Learning-Based Cost-Aware Defense of Parallel Server Systems against Malicious Attacks

Yuzhen Zhan and Li Jin

***Abstract*— We consider the cyber-physical security of parallel server systems, which is relevant for a variety of engineering applications such as networking, manufacturing, and transportation. These systems rely on feedback control and may thus be vulnerable to malicious attacks such as denial-of-service, data falsification, and instruction manipulations. In this paper, we develop a learning algorithm that computes a defensive strategy to balance technological cost for defensive actions and performance degradation due to cyber attacks as mentioned above. We consider a zero-sum Markov security game. We develop an approximate minimax-Q learning algorithm that efficiently computes the equilibrium of the game, and thus a cost-aware defensive strategy. The algorithm uses interpretable linear function approximation tailored to the system structure. We show that, under mild assumptions, the algorithm converges with probability one to an approximate Markov perfect equilibrium. We first use a Lyapunov method to address the unbounded temporal-difference error due to the unbounded state space. We then use an ordinary differential equation-based argument to establish convergence. Simulation results demonstrate that our algorithm converges about 50 times faster than a representative neural network-based method, with an insignificant optimality gap between 4%–8%, depending on the complexity of the linear approximator and the number of parallel servers.**

***Index Terms*— Cyber-physical security, stochastic games, reinforcement learning.**

I. Introduction

A. Motivation

Modern parallel server systems, such as cloud computing [1], industrial production lines [2], and intelligent transportation networks [3], rely on dynamic routing to optimize performance (delay and throughput). However, routing performance depends on connected and autonomous components subject to inherent cyber-physical security risks, especially in the face of increasingly sophisticated malicious cyberattacks [4]. Cyberattacks including Denial of Service (DoS), data falsification, and routing deception can severely compromise system performance [5]–[7]. In transportation, falsifed traffic information can mislead vehicles and cause congestion [8]. For web services, a falsified blocking report in a web server farm could redirect traffic to an already overloaded server [9]. Similar risks plague production lines and communication networks, where strategic attacks can inject erroneous instructions to disrupt critical operations [10]. Actual incidents have been reported [11]–[13]. Given the criticality of these systems in modern society, it is essential to address the potential physical performance degradation due to cyber security risks [14].

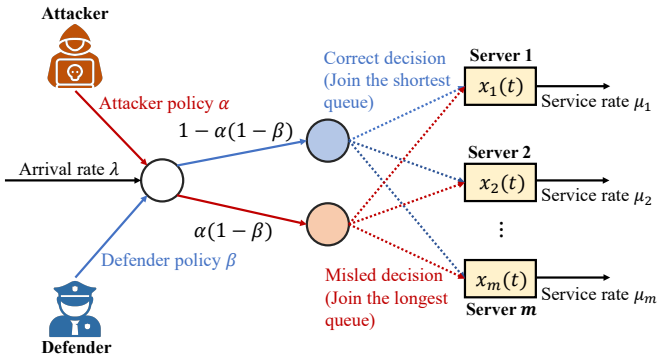

Fig. 1: An m-server system with join-the-shortest-queue routing subject to security failures.

Game theory has emerged as a powerful analytical framework for understanding cyber-physical security threats, particularly for strategic interactions between attackers and defenders [15]. By modeling the adversarial relationship as a game, a defender can anticipate attack strategies and optimize resource allocation to reduce potential losses [16]. Static security game models have been successfully used to study long-term trade-offs between security investments and defense effectiveness [17], [18]. Dynamic/stochastic game models such as the one considered in this paper are more suitable for real-time attacker-defender interactions [16], [19], [20]. However, these models usually have complex dynamics. Consequently, computational complexity is a critical bottleneck, which limits their applicability in realistic engineering systems.

One promising solution to the above challenges is reinforcement learning (RL) methods [21], [22]. RL methods are capable of computing optimal/equilibrium strategies in complex Markov games with unknown model information [23]–[25]. Since we consider parallel server systems with unbounded state spaces, we will use value function approximation. However, the broad class of neural network-based methods usually lack theoretical guarantees on training convergence and system stability [26], [27], which are essential for cyber-physical security analysis. Existing convergence guarantees for approximate methods are mostly developed for finite-state or bounded-state problems; very limited results have been developed for games with unbounded state spaces [28]. The key challenge for unbounded state spaces is that the ∞-norm-

This work was in part supported by NSFC Project 62473250, SJTU UM Joint Institute, and J. Wu & J. Sun Endowment Fund.

The authors are with the UM Joint Institute, Shanghai Jiao Tong University, China. (Emails: zyzhen1@sjtu.edu.cn, li.jin@sjtu.edu.cn.)

based argument used for finite/bounded problems does not directly apply.

B. Related work

For Markov decision processes with unbounded state space but with bounded reward, learning methods have been understood fairly well [29], [30]. For problems with unbounded reward, most studies considered convergence guarantees for RL methods under fairly stringent conditions, such as rapidly decaying discount factors [31], implicitly modified discount rates [32], and linear transition dynamics or linear reward [33]. In particular, Melo et al. [34] proposed a set of milder conditions under which Q-learning with linear function approximation converges with probability one, which provides a solid foundation for our analysis.

For games with finite/bounded state spaces, extensive learning methods have been developed and studied [35], [36]. In particular, Littman [37] proposed the minimax Q (MQ) learning algorithm for finite-state games; our algorithm builds on this classical baseline. Szepesvári and Littman [38] established convergence guarantees for this algorithm. Hu et al. [39] introduced Nash Q-learning, extending MQ to general-sum games. Lowe et al. [40] extended MQ to cooperative-competitive environments by incorporating neural networks. Recently, Chen et al. [41] extended the MQ learning to payoff-based scenarios with function approximation; although this work considers bounded state spaces, it gives useful insights for unbounded settings.

For games with unbounded state spaces, very limited learning-based methods have been developed. For such games, existing results typically focus on dynamic programming methods and the existence of a stationary equilibrium [28], [42]. A typical method to address unboundedness is to use Lyapunov functions, which is commonly used in for control of queuing systems [16], [43], [44]. However, Lyapunov methods are more suitable for traffic stabilization but are less helpful for equilibria computation.

A standard approach to convergence analysis of stochastic approximation is the the ordinary differential equation (ODE) method. The approach was formalized and extended by Borkar and Meyn [45]. Recent advancements such as quasi-stochastic approximations and dynamical systems views have improved its convergence rate and adaptation to more general noise conditions [46], [47]. However, current ODE-based frameworks often assume global Lipschitz continuity and noises with bounded variance—conditions that are relatively strict in practical scenarios [48]. The key to establish convergence of MQ learning in our setting is to jointly study the behavior of the traffic state and the approximate Q function. To the best of our knowledge, this problem has not been fully understood, especially in a game-theoretic setting.

C. Our contributions

In this paper, we develop an approximate minimax-Q (AMQ) learning algorithm to compute a near-equilibrium cost-aware defensive strategy for parallel server systems subject to malicious attacks (Fig. 1). This algorithm extends the classical MQ learning algorithm to the unbounded setting of parallel server systems. We use linear value function approximation, with a rather broad class of bases, to cover the unbounded state space. Importantly, we design the structure of the approximate Q function with insights about the systems dynamics, which makes the weights interpretable. We also provide a Foster-Lypaunov drift-based qualification for the behavior policy. We show that a qualified behavior policy must exist if the system is stabilizable.

The main result (Theorem 1) states that the proposed AMQ algorithm converges almost surely to an approximate Markov perfect equilibrium of the security game if the learning rates α_k satisfy the standard Robbins-Monro conditions. To show convergence, we adopt an ordinary differential equation-based argument by the Borkar and Meyn theorem [45]. We utilize properties of the queuing system and the feature functions to verify the Foster-Lyapunov drift condition [49]. Instead of the typical requirement of uniform boundedness of feature functions, our result only requires the boundedness of the expectation with respect to the equilibrium distribution, which is much less restrictive. The result offers insights into the learning of effective and cost-aware defense mechanisms in real-life scenarios.

To assess the performance of the AMQ method, we conduct numerical experiments with a neural network Q (NNQ) function as the benchmark. We demonstrate that the AMQ method computes defense strategies that is $94.1\% - 97.5\%$ (depending on the dimension of the approximators and the number of parallel servers) consistent with the NNQ method. The AMQ method approximates the equilibrium value function with an average error of $4.3\% - 8.2\%$. Additionally, the NNQ method converges after approximately 2×10^6 iterations, whereas the AMQ method achieves convergence in around 10^4 iterations. These results indicate that, in addition to the theoretical convergence guarantee, which is usually unavailable for neural networks, the AMQ method converges faster and attains an insignificant approximation error or optimality gap.

In summary, our main contributions include:

1) Development of an approximate minimax-Q learning algorithm for the Markov security game on parallel server systems,
2) Convergence analysis of the proposed algorithm under rather mild assumptions, and
3) Numerical experiments to validate the accuracy and efficiency of the proposed algorithm.

The rest of this paper is structured as follows. Section II presents the cyber-physical model and the approximate minimax-Q learning algorithm. Section III studies the convergence property of learning algorithm. Section IV conducts a numerical validation. Section V gives conclusions.

II. Model and Algorithm

In this section, we model the parallel server system and the strategic players, formulate the Markov security game, develop the function approximation scheme, and present the approximate minimax-Q learning algorithm.

A. System and players

Consider the parallel server system in Fig. 1. Jobs arrive according to a Poisson process of rate $\lambda > 0$ and go to one of the m servers. The ith server has exponentially distributed service times with service rate $\mu_i > 0$. Let $x(t) \in \mathbb{Z}^m_{\geq 0}$ be the vector of the number of jobs in the servers, either waiting or being served. In the absence of attacks, we assume that an incoming job is routed to the server with the shortest queue; ties are broken uniformly at random. We select this policy because of its intuitiveness and popularity in practice [50].

