Non-Equilibrium Learning and Cyber-Physical Security

Kyriakos G. Vamvoudakis, Senior Member, IEEE, Aris Kanellopoulos, Student Member, IEEE

Abstract—This paper introduces a framework for non-equilibrium behavior analysis in cyber-physical systems for security purposes. To categorize the player, we employ the principles of reinforcement learning in order to derive an iterative method of optimal responses that determine the policy of an agent with level-\(k\) intelligence in a general non-zero-sum, nonlinear environment. For the special case of zero-sum, linear quadratic games we derive appropriate non-equilibrium game Riccati equations. To obviate the need for complete knowledge of the system dynamics, we employ a Q-Learning algorithm as a best response solver. We then design an estimator that determines the distribution of intelligence levels in the adversarial environment of the system. Finally, simulation results showcase the efficacy of our approach.

I. INTRODUCTION

Cyber-physical systems (CPS) are complex, large-scale platforms in which the computational and physical components are tightly inter-connected. The increasing penetration of CPS in society can be readily seen by the plethora of applications they have found, from healthcare and medicine [1], to smart grids [2], transportation and autonomy [3].

The complexity that allows CPS to find use in such a variety of fields, has also increased the attack surface of those systems, that have become the target of malicious entities, able to exploit numerous vulnerabilities in the system. Successful integration of CPS, requires the thorough investigation of those security concerns. An important step towards this direction is in developing models for predicting human behavior, which will determine the decisions policies acting on a CPS. Well-known prediction models follow the concept of Nash equilibrium [4], wherein all the competitive agents involved, are assumed to share the same decision making mechanism as well as informed that are participating in a game. However, in many games where human players are involved, equilibrium models are unable to explain realistic behaviors. An alternative explanation is that agents are boundedly rational, i.e. they employ shifting, imperfect thinking methods based on preexisting knowledge. Decision makers may never play the exact same game very often, but they may also extrapolate between games and learn from experiences. Several recent experimental studies [5], [6] suggest that decision makers’ initial responses to games often deviate systematically from equilibrium, and that structural non-equilibrium models (e.g., cognitive hierarchy) often outpredict equilibrium. Thus, non-equilibrium models need to allow for players whose adjustment rules are not a best response to the adjustment rules of the others.

Related Work

Although the most important aspects of security still reside in the computational level, it has been argued in [7] that defense options focused on software are often not enough. Towards this, various models for games in security, both static and dynamic, are introduced in [8], which led to the introduction of control theoretic solutions to the problem. Mathematically, control theory and game theory can be brought together through the Hamilton-Jacobi equations [9], [10]. But such approaches follow the equilibrium approach to provide a solution.

Realistic, non-equilibrium solution concepts for static game theory has been reported in [11]–[14]. The authors of those works imply that, even if the players may not act according to the infinite rationality assumption, they do not follow naive stimulus-response models, either. Thus, agents are expected to try to improve their framework, albeit imperfectly. Reinforcement learning algorithms were presented in [15], [16] for simple experimental games, which have since led to the formulation of behavioral game theory. The authors in [17] overcame the irrationality issue with a Win-or-Learn Fast in a mini-max-Q learning framework, but with the penalty of loosing convergence assurance. Quantal response models [18] retain their infinite knowledge about other players, but they assume small stochastic perturbations from the equilibrium in the actual policies played.

The class of non-equilibrium models that have inspired this work, retain the deterministic optimality of the policies, but they assume finite best response iterations that may not converge to the equilibrium. Those iterations are conditioned based on the belief of each player regarding the actions observed by their competitors. In level-\(k\) thinking it is assumed that each player believes all of her opponents operate at the \(k\)th level of intelligence. On the other hand, in cognitive hierarchy models, each player believes that her opponents’ levels follow a Poisson distribution.

Contribution

The contributions of the present work are fourfold. We formulate a CPS operating in an adversarial environment by considering a general non-zero-sum, nonlinear differential game/ After deriving the expressions for the level-\(k\) best response for this case, we investigate the simpler problem of zero-sum linear quadratic differential games. Third, by utilizing a Q-Learning algorithm we show how to train the
different levels of intelligence. Finally, we present an estimation mechanism that is able to learn the mean distribution of intelligence.

