–817, Jun. 2024.

[19] G. Wu, Z. Xu, H. Zhang, S. Shen, and S. Yu, "Multi-agent DRL for joint completion delay and energy consumption with queuing theory in MEC-based IIoT," *J. Parallel Distrib. Comput.*, vol. 176, pp. 80–94, Jun. 2023.

[20] G. Wu, H. Wang, H. Zhang, Y. Zhao, S. Yu, and S. Shen, "Computation offloading method using stochastic games for software-defined-network-based multiagent mobile edge computing," *IEEE Internet Things J.*, vol. 10, no. 20, pp. 17 620–17 634, Oct. 2023.

[21] L. Nkenyereye, K. J. Baeg, and W. Y. Chung, "Deep reinforcement learning for containerized edge intelligence inference request processing in IoT edge computing," *IEEE Trans. Serv. Comput.*, vol. 16, no. 6, pp. 4328–4344, Nov.-Dec. 2023.

[22] S. Tang, L. Chen, K. He, J. Xia, L. Fan, and A. Nallanathan, "Computational intelligence and deep learning for next-generation edge-enabled Industrial IoT," *IEEE Trans. Netw. Sci. Eng.*, vol. 10, no. 5, pp. 2881–2893, Sept.-Oct. 2023.

[23] C. Xu, J. Ge, Y. Li, Y. Deng, L. Gao, M. Zhang, Y. Xiang, and X. Zheng, "SCEI: A smart-contract driven edge intelligence framework for IoT systems," *IEEE Trans. Mob. Comput.*, 2023, early Access, http://dx.doi.org/10.1109/TMC.2023.3290925.

[24] H. Xiao, C. Xu, Y. Ma, S. Yang, L. Zhong, and G. M. Muntean, "Edge intelligence: A computational task offloading scheme for dependent IoT application," *IEEE Trans. Wirel. Commun.*, vol. 21, no. 9, pp. 7222–7237, Sept. 2022.

[25] R. Ke, Y. Zhuang, Z. Pu, and Y. Wang, "A smart, efficient, and reliable parking surveillance system with edge artificial intelligence on IoT devices," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 8, pp. 4962–4974, Aug. 2021.

[26] M. A. Reh, J. Abu-Khalaf, P. Szewczyk, and J. J. Kang, "Malbot-drl: Malware botnet detection using deep reinforcement learning in IoT networks," *IEEE Internet Things J.*, 2023, early Access, http://dx.doi.org/10.1109/JIOT.2023.3324053.

[27] S. Shen, L. Xie, Y. Zhang, G. Wu, H. Zhang, and S. Yu, "Joint differential game and double deep Q-networks for suppressing malware spread in Industrial Internet of Things," *IEEE Trans. Inf. Forensic Secur.*, vol. 18, pp. 5302–5315, Aug. 2023.

[28] Y. Zhang, Y. Sui, S. Pan, Z. Zheng, B. Ning, I. Tsang, and W. Zhou, "Familial clustering for weakly-labeled android malware using hybrid representation learning," *IEEE Trans. Inf. Forensic Secur.*, vol. 15, pp. 3401–3414, Oct. 2019.

[29] H. Benaddi, K. Ibrahimi, A. Benslimane, M. Jouhari, and J. Qadir, "Robust enhancement of intrusion detection systems using deep reinforcement learning and stochastic game," *IEEE Trans. Veh. Technol.*, vol. 71, no. 10, pp. 11 089–11 102, Oct. 2022.

[30] B. Li, T. Shi, W. Zhao, and N. Wang, "Reinforcement learning-based intelligent reflecting surface assisted communications against smart attackers," *IEEE Trans. Commun.*, vol. 70, no. 7, pp. 4771–4779, May 2022.

[31] X. Liu, H. Zhang, S. Dong, and Y. Zhang, "Network defense decision-making based on a stochastic game system and a deep recurrent Q-network," *Comput. Secur.*, vol. 111, Dec. 2021, art. no. 102480.