We characterize the security problem as a two-player zero-sum game between a defender and an attacker. An attacker is able to manipulate the routing decision for an incoming job. The attacking cost is $c_1 > 0$ per unit time. A defender can defend the routing decision for an incoming job, at a cost of $c_2 > 0$ per unit time. These costs account for the resources that attacking/defending actions have to consume. If a routing decision is attacked and is not defended, the job will go to the longest server, as the consequence of a misled decision. Otherwise, the job will join the shortest queue correctly. Ties are broken uniformly at random. See Fig. 1.

The action space for the attacker is $\{0, 1\}$, where $a(t) = 0$ (resp. $a(t) = 1$) means "not to attack" (resp. "to attack") at time t. The action space for the defender is also $\{0, 1\}$, where $b(t) = 0$ (resp. $b(t) = 1$) means "not to defend" (resp. "to defend") at time t. The instantaneous reward (resp. cost) for the attacker (resp. defender) at time t is defined as

$$\rho\Big(x(t), a(t), b(t)\Big) := \|x(t)\|_1 - c_1 a(t) + c_2 b(t), \qquad (1)$$

where $\|\cdot\|_1$ is the 1-norm. The action-induced costs are included in the reward/cost function, since both players may be interested in maximizing the opponent's costs. Note that the above reward/cost function assumes that both the traffic state and the opponent's action are observable to both players. This assumption is technologically reasonable in many scenarios. We are aware that there exist more sophisticated information structures in the literature; we do not consider them in this paper, since our focus is the coupling between the security game and the traffic dynamics. We believe that our analysis provides a basis for study of more sophisticated models.

Note that the queuing cost term $\|x\|_1$ will grow unboundedly (regardless of the players' actions) if the traffic demand λ exceeds the total capacity $\sum_k \mu_k$. To exclude this less interesting case, we assume the following:

Assumption 1. *The parallel server system is stabilizable in the sense that* $\lambda < \sum_{n=1}^m \mu_n$.

Under this assumption, the default join-the-shortest-queue routing policy is guaranteed to stabilize the traffic states if every job is routed correctly [16]. Hence, there exists at least one defending policy (i.e., $b(t) = 1$ for all t) that ensures traffic stability.

B. Security game

Since we consider a version of off-policy learning algorithm, we differentiate the notations for the behavior policy and for the target policy. We use $\alpha(a|x) : \{0, 1\} \times \mathbb{Z}^m_{\geq 0} \to [0, 1]$ (resp. $\beta(b|x) : \{0, 1\} \times \mathbb{Z}^m_{\geq 0} \to [0, 1]$) to denote the probabilistic behavior policy for the attacker (resp. defender). We use $\pi(a|x) : \{0, 1\} \times \mathbb{Z}^m_{\geq 0} \to [0, 1]$ (resp. $\sigma(b|x) : \{0, 1\} \times \mathbb{Z}^m_{\geq 0} \to [0, 1]$) to denote the probabilistic target policy for the attacker (resp. defender). Given a policy pair (α, β), the transition dynamics of the parallel server system can be specified as follows. Let $e_i \in \{0, 1\}^m$ denote the unit vector that has a 1 in the i-th entry and 0 elsewhere. Then the transition rate $q_{\alpha,\beta} : \mathbb{Z}^m_{\geq 0} \times \mathbb{Z}^m_{\geq 0} \to \mathbb{R}_{\geq 0}$ of the traffic state under the policy pair (α, β) is given by

$$q_{\alpha,\beta}(y|x) = \begin{cases} \Big(\frac{\alpha(0|x)}{|\arg\min_j x_j|} + \frac{\alpha(1|x)\beta(1|x)}{|\arg\min_j x_j|}\Big)\lambda & \\ \qquad\qquad \text{if } y \in \{x + e_i; i \in \arg\min_j x_j\}, & \\ \frac{\alpha(1|x)\beta(0|x)}{|\arg\min_j x_j|}\lambda & \text{if } y \in \{x + e_i; i \in \arg\max_j x_j\}, \\ \mu_i & \text{if } y = x - e_i, \\ 0 & \text{otherwise}, \end{cases} \qquad (2)$$

where $|\cdot|$ denotes the cardinality of a set. We exclude the case of self-transition since it does not affect our analysis.

Since the system state is countable and changes only at discrete epochs, we can reformulate the Markov security game in discrete time (DT). Note that a DT formulation also facilitates the design of learning algorithm. Specifically, let t_k be the kth transition epoch of the continuous-tim (CT) process $\{x(t); t \geq 0\}$. With a slight abuse of notation, let

$$x_k = x(t_k),\ a_k = a(t_k),\ b_k = b(t_k), \quad k = 0, 1, ...$$

Thus, the transition probabilities $p(x'|x, a, b)$ for the DT process can be obtained from the transition rates in (2) by the classical theory of countable-state Markov processes [16]. The expected one-step reward/cost is given by

$$r\Big(x_k, a_k, b_k\Big) := \rho\Big(x(t_{k-1}), a(t_{k-1}), b(t_{k-1})\Big)\mathbb{E}[\Delta t_k | x_k, a_k, b_k], \qquad (3)$$

where $\Delta t_k = t_k - t_{k-1}$ is the exponentially distributed inter-transition interval. The queuing dynamics ensure that $\mathbb{E}[\Delta t_k]$ exists for any x_k, a_k, b_k. Now we are ready to formally define the security game to be considered:

Definition 1. We consider a Markov game specified by a tuple $(\mathbb{Z}^m_{\geq 0}, \mathcal{A}, \mathcal{B}, p, r, \gamma)$ defined as follows.

1) $\mathbb{Z}^m_{\geq 0}$ is the traffic state space of the parallel server system.
2) $\mathcal{A}$ (resp. $\mathcal{B}$) is the space of (mixed) strategies for the attacker (resp. defender).
3) $p : (\mathbb{Z}^m_{\geq 0} \times \{0, 1\}^2) \times \mathbb{Z}^m_{\geq 0} \to [0, 1]$ is the transition probability of the traffic state under a given pair of actions; these probabilities can be computed readily from the CT transition rates q.
4) $r : \mathbb{Z}^m_{\geq 0} \times \{0, 1\}^2 \to \mathbb{R}$ is the one-step reward/cost function.
5) $\gamma \in (0, 1)$ is the discount rate.

By the DT formulation, the value/cost function is thus given by

$$v_{\pi,\sigma}(x) = \mathbb{E}_{\pi,\sigma}\left[\sum_{k=0}^{\infty} \gamma^k r(x_k, a_k, b_k) \middle| x_0 = x\right].$$

In the zero-sum game, the attacker (resp. defender) attempts to maximize (resp. minimize) the above. Since the state space is unbounded, the existence of $v_{\pi,\sigma}$ is not readily guaranteed for any policy pair. Fortunately, the existence was proved in [16] under equilibria in the following sense.

Definition 2. The Markov perfect equilibrium (MPE) for the security game is a strategy pair (π^*, σ^*) such that for any $x \in \mathbb{Z}^m_{\geq 0}$,

$$\pi^*(\cdot|x) = \arg\max_{\pi} v_{\pi,\sigma^*}(x),$$
$$\sigma^*(\cdot|x) = \arg\min_{\sigma} v_{\pi^*,\sigma}(x).$$

Hence, the MPE is characterized by the equilibrium state value function

$$v^*(x) = v_{\pi^*,\sigma^*}(x).$$

Note that the corresponding action value function is given by

$$Q_{\pi,\sigma}(x,a,b) = r(x,a,b) + \sum_{x' \in \mathbb{Z}^m_{\geq 0}} p(x'|x,a,b) v_{\pi,\sigma}(x').$$

By the Shapley theory [51], v^* is associated with a unique action value function (also called the "minimax Q function") satisfying the minimax version of the Bellman equation [38]. Following [52], we take the defender's perspective of minimax Bellman operator $\mathbf{T}$ on the space of functions $\{Q : \mathbb{Z}^m_{\geq 0} \times \{0,1\}^2 \to \mathbb{R}\}$ as

$$(\mathbf{T}Q)(x,a,b) = r(x,a,b) + \gamma \min_{\sigma \in \mathcal{B}} \max_{a' \in \{0,1\}} \sum_{\substack{x' \in \mathbb{Z}^m_{\geq 0} \\ b' \in \{0,1\}}} p(x'|x,a',b') \sigma(b'|x) Q(x',a',b').$$

Then the minimax Bellman equation can be written compactly as

$$Q^* = \mathbf{T}Q^*,$$

where Q^* is also the action value function associated with v^*.

C. Function Approximation

Consider a set of md linearly independent basis functions $\{\phi_{i,j} : \mathbb{Z}^m_{\geq 0} \times \{0,1\}^2 \to \mathbb{R}; 1 \leq i \leq m, 1 \leq j \leq d\}$. Let

$$\phi = [\phi_{1,1}, \ldots, \phi_{1,d}, \phi_{2,1}, \ldots, \phi_{2,d}, \ldots, \phi_{m,1}, \ldots, \phi_{m,d}]^\top$$

be the md-dimensional list of basis functions. We follow [53] and assume the following on regularity of the basis functions.

Assumption 2. *The basis functions ϕ satisfy the following.*

1) (Subexponential and non-negative) ϕ is such that

$$0 \leq \sum_{j=1}^{d} \phi_{i,j}(x) \leq e^{x_i} \quad for\ i \in \{1,2,\ldots,m\}.$$

2) (Dominance over gradient) There exists a constant $B > 0$ such that for x satisfying

$$\|x\|_2^2 \geq B,$$

it holds that

$$\left\| \frac{\partial \phi}{\partial x}(x) \right\|_1 < \|\phi(x)\|_1.$$

Let

$$\mathcal{Q} = \left\{ \phi^\top w;\ w \in \mathbb{R}^{md} \right\}$$

be the space spanned by the basis functions. Then the approximate function $Q_w \in \mathcal{Q}$ is given by

$$Q_w(x,a,b) = \phi(x,a,b)^\top w, \tag{4}$$

where $w \in \mathbb{R}^{md}$ is the weight vector, with $w_{i,j}$ being the weight of $\phi_{i,j}$. Note that Assumption 2 makes $w_{i,1}, w_{i,2}, \ldots, w_{i,d}$ associated with server i; this construction incorporates particularly the parallel structure of server system and thus gives interpretability of the weights.