Structure: The remainder of the paper is structured as follows. Section II, formulates the non-zero-sum, nonlinear non-equilibrium problem and presents conditions both for the infinitely rational, and the boundedly rational solution. For the simplified zero-sum linear quadratic problem, the different levels of rationality are shown to correspond to Riccati equations in Section III. Specifically, this section proposes a framework to compute the policies corresponding to levels of intelligence and presents a framework that allows a defender to learn the true distribution of intelligent agents in the adversarial environment. Section IV shows the efficacy of the proposed framework through simulation results. Finally, Section V concludes and talks about future work.

Notation: The notation used here is standard. \( \lambda(A) \) is the maximum eigenvalue of the matrix \( A \) and \( \Delta(A) \) is its minimum eigenvalue. \( \| \cdot \| \) denotes the Euclidean norm of a vector and the Frobenius norm of a matrix. The superscript \( * \) is used to denote the optimal trajectories of a variable. \( \nabla \) and \( \frac{\partial}{\partial x} \) are used interchangeably and denote the partial derivative with respect to a vector \( x \). The Kronecker product between two vectors is denoted by \( x \otimes y \) and the half-vectorization of a matrix \( A \), is denoted by \( \text{vech}(A) \).

II. NON-EQUILIBRIUM NON-ZERO-SUM DIFFERENTIAL GAME

Initially, we formulate the problem in the general case for non-equilibrium non-zero-sum differential games with \( N \) players. We denote by \( N \) the set of all players, and by \( N_{-i} = N \setminus \{i\} \) the set containing all players except player \( i \). The environment evolves according to the following dynamics,

\[
\dot{x} = f(x) + \sum_{j \in N} g_j(x)u_j, \quad x(0) = x_0, \quad t \geq 0, \tag{1}
\]

where \( x(t) \in \mathbb{R}^n \) is the state vector which is available for feedback to all the players, \( u_i \in \mathbb{R}^m, \forall i \in N \) the decision vectors, \( f(x) : \mathbb{R}^n \to \mathbb{R}^n \) the drift dynamics, and \( g_i(x) : \mathbb{R}^n \to \mathbb{R}^m, \forall i \in N \) the input dynamics.

Each player aims to minimize a cost functional of the form,

\[
J_i(x(0); u_1, \ldots, u_N) = \int_0^\infty r_i(x(t), u_1, \ldots, u_N)dt
\]

where \( r_i(x(t), u_1, \ldots, u_N) \) is the cost function for player \( i \).

The derivation of the Nash policies \( u_i^*, i \in \{1, \ldots, N\} \), corresponds to solving the coupled minimization problems,

\[
J_i(x(0); u_i^*, u_{-i}^*) = \min_{u_i} J_i(x(0); u_i, u_{-i}^*). \tag{2}
\]

Towards that, we need to evaluate the optimal value functions \( V_i(x), i \in \{1, \ldots, N\} \) such that,

\[
V_i(x) = \min_{u_i} \int_0^\infty r_i(x, u_i, u_{-i}^*)dt.
\]

Initially, we define the Hamiltonian of each player as,

\[
H_i(x, u_i, u_{-i}) = r_i(x(t), u_1, \ldots, u_N) + \nabla V_i^T(f(x) + \sum_{j \in N} g_j(x)u_j). \tag{3}
\]

To compute the Nash policy for every player \( i \in \{1, \ldots, N\} \), we apply the stationarity conditions to (3), i.e. \( \frac{\partial H}{\partial u_i} = 0 \), to get,

\[
u_i^*(x) = -R_{i}^{-1}g_i^T(x)\frac{\partial V_i^*}{\partial x}. \tag{4}
\]

Substituting (4) into (3), yields the following coupled Hamilton-Jacobi system of \( N \) equations,

\[
Q_i(x) + \frac{1}{4} \sum_{j \in N} \nabla V_j^T g_j(x)R_{jj}^{-1}R_{ij}g_j^T(x) \nabla V_j + (\nabla V_i)^T(f(x) - \frac{1}{2} \sum_{j \in N} g_j(x)R_{jj}^{-1}g_j^T(x) \nabla V_j) = 0. \tag{5}
\]

In this work, we shall follow the approach of “steps of reasoning,” where each player belongs to a specific “cognitive type.”

Due to computational and cognitive limitations, the emergence of Nash solutions in experimental games is a rarity. In this work, we are interested in introducing opponents with bounded rationality in differential games for CPS security. Initially, we will define strategies of “level-\( k \)” thinkers as different steps of strategic thinking where a distribution of beliefs of the rest of the players is considered known. Subsequently, by limiting our analysis to two-player zero-sum games, we will introduce a fitting mechanism that enables estimation of the levels of rationality in the environment.