[32] M. G. N. Dunstatter, A. Tahsini and J. Tesic, "Solving cyber alert allocation Markov games with deep reinforcement learning," in *Proc. GameSec 2019: Decision and Game Theory for Security*, Stockholm, Sweden, 2019, pp. 164–183.

[33] L. Zhang, T. Zhu, F. K. Hussain, D. Ye, and W. Zhou, "A game-theoretic method for defending against advanced persistent threats in cyber systems," *IEEE Trans. Inf. Forensic Secur.*, vol. 18, pp. 1349–1364, Dec. 2022.

[34] E. Altman and A. Shwartz, "Constrained Markov games: Nash equilibria," in *Proc. 7th International Symposium on Dynamic Games and Applications*, Kanagawa, Japan, 2000, pp. 213–221.

[35] M. L. Puterman, *Markov decision processes: Discrete stochastic dynamic programming*. John Wiley & Sons, Inc., 1994, http://dx.doi.org/10.1002/9780470316887.

[36] J. Rincon-Zapatero and C. Rodriguez-Palmero, "Existence and uniqueness of solutions to the Bellman equation in the unbounded case," *Econometrica*, vol. 71, no. 5, pp. 1519–1555, Oct. 2003.

[37] P. A. Estevez and Y. Okabe, "Max-min propagation nets: learning by delta rule for the chebyshev norm," in *Proc. Proceedings of 1993 International Conference on Neural Networks (IJCNN)*, Nagoya, Japan, 1993, pp. 524–527.

[38] H. van Hasselt, A. Guez, and D. Silver, "Deep reinforcement learning with double Q-learning," in *Proc. 30th Association-for-the-Advancement-of-Artificial-Intelligence (AAAI) Conference on Artificial Intelligence*, Phoenix, AZ, 2016, pp. 2094–2100.

[39] Z. Wang, T. Schaul, M. Hessel, H. van Hasselt, M. Lanctot, and N. de Freitas, "Dueling network architectures for deep reinforcement learning," in *Proc. 33rd International Conference on Machine Learning*, New York, NY, 2016, pp. 1995–2003.

[40] K. Hammar and R. Stadler, "Finding effective security strategies through reinforcement learning and self-play," in *Proc. 2020 16th International Conference on Network and Service Management (CNSM)*, Izmir, Turkey, 2020, pp. 1–9.

[41] K. Hammar and R. Stadler, "Learning near-optimal intrusion responses against dynamic attackers," *IEEE Trans. Netw. Serv. Manag.*, 2023, early Access, http://dx.doi.org/10.1109/TNSM.2023.3293413.

[42] K. Hammar and R. Stadler, "Intrusion prevention through optimal stopping," *IEEE Trans. Netw. Serv. Manag.*, vol. 19, no. 3, pp. 2333–2348, May 2022.

[43] K. Hammar and R. Stadler, "Learning intrusion prevention policies through optimal stopping," in *Proc. 2021 17th International Conference on Network and Service Management (CNSM)*, Izmir, Turkey, 2021, pp. 509–517.
[44] S. Sun, Z. Cao, H. Zhu, and J. Zhao, "A survey of optimization methods from a machine learning perspective," *IEEE Trans. Cybern.*, vol. 50, no. 8, pp. 3668–3681, Nov. 2019.
[45] L. N. Smith, "Cyclical learning rates for training neural networks," in *Proc. 2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, Santa Rosa, CA, USA, 2017, pp. 464–472.
[46] X. Wang, Y. Han, C. Wang, Q. Zhao, X. Chen, and M. Chen, "In-edge AI: Intelligentizing mobile edge computing, caching and communication by federated learning," *IEEE Netw.*, vol. 33, no. 5, pp. 156–165, Jul. 2019.
[47] H. Yang, Z. Xiong, J. Zhao, D. Niyato, L. Xiao, and Q. Wu, "Deep reinforcement learning-based intelligent reflecting surface for secure wireless communications," *IEEE Trans. Wirel. Commun.*, vol. 20, no. 1, pp. 375–388, Jan. 2021.
[48] Y. Lu, X. Huang, K. Zhang, S. Maharjan, and Y. Zhang, "Low-latency federated learning and blockchain for edge association in digital twin empowered 6G networks," *IEEE Trans. Ind. Inform.*, vol. 17, no. 7, pp. 5098–5107, Aug. 2020.
[49] A. Masadeh, M. Alhafnawi, H. A. B. Salameh, A. Musa, and Y. Jararweh, "Reinforcement learning-based security/safety uav system for intrusion detection under dynamic and uncertain target movement," *IEEE Trans. Eng. Manage.*, 2022, early Access, http://dx.doi.org/10.1109/TEM.2022.3165375.