If the behavior policy pair $(\alpha, \beta) \in \mathcal{A} \times \mathcal{B}$ ensures ergodicity of the traffic state process $\{x(t); t \geq 0\}$, let $\mu_{\alpha,\beta}$ be the invariant probability measure. We will discuss the qualifications for the behavior policy in the next subsection. With the linear function approximation, we in fact approximates the actual equilibrium value function Q^* with a projection Q_{w^*} in $\mathcal{Q}$. Denote the orthogonal projection operator by $\mathbf{P}$ on the space of functions $\{Q : \mathbb{Z}^m_{\geq 0} \times \{0,1\}^2 \to \mathbb{R}\}$, which is given by

$$(\mathbf{P}Q)(x,a,b) = \phi^\top(x,a,b) \Sigma^{-1} \mathbb{E}_{\mu_{\alpha,\beta}} \left[\phi(x,a,b) Q(x,a,b) \right], \tag{5}$$

where $\mathbb{E}_{\mu_{\alpha,\beta}}$, with a slight abuse of notation, denotes the vector of expectations with respect to the invariant probability measure $\mu_{\alpha,\beta}$. The most intuitive approximation scheme is to directly project Q^* on $\mathcal{Q}$ and obtain function Q_{w^*} as

$$Q_{w^*}(x,a,b) = (\mathbf{P}Q^*)(x,a,b) = (\mathbf{P}\mathbf{T}Q^*)(x,a,b).$$

However, we generally can not obtain Q^* analytically; otherwise we would not have to approximate. Q_{w^*} here is also not a fixed point of any involved operator, and there exists no obvious procedure to write a stochastic approximation algorithm to find Q_{w^*} [34]. Alternatively, we define the optimal weight vector w^* to verify

$$Q_{w^*}(x,a,b) = (\mathbf{P}\mathbf{T}Q_{w^*})(x,a,b), \tag{6}$$

and approximate Q^* with Q_{w^*} as defined above. This Q_{w^*} is actually a fixed point of the projected Bellman operator $\mathbf{PT}$. Note that the corresponding optimal weight vector w^* can also be directly defined as a fixed point of a modified projected Bellman operator. Accordingly, we follow van Eck and van Wezel [54] and consider an approximated equilibrium as defined below:

Definition 3. The *linear approximated equilibrium* for the security game is a strategy pair $(\hat{\pi}^*, \hat{\sigma}^*)$ such that for any $x \in \mathbb{Z}^m_{\geq 0}$,

$$\hat{\pi}^*(\cdot|x) = \arg\max_{\hat{\pi} \in \mathcal{A}} \sum_{a \in \{0,1\}} \hat{\pi}(a|x) \sum_{b \in \{0,1\}} \hat{\sigma}^*(b|x) \phi^\top(x,a,b) w^*,$$
$$\hat{\sigma}^*(\cdot|x) = \arg\min_{\hat{\sigma} \in \mathcal{B}} \sum_{b \in \{0,1\}} \hat{\sigma}(b|x) \sum_{a \in \{0,1\}} \hat{\pi}^*(a|x) \phi^\top(x,a,b) w^*.$$

There are multiple metrics for the quality of approximation. One is the mean error between the actual value Q^* and the approximated value Q_{w^*}. Another is the consistency between

the MPE strategy profile (π^*, σ^*) and the approximated MPE strategy profile $(\hat{\pi}^*, \hat{\sigma}^*)$. We will study these metrics numerically in Section IV.

D. Learning Algorithm

We consider an approximate minimax-Q (AMQ) learning algorithm with the following update rule for the weights:

$$\begin{aligned} w_{k+1} &= w_k + \eta_k \nabla_w Q_w(x_k, a_k, b_k)\Delta_k \\ &= w_k + \eta_k \phi(x_k, a_k, b_k)\Delta_k, \end{aligned} \tag{7}$$

where Δ_k is the temporal difference at time t_k, given by

$$\begin{aligned} \Delta_k =& r_k + \gamma \min_{\sigma\in\mathcal{B}} \max_{a\in\{0,1\}} \sum_{b\in\{0,1\}} \sigma(b|x) Q_{w_k}(x_{k+1}, a, b) \\ &- Q_{w_k}(x_k, a_k, b_k). \end{aligned} \tag{8}$$

To obtain σ at iteration k, we actually solve a linear programming as follows, where the optimum objective $c = \max_{a\in\{0,1\}} \sum_{b\in\{0,1\}} \sigma(b|x) Q_{w_k}(x_{k+1}, a, b)$.

$$\begin{aligned} \min \quad & c \\ s.t \quad & \sum_b \sigma(b|x) Q_{w_k}(x_{k+1}, a, b) \le c \qquad \forall a \in \{0,1\} \\ & \sigma(b|x) \ge 0, \qquad \sum_b \sigma(b|x) = 1 \qquad \forall b \in \{0,1\} \end{aligned} \tag{9}$$

The initial weight w_0 is arbitrary. The pseudo-code is presented below.

Algorithm 1 AMQ learning for the security game

Input:

Initial weights w_0, behavior policy α, β, step sizes sequence η_k, γ;

1: Initialize weights $w_0 \leftarrow w_0$
2: **for** $k = 0, 1, \cdots$ **do**
3: Sample $A_k \sim \alpha(\cdot|X_k)$, $B_k \sim \beta(\cdot|X_k)$
4: Receive R_{k+1} and observe X_{k+1}
5: Update Δ_k via (8) and (9)
6: Update w_k via (7)
7: **end for**

We assume the following conditions for the behavior policy pair and for the learning rates.

Assumption 3. *Let $(\alpha, \beta) \in \mathcal{A} \times \mathcal{B}$ be the behavior policy pair.*

1) *$\alpha(a|x) > 0, \beta(b|x) > 0$ for $\mu_{\alpha,\beta}$-almost all $x \in \mathbb{Z}^m_{\ge 0}$.*
2) *There exist $\nu > 0, c > 0, d < \infty$ such that with $V(x) = \sum_{n=1}^m e^{\nu x_n}$,*

$$\begin{aligned} \mathcal{L}_{\alpha,\beta} V(x) &= \sum_{y\in\mathbb{Z}^m_{\ge 0}} q_{\alpha,\beta}(y|x) V(y) - V(x) \\ &\le -cV(x) + d, \qquad \forall x \in \mathbb{Z}^m_{\ge 0}, \end{aligned}$$

where $\mathcal{L}_{\alpha,\beta}$ is the infinitesimal generator under policy pair (α, β) and $q_{\alpha,\beta}(y|x)$ is defined in (2)*.*

Assumption 4. *The learning rates satisfy*

$$\sum_{k=1}^{\infty} \eta_k = \infty, \quad \sum_{k=1}^{\infty} \eta_k^2 < \infty.$$

Assumption 3 ensures ergodicity under the behavior policy pair. The class of policy pairs satisfying this assumption is fairly broad. In fact, any ϵ-greedy-type policy pair would verify part 1). Part 2) essentially ensures positive Harris of the traffic state process. An illustrative example is provided below. Under Assumption 1, there exists a positive constant C_0 satisfying $0 < C_0 < \min\{1, \frac{\sum_{k=1}^m \mu_k - \lambda}{\lambda}\}$. Then a qualified behavior policy pair is

$$\alpha(1|x) = C_0 e^{-\frac{|x|_1}{2}}, \quad \alpha(0|x) = 1 - C_0 e^{-\frac{|x|_1}{2}}, \tag{10a}$$

$$\beta(1|x) = \begin{cases} 1 - e^{-\frac{|x|_1}{2}} & \text{if } x \ne 0^m, \\ 0.5 & \text{if } x = 0^m. \end{cases} \tag{10b}$$

$$\beta(0|x) = \begin{cases} e^{-\frac{|x|_1}{2}} & \text{if } x \ne 0^m, \\ 0.5 & \text{if } x = 0^m. \end{cases} \tag{10c}$$

One can verify that this behavior policy pair satisfies Assumption 3; see Appendix. The assumptions on the learning rates are in fact the standard Robbins-Monro conditions for convergence analysis.

Finally, the AMQ learning algorithm is said to be convergent if $w_k \to w^*$ w.p.1, where w^* verifies the projected Bellman equation (6). The next section is devoted to show this.

III. Convergence Analysis

The main result of this paper states that the approximate minimax-Q (AMQ) learning algorithm is guaranteed to converge to a solution to the projected minimax Bellman equation.

Theorem 1. *Consider the Markov security game $(\mathbb{Z}^m_{\ge 0}, \mathcal{A}, \mathcal{B}, p, r, \gamma)$ on a parallel server system. Under Assumptions 1–4, for any initial weight $w_0 \in \mathbb{R}^d$ and any initial state $x_0 \in \mathbb{Z}^m_{\ge 0}$, the approximate minimax-Q learning algorithm* (7) *converges in the sense that $w_k \to w^*$ w.p.1., where w^* verifies the projected Bellman equation* (6)*.*

Theorem 1 provides a convergence guarantee for the proposed learning method under rather mild assumptions, viz. (i) stabilizability of the parallel server system, (ii) regularity of the basis functions, (iii) ergodicity under the behavior policy pair, and (iv) Robbins-Monroe conditions for the learning rates. Thus, the AMQ algorithm will reliably generate effective defense policies for managing parallel server systems in practical scenarios.