Consider a player \( i \in N \) belonging to a cognitive \( k \in \mathbb{Z}^+ \). This player is able to perform \( k \)-steps of iterative strategic thinking while her beliefs about the behavior of the rest of the players are constructed based on a fixed, full-support distribution, e.g. – according to the work of [19] – a Poisson distribution.

Level-0 policies: Bounded rationality approaches to non-equilibrium games that depend on iterative best responses assume that all the players have prior knowledge of the strategies of those that do not conduct strategic thinking, i.e. level-0 policies \( u_i^{0}, i \in N \). Previous work on non-equilibrium game theory [20], proposes two different definitions of anchor policies; random policies and intuitive responses to the game.

Level-\( k \) policies: For a level-\( k \) player to compute their higher order policies, they follow a sequence of \( k \) thinking steps, which will be modeled through a Policy Iteration scheme. The level-\( k \) player initializes her thinking process by assuming that every other player uses their level-0 policies. Subsequently, she computes the best response to those policies, which constitutes her level-1 policy. At each subsequent level, the player computes the best response according to the expected behavior of the other players. For example, a level-3 player best responds to the probability that each of the other
players follows a level $l < 3$ policy, where those probabilities are sampled from a predefined distribution.

The single agent optimization that a level-$k$ player performs corresponds to solving the following Hamilton-Jacobi-Bellman (HJB) equation with respect to a non-equilibrium value function $V^k(x) : \mathbb{R}^n \to \mathbb{R}$, which is conditioned by the belief probabilities of the agent of level-$k$, regarding the intelligence level of player $h \in \mathbb{N} \setminus \{i\}$, denoted as $b^k(h)$,

$$ Q(x) + u^k_i R_{ix} u^k_i + \sum_{j \in \mathbb{N} \setminus \{i\}} \sum_{h=0}^{k-1} (b^k(h)u_j) \mathbb{H} R_{ij} b^k(h) u^k_j + (\nabla V^k_i)^T (f(x) + g(x)u^k_i + \sum_{j \in \mathbb{N} \setminus \{i\}} \sum_{h=0}^{k-1} (g(x)b^k(h)u_j)) = 0, $$

which yields the level-$k$ policy given by,

$$ u^k_i(x) = -R_{ii}^{-1} g_i^T V_i^k, \forall x. $$

### III. Linear Quadratic Zero-sum Boundedly Rational Games

In this section, we present an implementation of the non-equilibrium framework to zero-sum games of a defending and an attacking agent competing over a CPS modeled as a linear system. Furthermore, we consider that the belief probabilities over the lower intelligence levels are Dirac distributions on level-$k-1$, i.e., every level-$k$ agent assumes their opponent is a level-$k-1$ player. Thus, the dynamics evolve according to the simplified version of (1) given by,

$$ \dot{x}(t) = Fx(t) + Gu(t) + Dd(t), \quad x(0) = x_0, $$

where $u(t)$ is the input of the defender and $d(t)$ the input of the adversary.

Agents of infinite rationality seek to compute the saddle point solution to the simplified version of (2),

$$ J(x(0);u,d) = \frac{1}{2} \int_0^\infty \left( x^T Mx + u^T R u - \gamma^2 ||d||^2 \right) dt, $$

where the user-defined matrices satisfy $M \succeq 0$, $R > 0$ and $\gamma > \gamma^*$, with $\gamma^*$ the attenuation level that renders the integral finite.

**Assumption 1.** The pair $(F,G)$ is controllable and the pair $(F,\sqrt{M})$ is detectable.

For the zero-sum game, the Nash equilibrium is equivalent to the following minimax optimization problem,

$$ V^*(x) = \min_u \max_d \int_0^\infty \left( x^T Mx + u^T R u - \gamma^2 ||d||^2 \right) dt. $$

**Definition 1.** The pair of policies $(u^*,d^*)$ are the ones that satisfy, $J(u^*,d^*) \leq J(u^*,d^*) \leq J(u,d^*), \forall u,d.$

The coupled Hamilton-Jacobi equations presented in (5) are known [9] to correspond to the following Riccati equation for linear quadratic zero-sum games,

$$ FT + PF - PGR^{-1}GT + \frac{1}{\gamma^2} PDD^T P + M = 0. $$

Under Assumption 1, the Riccati (7) has a unique solution which is described by the saddle point of the zero-sum game.

**Cognitive Hierarchy**

a) **Levels of Rationality:** In this section, we will present the application of the proposed non-equilibrium framework in zero-sum games, and the resulting Riccati equations.