Yizhou Shen received her B.Sc. (Hons.) degree in Computer Science from Cardiff University, Cardiff, United Kingdom, in 2023. She is currently pursuing the Ph.D. degree in Computer Science with Newcastle University, Newcastle upon Tyne, United Kingdom. Her research interests include cyber security, game theory, deep reinforcement learning, edge intelligence, and Internet of Things.

Carlton Shepherd received his PhD in information security from Royal Holloway, University of London, UK, and his BS in computer science from Newcastle University, UK. He is currently a Lecturer (Assistant Professor) in Computer Science at Newcastle University. Before this, he was a Senior Research Fellow at the Information Security Group at Royal Holloway, University of London, UK between 2020-2023. His research interests centre around the security of trusted execution environments (TEEs) and their applications, secure CPU design, embedded systems, applied cryptography, and hardware security.

Chuadhry Mujeeb Ahmed received the Ph.D. degree in information systems technology and design from Singapore University of Technology and Design (SUTD), Singapore, in 2019, under the supervision of Prof. A. Mathur, Prof. J. Zhou, and Prof. M. Ochoa.

He is currently a Senior Lecturer in Computer Science at Newcastle University. During the Ph.D., he worked on the Cyber Physical Systems Security, iTrust Labs and Testbeds, SUTD. He also worked with Professor Raheem Beyah at Georgia Tech, Atlanta, GA, USA, during his Ph.D. exchange program. His research interests are in the security and privacy of cyber–physical systems, Internet of Things, communication systems, and critical infrastructures.

Shui Yu (Fellow, IEEE) received the B.Eng. degree in electronic engineering, the Associate degree in mathematics, and the M.Eng. degree in computer science from the University of Electronic Science and Technology of China, Chengdu, China, in 1993, 1993, and 1999, respectively, and the Ph.D. degree in computer science from Deakin University, Melbourne, VIC, Australia, in 2004.

He is a Professor with the School of Computer Science, University of Technology Sydney, Sydney, NSW, Australia. He has published three monographs and edited two books, more than 600 technical papers, including top journals and top conferences, such as IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING, IEEE TRANSACTIONS ON COMPUTERS, IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, IEEE TRANSACTIONS ON MOBILE COMPUTING, IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, IEEE/ACM TRANSACTIONS ON NETWORKING, and INFOCOM. His H-index is 68. His research interest includes big data, security and privacy, networking, and mathematical modeling.

Prof. Yu initiated the research field of networking for big data in 2013, and his research outputs have been adopted by industrial systems. He is currently serving for number of prestigious editorial boards, including IEEE Communications Surveys and Tutorials (Area Editor), IEEE Transactions on Computational Social Systems, and IEEE Internet of Things Journal. He is a member of AAAS and ACM, and a Distinguished Lecturer of IEEE Communication Society.

Tingting Li is currently a Lecturer (Assistant Professor) in CyberSecurity with Centre for Cyber Security Research at Cardiff University, and holds an Honorary Research Fellow position at Imperial College London. Her research interests primarily lie in AI for cyber security, knowledge representation and reasoning.

Prior to joining Cardiff, she was a PostDoctoral Research Associate with Prof. Chris Hankin at the Institute for Security Science & Technology, Imperial College London. She obtained her PhD degree in Artificial Intelligence from University of Bath under the supervision of Dr. Julian Padget and Dr. Marina De Vos. She also received her MSc degree in Computing at Imperial College London, and her Bachelor degree in Information Security at Xidian University, China.