We will prove Theorem 1 in three steps. In Section III-A, we show that the traffic state is geometrically ergodic under the behavior policy pair (Lemma 1) and that the basis function has a bounded first moment with respect to the corresponding invariant probability measure (Lemma 2). In Section III-B, we show that the first moment of the temporal-difference (TD) error is bounded by a linear function of the norm of the weight vector (Lemma 3). In Section III-C, we apply the ordinary differential equation-based argument to the first moment of the TD error and establish the convergence of the proposed algorithm.

A. Geometric ergodicity and boundedness of basis functions

Under a behavior policy pair (α, β) satisfying Assumption 3, the induced chain $(\mathcal{X}, P_{\alpha,\beta})$ is geometrically ergodic with corresponding equilibrium probability measure $\mu_{\alpha,\beta}$. To argue for the irreducibility of the induced chain, note that the state $x = 0$ can be accessible from any initial condition with positive probability. Hence, the induced chain is exponentially ergodic.

To prove the boundedness of feature functions, we first derive Lemma 1 to show the quadratic version of Lyapunov function $V(x) = \sum_{n=1}^{m} e^{\nu x_n}$ has a negative drift with $v > 0$, based on which we can then conclude the boundedness of feature functions in Lemma 2.

Lemma 1. *Suppose that assumption* $1, 3$ *hold. Let* $W(x) = (\sum_{n=1}^{m} e^{\nu x_n})^2$, $\nu > 0$. *Then there exist some* $d' < \infty$ *such that*

$$\mathcal{L}_{\alpha,\beta}W(x) = \sum_{y\in\mathbb{Z}^m_{\geq 0}} q_{\alpha,\beta}(y|x)W(y) - W(x) \leq -cW(x) + d', \quad x \in \mathbb{Z}^m_{\geq 0}, \tag{11}$$

where $\mathcal{L}_{\alpha,\beta}$, $q_{\alpha,\beta}(y|x)$ *and constant* c *are defined in Assumption 3.*

Proof. By Assumption 3 we obtain that

$$\mathcal{L}_{\alpha,\beta}\Big(\sum_{n=1}^{m} e^{\nu x_n}\Big) \leq -c\Big(\sum_{n=1}^{m} e^{\nu x_n}\Big) + d, \quad x \in \mathbb{Z}^m_{\geq 0},$$

where c, d is finite constant defined in Assumption 3. Note that $c > 0$. Then the infinitesimal generator of $W(x)$

$$\begin{aligned}\mathcal{L}_{\alpha,\beta}W(x) &= 2\Big(\sum_{n=1}^{m} e^{\nu x_n}\Big)\mathcal{L}\Big(\sum_{n=1}^{m} e^{\nu x_n}\Big)\\ &\leq -2c\Big(\sum_{n=1}^{m} e^{\nu x_n}\Big)^2 + 2d\Big(\sum_{n=1}^{m} e^{\nu x_n}\Big)\\ &= -cW(x) + d',\end{aligned}$$

where d' is a finite positive constant satisfying

$$d' = -c\Big(\sum_{n=1}^{m} e^{\nu x_n}\Big)^2 + 2d\Big(\sum_{n=1}^{m} e^{\nu x_n}\Big) \leq \frac{d^2}{c}.$$

□

Let Φ be the matrix defined as

$$\Phi = \mathbb{E}_{\mu_{\alpha,\beta}}\left[\phi(x,a,b)\phi^\top(x,a,b)\right],$$

where $\mathbb{E}_{\mu_{\alpha,\beta}}$, with a slight abuse of notation, denotes the matrix of expectations with respect to the invariant probability measure $\mu_{\alpha,\beta}$. The following result ensures the existence of Φ.

Lemma 2. *Suppose that assumption* $1-3$ *hold, the feature function* ϕ *satisfies*

$$\Big\|\mathbb{E}_{\mu_{\alpha,\beta}}\Big[\phi(x,a,b)\phi^\top(x,a,b,y)\Big\|_\infty \leq \frac{d'}{c}, \tag{12}$$

for any $i \in \{1,\dots,m\}$, *where* c, d' *is the constant in Lemma 1.*

Proof. By Lemma 1 we obtain that

$$\lim_{t\to\infty}\frac{1}{t}\int_{s=0}^{t}\mathbb{E}\Big[\sum_{i=1}^{m}\sum_{j=1}^{m} e^{\nu\left(x_i(s)+x_j(s)\right)}\Big]ds \leq \frac{d'}{c} < \infty.$$

Hence, by Assumption 2, since

$$\sum_{j=1}^{d}\phi_{i,j}(x) \leq e^{x_i} \quad for\ i \in \{1,2,\dots,m\},$$

then with $\psi(x) = \mathbb{E}_{\mu_{\alpha,\beta}}[\sum_{i=1}^{m}\sum_{j=1}^{d}\phi_{i,j}(x,a,b)] \leq \mathbb{E}_{\mu_{\alpha,\beta}}[\sum_{i=1}^{m} e^{x_i}]$ that

$$\lim_{t\to\infty}\frac{1}{t}\int_{s=0}^{t}\mathbb{E}[\psi^2(x(s))]ds \leq \frac{d'}{c}$$

for any initial condition $x(0)$. Then we can conclude that

$$\lim_{t\to\infty}\mathbb{E}[\psi^2(x(t))] \leq \frac{d'}{c},$$

which means

$$\Big\|\mathbb{E}_{\mu_{\alpha,\beta}}\Big[\phi(x,a,b)\phi^\top(x,a,b,y)\Big\|_\infty \leq \frac{d'}{c},$$

where $\mu_{\alpha,\beta}$ is the equilibrium state-action distribution under policy α, β. □

B. Boundedness of gradient

We write (7) in the form

$$w_{k+1} = w_k + \eta_k H(w_k, Y_{k+1}),$$

where $Y_{k+1} = (x_k, a_k, b_k)$, and

$$\begin{aligned}H(w,Y) =&\phi(x,a,b)\Big(r(x,a,b,y)+\\ &\gamma\min_{\sigma\in\mathcal{B}}\max_{a'\in\{0,1\}}\sum_{b'\in\{0,1\}}\sigma(b')Q_w(y,a',b') - Q_w(x,a,b)\Big).\end{aligned} \tag{13}$$

To prove the boundedness of the gradient, we mainly utilize the properties of feature function and queuing system.

Lemma 3. *The function* H *satisfies*

$$\Big\|\mathbb{E}_{\mu_{\alpha,\beta}}[H(w,x,a,b)]\Big\|_\infty \leq C(1+\|w\|_\infty), \tag{14}$$

for any w, *where* C *is a finite constant.*

Proof. Denote e_i as the unit vector with only the ith element equals 1. Also denote the longest queue as $x_{\max}$ and its corresponding index as i. Define similarly the shortest queue $x_{\min}$ and its index j. Denote by $g(x,a,b,y)$ the vector as

$$g(x,a,b,y) = \max_{\substack{a'\in\{0,1\}\\ b'\in\{0,1\}}}\phi(y,a',b') - \phi(x,a,b)$$

Hence, we can obtain by definition of (13) that

$$\begin{aligned}&\Big\|\mathbb{E}_{\mu_{\alpha,\beta}}[H(w,x,a,b)]\Big\|_\infty \leq\\ &\leq \Big\|\mathbb{E}_{\mu_{\alpha,\beta}}\Big[\phi(x,a,b)[r(x,a,b,y) + g^\top(x,a,b,y)\cdot w]\Big]\Big\|_\infty\\ &\leq \Big\|\mathbb{E}_{\mu_{\alpha,\beta}}[\phi(x,a,b)\cdot r(x,a,b,y)]\Big\|_\infty\\ &\quad + \Big\|\mathbb{E}_{\mu_{\alpha,\beta}}[\phi(x,a,b)\cdot g^\top(x,a,b,y)]\Big\|_\infty\cdot\|w\|_\infty\end{aligned} \tag{15}$$

When $\sum_{k=1}^{m} x_k^2 \geq B$, it can be deduced that $\|g^\top(x,a,b,y)\|_1 \leq \|\phi^\top(x,a,b)\|_1$ by Assumption 2. Thus by Lemma 2,

$$\begin{aligned}
&\Big\|\mathbb{E}_{\mu_{\alpha,\beta}}\Big[\phi(x,a,b)g^\top(x,a,b,y)\Big|\sum_{k=1}^{m} x_k^2 \geq B\Big] \\
&\cdot P\Big(\sum_{k=1}^{m} x_k^2 \geq B\Big)\Big\|_\infty \\
&+\Big\|\mathbb{E}_{\mu_{\alpha,\beta}}\Big[\phi(x,a,b)g^\top(x,a,b,y)\Big|\sum_{k=1}^{m} x_k^2 < B\Big] \\
&\cdot P\Big(\sum_{k=1}^{m} x_k^2 < B\Big)\Big\|_\infty \\
&< \frac{d'}{c} + \Big\|\mathbb{E}_{\mu_{\alpha,\beta}}[\phi(x,a,b)(\zeta(B))^2]\Big\|_\infty
\end{aligned}$$

where c, d' is the constant in Lemma 1, $\zeta(B)$ is a finite positive constant related to the specific form of ϕ. It can be easily derived according to different combination of state action pairs, e.g. $\zeta(B) = (\sqrt{B}+1)^2$ when adopting polynomial approximators. By Assumption 3, we obtain

$$\lim_{t\to\infty}\frac{1}{t}\int_{s=0}^{t}\mathbb{E}\Big[\sum_{i=1}^{m} e^{\nu(x_i(s))}\Big]ds \leq \frac{d}{c} < \infty.$$

where d is the constant in Assumption 3. Hence, we can derive with $\psi(x) = \mathbb{E}_{\mu_{\alpha,\beta}}[\|\phi(x,a,b)\|_1]$,

$$\lim_{t\to\infty}\frac{1}{t}\int_{s=0}^{t}\mathbb{E}[\psi(x(s))]ds \leq \frac{d}{c}$$

for any initial condition $x(0)$. Then we conclude that

$$\lim_{t\to\infty}\mathbb{E}[\psi(x(t))] \leq \frac{d}{c},$$

which implies

$$\Big\|\mathbb{E}_{\mu_{\alpha,\beta}}[\phi(x,a,b)]\Big\|_\infty \leq \frac{d}{c}.$$