**Level-0 (Anchor) Policy:** By assuming intuitive responses as anchor policies, we consider the defended as controlling the system while being agnostic to attacks. Consequently, the level-0 input is found by solving the optimal control problem described by,

$$ V_u^0(x_0) = \min_u \int_0^\infty (x^T M x + u^T R u) dt, \forall x. $$

The optimal control input for the optimization problem (8) given (6) with $d_i = 0$ is,

$$ u^0(x) = -R^{-1} G^T \frac{\partial V_u^0(x)}{\partial x} = -R^{-1} B^T P^0_u x, \forall x, $$

where the value function is taken as quadratic in the states $V_u^0(x) = x^T P^0_u x$ and the kernel $P^0_u$ solves the following Riccati equation $F^T P^0_u + P^0_u F + M - P^0_u G R^{-1} G^T P^0_u = 0$.

Subsequently, the intuitive response of a level-0 adversary is an optimal attack under the belief that the defender is unaware of her existence. To this end, we define the optimization problem from the point of view of the adversary for the anchor defender input $u = u^0(x)$,

$$ V_u^0(x_0) = \max_d \int_0^\infty (x^T M x + u^0 R u^0 - \gamma^2 ||d||^2) dt, $$

subject to the system dynamics, $\dot{x} = (F - GR^{-1} GT P^0_u)x + D d$.

The level-0 attack input is computed as, $d^0(x) = \frac{1}{\sqrt{\gamma}} D^T P^0_d x$, where the matrix $P^0_d$ solves the following Riccati equation,

$$ (F - GR^{-1} GT P^0_u)^T P^0_d + P^0_d (F - GR^{-1} GT P^0_u) $$

$$ + (M - P^0_u GR^{-1} GT P^0_u) \frac{1}{\gamma^2} P^0_d D D^T P^0_d = 0. $$

**Level-k Policies:** To derive the policies for the agents of higher levels of rationality, we apply the proposed iterative process. A defender performing an arbitrary number of $k$ strategic thinking interactions solves the following minimization problem given (6),

$$ V^k_u(x_0) = \min_u \int_0^\infty (x^T M x + u^0 R u - \gamma^2 ||d||^2) dt. $$

The Hamiltonian associated with (9) and (6) is,

$$ H^k_u(x,u,d^{k-1}) = x^T M x + u^T R u - \gamma^2 ||d||^2 $$

$$ + \left( \frac{\partial V^k_u}{\partial x} \right)^T (F x + G u + D d^{k-1}). $$

Substituting the adversarial input with the policy of the previous level $d^{k-1} = \frac{1}{\gamma^2} D^T P^{k-1}_d x$, starting from level-0, yields,

$$ u^k(x) = -R^{-1} G^T P^k u, $$

where the level-$k$ defender Riccati matrix is the solution of,

$$ (F + \frac{1}{\gamma^2} P^{k-1}_d D D^T P^{k-1}_d)^T P^k $$
Similarly, the adversary of an arbitrary $k$ level of thinking, maximizes her response to the input of a defender of level-$k$,

$$V_k^d(x_0) = \max_d \int_0^\infty (x^T M x + u^k T R u^k - \gamma^2 \|q\|^2) d\tau.$$  \hspace{1cm} (12)

The corresponding Hamiltonian is,

$$H^d_k(x, u^k, d) = x^T M x + u^k T R u^k - \gamma^2 \|q\|^2 + \frac{\partial V_k^d}{\partial x^T} (F x + G u^k + D d).$$ \hspace{1cm} (13)

Substituting (11) in (13) yields the following attack response, $d^k(x) = \frac{1}{\gamma^2} D T P_d^k x$, where the matrix $P_d^k$ is the solution of,

\begin{align*}
(F - G R^\top G T P_u^k)^T P_d^k + P_d^k (F - G R^\top G T P_u^k) \\
+ (M - P_d^k G R^\top G T P_u^k) + \frac{1}{\gamma^2} P_d^k D T P_d^k = 0.
\end{align*}

With this iterative procedure, a defending agent is able to compute the strategies of the adversaries with finite cognitive abilities, for a given number of levels.

### A. Q-Learning for Interaction with Level-$k$ Attacks

Since computing the best responses of the competitive players is of increased difficulty even in the simpler zero-sum problem, we employ the model-free Q-learning methodology developed in [21].