Hence,

$$\Big\|\mathbb{E}_{\mu_{\alpha,\beta}}[\phi(x,a,b)g^\top(x,a,b,y)]\Big\|_\infty < \frac{1}{c}\Big(d' + d(\zeta(B))^2\Big). \tag{16}$$

Then we prove the term $\|\mathbb{E}_{\mu_{\alpha,\beta}}[\phi(x,a,b)\cdot r(x,a,b,y)]\|_\infty$ is bounded. Recall the definition of reward (3). We first denote the interval time until next arrival as Δt^1 with its distribution $f_a(\Delta t^1)$, the interval time until next service as Δt^2 with its distribution $f_s(\Delta t^2)$. The number of current activated servers (i.e. with job in queue) is defined as

$$k_x := k(x) = \sum_{i=0}^{m}\mathbb{I}\{x_i \neq 0\}.$$

It is known from property of parallel queuing system that

$$\begin{aligned}
f_a(\Delta t^1) &= \lambda\exp(-\lambda\Delta t^1) \\
f_s(\Delta t^2) &= k_x\mu\exp(-k_x\mu\Delta t^2).
\end{aligned}$$

Since $\Delta t = \min\{\Delta t^1, \Delta t^2\}$, we can derived the distribution of Δt as $f(\Delta t)$

$$f(\Delta t) = (\lambda + k_x\mu)\exp(-(\lambda+k_x\mu)\Delta t),$$

with expectation of Δt as $\mathbb{E}[\Delta t] = \frac{1}{\lambda+k_x\mu} \leq \frac{1}{\lambda}$. By definition of (1) and the fact our state $x \in \mathbb{Z}_{\geq 0}^m$,

$$\rho(x,a,b) \leq \|\phi^\top(x,a,b)\|_1.$$

Then by Lemma 2, we can conclude

$$\begin{aligned}
&\Big\|\mathbb{E}_{\mu_{\alpha,\beta}}[\phi(x,a,b)\cdot r(x,a,b,y)]\Big\|_\infty \\
&\leq \frac{1}{\lambda}\Big\|\mathbb{E}_{\mu_{\alpha,\beta}}\Big[\phi(x,a,b)\phi^\top(x,a,b,y)\Big\|_\infty \leq \frac{d'}{c\lambda}.
\end{aligned}$$

Hence, (14) can be satisfied by selecting $C = \max\Big\{\frac{1}{c}\Big(d' + d(\zeta(B))^2\Big), \frac{d'}{c\lambda}\Big\}$. □

C. Proof of Theorem 1

We first prove the convergence of approximate minimax Q learning w.p.1. Let μ_x be the corresponding invariant probability measure, $\mu_{\alpha,\beta}$ be the invariant state-action distribution under the given behavior policy pair (α,β). It verifies the existence of function

$$h(w) = \int H(w,Y)\mu_{\alpha,\beta}(dY)$$

by bound of function $H(w,Y)$ derived in Lemma 3.

Since the chain is geometrically ergodic, it follows that so is the chain Y_k. The geometric ergodicity of Y_k and the fact that α,β do not depend on w ensure that the requirements are satisfied. Hence, by [47, Theorem 17] , the convergence of w_k w.p.1 is established as long as the ODE

$$\dot{w}_k = h(w_k) \tag{17}$$

with

$$\begin{aligned}
h(w) = \mathbb{E}_{\mu_{\alpha,\beta}}\Big[\phi(x,a,b)\Big(r(x,a,b,y)+ \\
+\gamma\min_{\sigma\in\mathcal{B}}\max_{a'\in\{0,1\}}\sum_{b'\in\{0,1\}}\sigma(b')\phi^\top(y,a',b')w - \phi^\top(x,a,b)w\Big)\Big],
\end{aligned}$$

has a globally asymptotically stable equilibrium w^*.

We can write h as

$$h(w) = h_1(w) - h_2(w),$$

with

$$\begin{aligned}
h_1(w) =&\mathbb{E}_{\mu_{\alpha,\beta}}[\phi(x,a,b)(r(x,a,b,y)+ \\
&+\gamma\min_{\sigma\in\mathcal{B}}\max_{a'\in\{0,1\}}\sum_{b'\in\{0,1\}}\sigma(b')\phi^\top(y,a',b')w)]
\end{aligned}$$

and

$$h_2(w) = \mathbb{E}_{\mu_{\alpha,\beta}}[\phi(x,a,b)\phi^\top(x,a,b)w)].$$

Then using the non-expansiveness of *min* and *max* operator, we can conclude

$$\begin{aligned}
&\|h_1(w_1) - h_1(w_2)\|_\infty = \\
&\|\mathbb{E}_{\mu_{\alpha,\beta}}[\gamma\phi(x,a,b)(\min_{\sigma\in\mathcal{B}}\max_{a'\in\{0,1\}}\sum_{b'\in\{0,1\}}\sigma(b')\phi^\top(y,a',b')w_1 \\
&-\min_{\sigma\in\mathcal{B}}\max_{a'\in\{0,1\}}\sum_{b'\in\{0,1\}}\sigma(b')\phi^\top(y,a',b')w_2)]\|_\infty
\end{aligned}$$

$$\leq \|\mathbb{E}_{\mu_{\alpha,\beta}}[\gamma\phi(x,a,b)\max_{\sigma\in\mathcal{B}}(\max_{a'\in\{0,1\}}\sum_{b'\in\{0,1\}}\sigma(b')\phi^\top(y,a',b')w_1$$
$$-\max_{a'\in\{0,1\}}\sum_{b'\in\{0,1\}}\sigma(b')\phi^\top(y,a',b')w_2)]\|_\infty$$
$$\leq \gamma\|\mathbb{E}_{\mu_{\alpha,\beta}}[\phi(x,a,b)\max_{\sigma\in\mathcal{B}}\max_{a'\in\{0,1\}}\sum_{b'\in\{0,1\}}\sigma(b')\phi^\top(y,a',b')$$
$$(w_1-w_2)]\|_\infty$$
$$\leq \gamma\Big(\|\mathbb{E}_{\mu_{\alpha,\beta}}[\phi(x,a,b)\phi^\top(x,a,b)]\|_\infty+$$
$$\|\mathbb{E}_{\mu_{\alpha,\beta}}[\phi(x,a,b)g^\top(x,a,b,a',b')]\|_\infty\Big)\cdot\|w_1-w_2\|_\infty,$$

where $g(x,a,b,y)$ is defined in Lemma 3.

Actually, we can scale the feature function $\phi(x,a,b)$ arbitrarily to make h_1 be γ-contraction. Scale $\phi(x,a,b)$ by a constant factor $\varepsilon \leq \frac{\sqrt{[d'+d(\zeta(B))^2]^2+4d'c}-[d'+d(\zeta(B))^2]}{2d'}$, where B is the constant defined in Assumption 2. Then by Lemma 2 and condition (16) in Lemma 3 we can ensure

$$\begin{aligned}
&\|h_1(w_1)-h_1(w_2)\|_\infty=\\
&\leq\gamma\Big(\varepsilon^2\|\mathbb{E}_{\mu_{\alpha,\beta}}[\phi(x,a,b)\phi^\top(x,a,b)]\|_\infty+\\
&\qquad\varepsilon\|\mathbb{E}_{\mu_{\alpha,\beta}}[\phi(x,a,b)g^\top(x,a,b,y)]\|_\infty\Big)\cdot\|w_1-w_2\|_\infty\\
&\leq\gamma\Big[\frac{\varepsilon^2 d'}{c}+\varepsilon\Big(\frac{d'}{c}+\frac{d}{c}(\zeta(B))^2\Big)\Big]\cdot\|w_1-w_2\|_\infty\\
&\leq\gamma\|w_1-w_2\|_\infty.
\end{aligned}\tag{18}$$

Also, we can conclude by Lemma 2 that

$$\begin{aligned}
&\|h_2(w_1)-h_2(w_2)\|_\infty=\\
&\|\mathbb{E}_{\mu_{\alpha,\beta}}[\phi(x,a,b)\phi^\top(x,a,b)(w_1-w_2)]\|_\infty\leq\|(w_1-w_2)\|_\infty.
\end{aligned}\tag{19}$$

Next we calculate the derivative of p-norm of term $(w_k - w^*)$, where w^* is the equilibrium point of (17) which verifies $h(w^*)=0$.

$$\begin{aligned}
\frac{d}{dk}\|w_k-w^*\|_p&=\|w_k-w^*\|_p^{1-p}\\
&\cdot\Big(\sum_{i=1}^{d}(w_k(i)-w^*(i))^{p-1}\cdot((h_1(w_k))_i-(h_1(w^*))_i)+\\
&\sum_{i=1}^{d}(w_k(i)-w^*(i))^{p-1}\cdot((h_2(w^*))_i-(h_2(w_k))_i)\Big),
\end{aligned}$$

where we denote by $(h_1(w))_i$ the i^{th} component of $h_1(w)$ and similarly for h_2. Applying Hölder's inequality to the above summations yields

$$\frac{d}{dk}\|w_k-w^*\|_p\leq\|h_1(w_k)-h_1(w^*)\|_p+\|h_2(w^*)-h_2(w_k)\|_p.$$

Taking the limit as $p\to\infty$ and using (18) and (19) leads to

$$\frac{d}{dk}\|w_k-w^*\|_\infty\leq(\gamma-1)\|w_k-w^*\|_\infty.\tag{20}$$

Let $\lambda=1-\gamma>0$. Integrate w.r.t k, (20) becomes

$$\|w_k-w^*\|_\infty\leq e^{-\lambda k}\|w_0-w^*\|_\infty,$$

which establishes the existence of a globally asymptotically stable equilibrium point for (17). And it is clear that $h(w^*)=0$ leads to

$$\begin{aligned}
w^*=\Sigma^{-1}\mathbb{E}_{\mu_{\alpha,\beta}}[\phi(x,a,b)(r(x,a,b,y)+\\
\gamma\min_{\sigma\in\mathcal{B}}\max_{a'\in\{0,1\}}\sum_{b'\in\{0,1\}}\sigma(b')\phi^\top(y,a',b')w^*)].
\end{aligned}\tag{21}$$

Hence, the sequence w_k converges w.p.1 to the globally asymptotically stable equilibrium point w^*.

Then we further prove that the limit of approximate minimax-Q function is the fixed point of projected Bellman operator. Given w^* as (21), the corresponding approximate Q function

$$\begin{aligned}
Q_{w^*}(x,a,b)=&\\
&\phi^\top(x,a,b)\Sigma^{-1}\mathbb{E}_{\mu_{\alpha,\beta}}\left[\phi(x,a,b)(\mathbf{T}Q_{w^*})(x,a,b)\right]\\
&=(\mathbf{PT}Q_{w^*})(x,a,b).
\end{aligned}$$

This implies that Q_{w^*} verifies the fixed point equation in (6).

IV. Numerical Validation

In this section, we implement the approximate minimax-Q (AMQ) learning algorithm and numerically evaluate its performance. The objectives of this section is (i) to present and interpret the cost-aware defending strategy given by the AMQ method and (ii) to study the computational efficiency and approximation accuracy of the AMQ method.

A. Experiment setup

We simulate two system models, one with three parallel servers and one with six; this is intended to study the impact of system complexity. The service rates are listed in Table I:

TABLE I: Experiment parameters.

Parameter	Notation	Value
Arrival rate	λ	5 per unit time
Service rate 1	μ_1	2 per unit time
Service rate 2	μ_2	3 per unit time
Service rate 3	μ_3	4 per unit time
Service rate 4	μ_4	2 per unit time
Service rate 5	μ_5	0.5 per unit time
Service rate 6	μ_6	1 per unit time
Attacking cost	c_1	8 per unit time
Defending cost	c_2	6 per unit time
Discount factor	γ	0.9
Behavior policy constant	C_0	0.6

μ_1–μ_3 are used for the three-server model, while μ_1–μ_6 are used for the six-server model. The table also gives the other parameters. The policies given by (10a)–(10c) are used as the behavior policies. The initial target policies are set to be the random policies $\sigma(0|x)=\sigma(1|x)=0.5$ and $\pi(0|x)=\pi(1|x)=0.5$ for all $x\in\mathbb{Z}_{\geq 0}^m$. The initial traffic state is randomly generated.

We use a neural network Q (NNQ) learning as the benchmark for evaluate the AMQ method. The NNQ methods approximates the value function $Q(x,a,b)$ with a neural network and trains it according to the minimax Bellman equation. Since

NNs have extremely strong approximation performance, we use the NNQ function as a proxy for the ground truth of the equilibrium value, which cannot be analytically obtained. The architecture of the NN comprises two fully connected layers, employing a rectified linear unit (ReLU) as the activation function. The NN is updated via adaptive moment estimation. The loss function used is the mean squared error between the predicted one-step and calculated state-action value.

For the AMQ method, we consider two approximators with different dimensions. The first, named "AMQ1", is a collection of affine functions of the traffic states: for $i = 1, 2, \ldots, m$,

$$\phi_{i,1}(x,a,b) = 1, \quad \phi_{i,2}(x,a,b) = x_i + \delta_i(x,a,b),$$
$$\phi_{i,3}(x,a,b) = a, \quad \phi_{i,4}(x,a,b) = b,$$

where $\delta_i(x,a,b)$ is given by

$$\delta_i(a,b) := \begin{cases} 1 & \text{if } i = \arg\max_i x_i, (a,b) = (1,0), \\ 1 & \text{if } i = \arg\min_i x_i, (a,b) \neq (1,0), \\ 0 & \text{otherwise.} \end{cases}$$

Intuitively, the feature functions are motivated by the reward function in (1). The second, named "AMQ2", is a collection of second-order polynomials of the traffic states: for $i = 1, 2, \ldots, m$,

$$\phi_{i,1}(x,a,b) = 1, \quad \phi_{i,2}(x,a,b) = x_i + \delta_i(x,a,b),$$
$$\phi_{i,3}(x,a,b) = \Big(x_i + \delta_i(x,a,b)\Big)^2,$$
$$\phi_{i,4}(x,a,b) = a, \quad \phi_{i,5}(x,a,b) = b.$$

Hence, AMQ2 is more flexible than AMQ1 and will turn out to be more accurate than AMQ1.

Table II summarizes the three algorithms that we consider. Note that they are all off-policy temporal-difference learning methods.

TABLE II: Algorithms to be compared.

Algorithm	Approximator
NNQ (Baseline)	Two-layer neural network with ReLU
AMQ1 (Ours)	Affine functions of traffic state
AMQ2 (Ours)	Second-order polynomials of traffic state

We trained and evaluated the learning algorithms for 2×10^6 epochs. A discrete time step of 0.1 seconds was employed for simulation. All experiments were conducted using Jupyter Notebook, hosted on a system equipped with an Intel(R) Xeon(R) CPU with 36.7 GB of memory.

B. Interpretation of trained weights

Every experiment that we conducted converged to an approximate equilibrium. As an illustration, consider the three-server setting. The weights for the AMQ2 in this setting turn out to be

$$\begin{array}{lll} w_{1,1} = 6.55, & w_{2,1} = 5.55, & w_{3,1} = 4.55, \\ w_{1,2} = 9.74, & w_{2,2} = 9.23, & w_{3,2} = 9.02, \\ w_{1,3} = 0.46, & w_{2,3} = 0.41, & w_{3,3} = 0.34, \\ w_{1,4} = 0.9, & w_{2,4} = 0.8, & w_{3,4} = 0.8, \\ w_{1,5} = -1.1, & w_{2,5} = -1.0, & w_{3,5} = -0.89. \end{array}$$

Recall that the function $\phi^\top w^*$ is the (approximate) equilibrium cost for the defender; the first index in the subscript is actually the server index.

There are several insights about the weights associated with the same server worth mentioning. First, the first-order terms are associated with weights ($w_{i,3}$) greater than the second-order terms ($w_{i,2}$); this implies that the value function grows roughly linearly with the traffic states. Second, a non-trivial intercept exists ($w_{i,1}$) for every server, which implies that a server might be associated with a risk even if it is idling. Finally, the weights ($w_{i,4}, w_{i,5}$) associated with the player actions have the correct signs. In addition, attacks are associated with smaller weights than defenses, so the defender seems to have a stronger incentive to defend than the attacker to attack; this is probably due to that the defending cost is lower than the attacking cost.

Across various servers, it turns out that queues with lower service rates are in general associated with higher risks, which is intuitive. Interestingly, the greater intercepts ($w_{i,1}$) are consistently associated with higher service rates; that is, an incorrect routing to a slow server, even if it is idling, may still be costly. In addition, the weights ($w_{i,4}, w_{i,5}$) associated with the player actions directly indicates the benefit of attacking/defending a particular server; servers with slower service rates are associated with, without surprise, higher weights.

C. Evaluation of algorithm

Table III presents the normalized learned values and policies with respect to the equilibrium state distribution. The initial state is sampled from this equilibrium distribution, and empirical data is obtained using the Monte Carlo method. The reported results represent the average of 10 repeated experiments. The findings indicate that the learned results of AMQ2 approximate optimal defense strategies with an average error of 2.5%, and approximate the optimal values with an average error of 4.3% under the equilibrium distribution, thus validating the precision of the proposed algorithm in approximating both optimal values and optimal policies. The performance of the AMQ2 algorithm further highlights that the inclusion of quadratic terms in the feature functions improves the empirical average cost by 3.6% and the empirical policy consistency by 3.3%. These results underscore the necessity of incorporating quadratic feature functions to achieve more accurate learning outcomes.

TABLE III: Performance of various methods

Metric	System	AMQ1	AMQ2	NNQ
Normalized mean cost	3-server	1.079	1.043	1.000
Policy consistency	3-server	94.2%	97.5%	100%
Normalized mean cost	6-server	1.082	1.045	1.000
Policy consistency	6-server	94.1%	97.3%	100%

Fig. 2 illustrates the normalized l_2-norm difference between the weights w_t and the optimal weights w^* throughout the learning process for both the three methods. It is evident that the NN method converges after approximately 2.4×10^5

iterations, whereas the AMQ1 and the AMQ2 method achieves convergence after around 5×10^3 iterations. Hence, our proposed algorithm has a much higher convergence rate compared to NN, validating the efficiency of the AMQ learning algorithm.

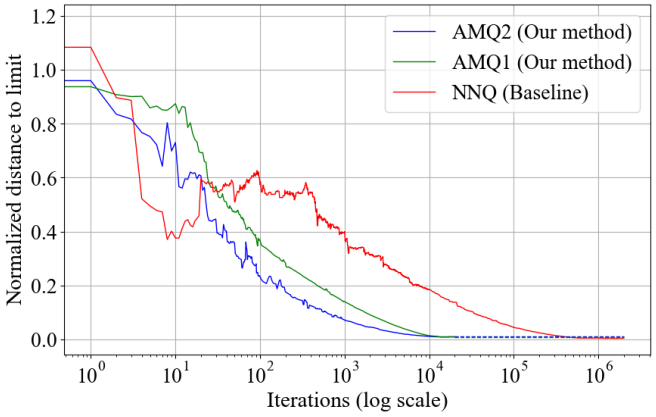

Fig. 2: Performance comparison on distance to limit.

To test the scalability of AMQ, we further implement identical experiments on six servers. The results are also shown in Table III. It can be seen that in six servers setting, the performance of AMQ degrades at most 0.2% in approximating optimal defense strategies and 0.3% in approximating the optimal values, compared to the three servers case. The results indicate that the computational advantage of linear approximation remains at more servers.

V. Concluding Remarks

This paper considers securing parallel server systems against malicious cyber-physical attacks. The proposed approximate minimax-Q (AMQ) learning algorithm efficiently balances security costs and performance losses. The algorithm uses an interpretable linear approximation scheme and adapts to the system's structure. A key advantage of this method compared with deep reinforcement learning methods is a theoretical guarantee for convergence with probability one to an equilibrium under mild assumptions. We established this result by combining the stability theory of Markov processes and an ordinary differential equation-based technique. Tests show the AMQ learning converges faster than neural networks, with an insignificant optimality gap. The approach combines theory and practice, offering scalable security for cloud, manufacturing, and transport systems. It highlights reinforcement learning's potential in adversarial settings with complex, unbounded state spaces. Future work will extend the framework to payoff-based learning algorithms and partially observable environments.

Appendix

Lemma 4 shows that there exist behavior policies satisfying Assumption 3.

Lemma 4. *Under the policy pair (10a)-(10c). Suppose that assumption 1 holds. Let* $V(x) = \sum_{n=1}^m e^{vx_n}$, $v > 0$. *Then there exist some* $c > 0, d < \infty$ *such that*

$$\mathcal{L}_{\alpha,\beta}V(x) = \sum_{y\in\mathbb{Z}^m_{\geq 0}} q_{\alpha,\beta}(y|x)V(y) - V(x) \leq -cV(x) + d, \quad x \in \mathbb{Z}^m_{\geq 0}, \tag{22}$$

where $\mathcal{L}_{\alpha,\beta}$ *is the infinitesimal generator under policy pair* α, β, $q_{\alpha,\beta}(y|x)$ *are the transition rates from state* x *to* y *defined in (2).*

Proof. Denote the longest queue as $x_{\max}$ and its corresponding index as i. Define similarly the shortest queue $x_{\min}$ and its index j. Let $l_n = \mathbb{I}_{\{x_n \geq 1\}}$.We have

$$\begin{aligned}\mathcal{L}_{\alpha,\beta}V(x) =& \sum_{n=1}^m l_n\mu_n(e^{v(x_n-1)} - e^{vx_n}) \\ &+\lambda\Big(C_0e^{-|x|_1}\cdot(e^{v(x_{\max}+1)} - e^{x_{\max}}) \\ &+(1 - C_0e^{-|x|_1})\cdot(e^{v(x_{\min}+1)} - e^{x_{\min}})\Big).\end{aligned}$$

Note that $l_n \cdot e^{vx_n} = (l_n - 1) + e^{vx_n}$, so we have

$$\begin{aligned}\mathcal{L}_{\alpha,\beta}V(x) =& \sum_{\substack{n=1\\n\neq i,j}}^m \mu_n e^{vx_n}(e^{-v} - 1) + B_0 \\ &+\Big(\mu_i(e^{-v} - 1) + \lambda(e^v - 1)C_0e^{-|x|_1}\Big)e^{vx_{\max}} \\ &+\Big(\mu_j(e^{-v} - 1) + \lambda(e^v - 1)(1 - C_0e^{-|x|_1})\Big)e^{vx_{\min}}.\end{aligned} \tag{23}$$

where $B_0 = \sum_{n=1}^m (l_n - 1)\mu_n(e^{-v} - 1)$ is a finite non-negative constant. It can be deduced that the drift equation (23)

$$\mathcal{L}_{\alpha,\beta}V(x) < \sum_{\substack{n=1\\n\neq i,j}}^m e^{vx_n}\cdot f_n(v) + B_0 + B_1,$$

where

$$\begin{aligned}f_n(v) :=& \mu_n(e^{-v} - 1) \\ &+ \frac{\mu_n}{\sum_{k\neq i,j}^m \mu_k}\Big(\mu_i(e^{-v} - 1) + \lambda(e^v - 1)C_0\Big) \\ &+ \frac{\mu_n}{\sum_{k\neq i,j}^m \mu_k}\Big(\mu_j(e^{-v} - 1) + \lambda(e^v - 1)\Big), \qquad \forall n,\end{aligned}$$

$$\begin{aligned}B_1 =& \sum_{n\neq i,j}^m \frac{\mu_n}{\sum_{k\neq i,j}^m \mu_k}\Big(\mu_i(e^{-v} - 1) + \\ &\lambda C_0(e^v - 1)e^{-|x|_1}\Big)(e^{vx_{\max}} - e^{vx_n}) \\ \leq& \sum_{n\neq i,j}^m \frac{\mu_n}{\sum_{k\neq i,j}^m \mu_k}\Big(\mu_i(e^{-v} - 1)(e^{vx_{\max}} - e^{vx_n}) + \\ &(\sum_{k=1}^m \mu_k - \lambda)(e^v - 1)(e^{vx_{\max}-|x|_1} - e^{vx_n-|x|_1})\Big).\end{aligned}$$

Note that B_1 is finite as long as $v \leq 1$. Note that f_n is continuous and $f_n(0) = 0$, $f_n(\infty) = \infty$. The derivative of $f(v)$ at $v = 0$ is calculated as

$$\frac{df_n}{dv}\Big|_{v=0} = \mu_n\Big(\frac{\lambda(1 + C_0) - \mu_i - \mu_j}{\sum_{k\neq i,j}^m \mu_k} - 1\Big) < 0.$$

Then the fact that derivative of $f_n(v)$ is negative at 0 implies that there exist $v_0 > 0$ as the second zero of $f_n(v)$ such that $f_n(v) < 0, v \in (0, v_0)$. Hence, we can guarantee (22) with a proper selection of $v^* \in (0, \min\{v_0, 1\})$. The corresponding $c = -\max_n f_n(v^*)$, $d = B_0 + B_1$ by [49, Theorem 7.1]. □

ACKNOWLEDGMENTS

The authors appreciate the inputs from Qian Xie, Yidan Wu, Yule Zhang, and other members of the Smart & Connected Systems Lab at Shanghai Jiao Tong University.

REFERENCES

[1] A. Laszka, W. Abbas, Y. Vorobeychik, and X. Koutsoukos, "Detection and mitigation of attacks on transportation networks as a multi-stage security game," *Computers & Security*, vol. 87, p. 101576, 2019.

[2] F. Fraile, T. Tagawa, R. Poler, and A. Ortiz, "Trustworthy industrial IoT gateways for interoperability platforms and ecosystems," *IEEE Internet of Things Journal*, vol. 5, no. 6, pp. 4506–4514, 2018.

[3] L. Jin and S. Amin, "Stability of Fluid Queueing Systems with Parallel Servers and Stochastic Capacities," *IEEE Transactions on Automatic Control*, vol. 63, no. 11, pp. 3948–3955, 2018.

[4] S. Hu, D. Yue, X. Xie, X. Chen, and X. Yin, "Resilient event-triggered controller synthesis of networked control systems under periodic dos jamming attacks," *IEEE Transactions on Cybernetics*, vol. 49, no. 12, pp. 4271–4281, 2018.

[5] W. Liu, J. Sun, G. Wang, F. Bullo, and J. Chen, "Resilient control under quantization and denial-of-service: Codesigning a deadbeat controller and transmission protocol," *IEEE Transactions on Automatic Control*, vol. 67, no. 8, pp. 3879–3891, 2021.

[6] W. Lucia, G. Franzè, and B. Sinopoli, "A supervisor-based control architecture for constrained cyber-physical systems subject to network attacks," *IEEE Transactions on Control of Network Systems*, vol. 10, no. 3, pp. 1184–1194, 2022.

[7] Z. Cao, Z. Wang, Y. Niu, J. Song, and H. Liu, "Sliding mode control for sampled-data systems subject to deception attacks: Handling randomly perturbed sampling periods," *IEEE Transactions on Cybernetics*, vol. 53, no. 11, pp. 7034–7047, 2022.

[8] Y. Feng, S. E. Huang, W. Wong, Q. A. Chen, Z. M. Mao, and H. X. Liu, "On the cybersecurity of traffic signal control system with connected vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 9, pp. 16 267–16 279, 2022.

[9] M. M. Aslam, A. Tufail, R. A. A. H. M. Apong, L. C. De Silva, and M. T. Raza, "Scrutinizing security in industrial control systems: An architectural vulnerabilities and communication network perspective," *IEEE Access*, 2024.

[10] S. Achleitner, T. La Porta, P. McDaniel, S. Sugrim, S. V. Krishnamurthy, and R. Chadha, "Cyber deception: Virtual networks to defend insider reconnaissance," in *Proceedings of the 8th ACM CCS International Workshop on Managing Insider Security Threats*, 2016, pp. 57–68.

[11] Amazon 'thwarts largest ever ddos cyber-attack'. [Online]. Available: https://www.bbc.com/news/technology-53093611

[12] Malware attack stifles philadelphia area transit agency. [Online]. Available: https://www.govtech.com/public-safety/malware-attack-stifles-philadelphia-area-transit-agency.html

[13] Bobst a réussi à déjouer deux cyberattaques ciblées. [Online]. Available: https://www.letemps.ch/cyber/bobst-reussi-dejouer-deux-cyberattaques-cibleessrsltid=AfmBOopxzZSAPmdAxu6hllhDHxyLItiCvvRCZnnF4ybwei7c5EXlkdos

[14] P. C. van Oorschot and S. W. Smith, "The Internet of things: Security challenges," *IEEE Security & Privacy*, vol. 17, no. 5, pp. 7–9, 2019.

[15] W. Tushar, C. Yuen, T. K. Saha, S. Nizami, M. R. Alam, D. B. Smith, and H. V. Poor, "A survey of cyber-physical systems from a game-theoretic perspective," *IEEE Access*, vol. 11, pp. 9799–9834, 2023.

[16] Q. Xie, J. Wang, and L. Jin, "Cost-aware defense for parallel server systems against reliability and security failures," *Automatica*, vol. 160, p. 111467, 2024.

[17] X. Liang and Y. Xiao, "Game theory for network security," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 1, pp. 472–486, 2012.

[18] M. H. R. Khouzani, E. Altman, and S. Sarkar, "Optimal quarantining of wireless malware through reception gain control," *IEEE Transactions on Automatic Control*, vol. 57, no. 1, pp. 49–61, 2011.

[19] M. Wu and S. Amin, "Securing infrastructure facilities: When does proactive defense help?" *Dynamic Games and Applications*, vol. 9, pp. 984–1025, 2019.

[20] F. Miao, Q. Zhu, M. Pajic, and G. J. Pappas, "A hybrid stochastic game for secure control of cyber-physical systems," *Automatica*, vol. 93, pp. 55–63, 2018.

[21] B. Lian, W. Xue, F. L. Lewis, and T. Chai, "Inverse reinforcement learning for adversarial apprentice games," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 34, no. 8, pp. 4596–4609, 2021.

[22] Y. Xu, Z.-G. Wu, and Y.-J. Pan, "Cooperative path following control in autonomous vehicles graphical games: A data-based off-policy learning approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 8, pp. 9364–9374, 2024.

[23] Q. Xie, Y. Chen, Z. Wang, and Z. Yang, "Learning zero-sum simultaneous-move markov games using function approximation and correlated equilibrium," in *Conference on learning theory*. PMLR, 2020, pp. 3674–3682.

[24] Z. Zhou, P. Mertikopoulos, A. L. Moustakas, N. Bambos, and P. Glynn, "Robust power management via learning and game design," *Operations Research*, vol. 69, no. 1, pp. 331–345, 2021.

[25] A. K. Bozkurt, Y. Wang, M. M. Zavlanos, and M. Pajic, "Learning optimal strategies for temporal tasks in stochastic games," *IEEE Transactions on Automatic Control*, 2024.

[26] L. Buşoniu, D. Ernst, B. De Schutter, and R. Babuška, "Approximate reinforcement learning: An overview," in *2011 IEEE Symposium on Adaptive Dynamic Programming and Reinforcement Learning (ADPRL)*. IEEE, 2011, pp. 1–8.

[27] Z. Ma, Q. Zhang, and Z. Wang, "Safe and stable secondary voltage control of microgrids based on explicit neural networks," *IEEE Transactions on Smart Grid*, vol. 14, no. 5, pp. 3375–3387, 2023.

[28] T. Prieto-Rumeau and J. M. Lorenzo, "Approximation of zero-sum continuous-time markov games under the discounted payoff criterion," *Top*, vol. 23, no. 3, pp. 799–836, 2015.

[29] D. Shah, Q. Xie, and Z. Xu, "Stable reinforcement learning with unbounded state space," in *Learning for Dynamics and Control*. PMLR, 2020, pp. 581–581.

[30] D. Ding, C.-Y. Wei, K. Zhang, and M. Jovanovic, "Independent policy gradient for large-scale markov potential games: Sharper rates, function approximation, and game-agnostic convergence," in *International Conference on Machine Learning*. PMLR, 2022, pp. 5166–5220.

[31] Z. Chen, S. Zhang, T. T. Doan, J.-P. Clarke, and S. T. Maguluri, "Finite-sample analysis of nonlinear stochastic approximation with applications in reinforcement learning," *Automatica*, vol. 146, p. 110623, 2022.

[32] S. Zhang, H. Yao, and S. Whiteson, "Breaking the deadly triad with a target network," in *International Conference on Machine Learning*. PMLR, 2021, pp. 12 621–12 631.

[33] D. Zhou, J. He, and Q. Gu, "Provably efficient reinforcement learning for discounted mdps with feature mapping," in *International Conference on Machine Learning*. PMLR, 2021, pp. 12 793–12 802.

[34] F. S. Melo and M. I. Ribeiro, "Convergence of q-learning with linear function approximation," in *2007 European Control Conference (ECC)*. IEEE, 2007, pp. 2671–2678.

[35] M. Sayin, K. Zhang, D. Leslie, T. Basar, and A. Ozdaglar, "Decentralized q-learning in zero-sum markov games," *Advances in Neural Information Processing Systems*, vol. 34, pp. 18 320–18 334, 2021.

[36] A. Ozdaglar, M. O. Sayin, and K. Zhang, "Independent learning in stochastic games," in *International Congress of Mathematicians*, 2021.

[37] M. L. Littman, "Markov games as a framework for multi-agent reinforcement learning," in *Machine Learning Proceedings 1994*. Elsevier, 1994, pp. 157–163.

[38] C. Szepesvári and M. L. Littman, "A unified analysis of value-function-based reinforcement-learning algorithms," *Neural Computation*, vol. 11, no. 8, pp. 2017–2060, 1999.

[39] J. Hu and M. P. Wellman, "Nash Q-learning for general-sum stochastic games," *Journal of Machine Learning Research*, vol. 4, no. Nov, pp. 1039–1069, 2003.

[40] R. Lowe, Y. I. Wu, A. Tamar, J. Harb, O. Pieter Abbeel, and I. Mordatch, "Multi-agent actor-critic for mixed cooperative-competitive environments," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[41] Z. Chen, K. Zhang, E. Mazumdar, A. Ozdaglar, and A. Wierman, "Two-timescale q-learning with function approximation in zero-sum stochastic games," *arXiv preprint arXiv:2312.04905*, 2023.

[42] X. Guo and O. Hernández-Lerma, "Nonzero-sum games for continuous-time markov chains with unbounded discounted payoffs," *Journal of Applied Probability*, vol. 42, no. 2, pp. 303–320, 2005.

[43] P. Kumar and S. P. Meyn, "Stability of queueing networks and scheduling policies," *IEEE Transactions on Automatic Control*, vol. 40, no. 2, pp. 251–260, 1995.
[44] Q. Xie and L. Jin, "Stabilizing queuing networks with model data-independent control," *IEEE Transactions on Control of Network Systems*, vol. 9, no. 3, pp. 1317–1326, 2022.
[45] V. Borkar and S. Meyn, "Stability and convergence of stochastic approximation using the ode method," in *Proceedings of the 37th IEEE Conference on Decision and Control (Cat. No. 98CH36171)*, vol. 1. IEEE, 1998, pp. 277–282.
[46] S. Chen, A. Devraj, A. Berstein, and S. Meyn, "Revisiting the ode method for recursive algorithms: Fast convergence using quasi stochastic approximation," *Journal of Systems Science and Complexity*, vol. 34, pp. 1681–1702, 2021.
[47] A. Benveniste, M. Métivier, and P. Priouret, *Adaptive Algorithms and Stochastic Approximations*. Springer Science & Business Media, 2012, vol. 22.
[48] S. D. Liu, S. Chen, and S. Zhang, "The ODE method for stochastic approximation and reinforcement learning with Markovian noise," *Journal of Machine Learning Research*, vol. 26, no. 24, pp. 1–76, 2025.
[49] S. P. Meyn and R. L. Tweedie, "Stability of Markovian processes III: Foster–Lyapunov criteria for continuous-time processes," *Advances in Applied Probability*, vol. 25, no. 3, pp. 518–548, 1993.
[50] R. Singh and P. Kumar, "Throughput optimal decentralized scheduling of multihop networks with end-to-end deadline constraints: Unreliable links," *IEEE Transactions on Automatic Control*, vol. 64, no. 1, pp. 127–142, 2018.
[51] L. S. Shapley, "Stochastic Games," *Proceedings of the National Academy of Sciences*, vol. 39, no. 10, pp. 1095–1100, 1953.
[52] Y. Zhu and D. Zhao, "Online minimax Q network learning for two-player zero-sum Markov games," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 3, pp. 1228–1241, 2020.
[53] A. Tanwani, B. Brogliato, and C. Prieur, "Stability notions for a class of nonlinear systems with measure controls," *Mathematics of Control, Signals, and Systems*, vol. 27, pp. 245–275, 2015.
[54] N. J. van Eck and M. van Wezel, "Application of reinforcement learning to the game of Othello," *Computers & Operations Research*, vol. 35, no. 6, pp. 1999–2017, 2008.

Yuzhen Zhan is a master student (2023–present) at the UM Joint Institute, Shanghai Jiao Tong University, China. She received her B.Eng. degree in Automation from Wuhan University, China in 2023. Her research focuses on game-theoretical model of adversarial dynamic and applying reinforcement learning methods to solve practical security challenges in cyber-physical systems. In particular, she is interested in theoretical guarantees of algorithms.

Li Jin is John Wu & Jane Sun Associate Professor (2022–present) and was Assistant Professor (2021-2022) of Electrical and Computer Engineering at the UM Joint Institute and the Department of Automation, Shanghai Jiao Tong University (SJTU), China. He was Assistant Professor (2018–2020) at the Tandon School of Engineering, New York University, USA. He received his B.Eng. from SJTU in 2011, M.S. from Purdue University, USA in 2012, and Ph.D. from the Massachusetts Institute of Technology, USA in 2018. He was also a Visiting Scholar at the University of Erlangen-Nuremberg, Germany in 2016. He is the recipient of multiple research grants/awards from the US National Science Foundation and the National Natural Science Foundation of China on topics including connected and autonomous vehicles, network system control, and cyber-physical security.