Consider the problem solved by an agent, described by (9) for the defender and (12) for the adversary. For ease of exposition, we shall introduce the notation $V_j^q(\cdot)$, $a_j^q(x)$, and $H_j^q(x, a_j^q, \nabla V_j^q)$ to mean the value function, action policy and Hamiltonian of a level-$k$ agent, where $j \in \{u, d\}$, i.e., the defender and the adversary.

The Q-function can be defined to be,

$$Q_j^k(x, a_j^k) = V_j^k(x) + H_j^k(x, a_j^k, \nabla V_j^k).$$ \hspace{1cm} (14)

Substituting the Hamiltonians (10) for the defender and (13) for the adversary in (14), we can rewrite the Q-function (14) in a compact quadratic in the state and action form as,

$$Q_j^k(x, a_j^k) = U_j^k \begin{bmatrix} Q_{j,xx}^k & Q_{j,xa}^k \\ Q_{j,ax}^k & Q_{j,aa}^k \end{bmatrix} U_j^\top,$$

where for the level-$k$ defender’s problem, $j = u$, we have that $U_u^k = [x^T \ u^k]^\top$,

$$Q_u^k = (F + \frac{1}{\gamma^2} D T P_d^k) P_u^k + (M + \frac{1}{\gamma^2} D T P_d^k) - P_u^k G R^\top G T P_u^k,$$

$$Q_{u,ax}^k = G^\top P_u^k, \ Q_{u,aa}^k = G P_u^k, \ Q_{u,aa}^k = R.$$

Accordingly, for the level-$k$ adversarial problem, $j := d$, we have that $U_d^k = [x^T \ d^k]^\top$,

$$Q_d^k = (F - G R^\top G T P_u^k)^T P_d^k + P_d^k (F - G R^\top G T P_u^k) + (M - P_d^k G R^\top G T P_u^k) + \frac{1}{\gamma^2} P_d^k D T P_d^k,$$

$$Q_{d,ax}^k = D^T P_d^k, \ Q_{d,aa}^k = D P_d^k, \ Q_{d,aa}^k = -\gamma^2.$$

The action for each agent at each level can be found by solving $\frac{\partial Q_j^k(x, a_j^k)}{\partial a_j^k} = 0$ as,

$$a_j^k(x) = -(Q_{j,aa}^k)^{-1} Q_{j,ax}^k.$$ \hspace{1cm} (16)

We will now use an actor/critic structure to tune the parameters online by utilizing an integral reinforcement learning approach [22]. The level-$k$ critic approximator shall approximate the Q-function (15), while the level-$k$ actor shall approximate the appropriate defense or attack policy (16). In order to do that, we write the Q-function as,

$$Q_j^k(x, a_j^k) = \text{vech}(\hat{Q}^k_j) \text{Tr}(U_j^k \otimes U_j^k),$$ \hspace{1cm} (17)

where the Kronecker product-based polynomial quadratic polynomial function $(U_j^k \otimes U_j^k)$ is reduced to guarantee linear independence of the elements.

The vectorized Q-function (17) can be described in terms of the ideal weights $W_j^k = \text{vech}(\hat{Q}^k)$, leading to the compact form $Q_j^k(x, a_j^k) = W_j^k \text{Tr}(U_j^k \otimes U_j^k)$.

Since the ideal weights are unknown, we will consider the following estimated level-$k$ Q-function, according to the estimated critic weights, $\hat{W}_j^k = \text{vech}(\hat{Q}^k)$,

$$\hat{Q}_j^k(x, a_j^k) = \hat{W}_j^k \text{Tr}(U_j^k \otimes U_j^k),$$ \hspace{1cm} (18)

as well as the estimated actor approximator,

$$\hat{a}_j^k(x) = \tilde{W}_j^k \text{Tr}(x \otimes \tilde{x}),$$ \hspace{1cm} (19)

where the state $x$ is serving as the basis for the actor approximator and $\tilde{W}_{a,j}^k$ denotes the weight estimate of the level-$k$ agent’s policy.

The Q-function with the ideal weights, (15), has been shown [21] to satisfy the integral form of the HJB equation,

$$Q_j^k(x(t), a_j^k(t)) = Q_j^k(x(t - T_{IRL}), a_j^k(t - T_{IRL})) - \int_{t-T_{IRL}}^{t} (x^T \hat{M}_j^k x + a_j^k T \hat{R}_j a_j^k) d\tau,$$

where $T_{IRL} \in \mathbb{R}^+$ and during training of the level-$k$th defending agent $\hat{M}_j^k = M + \frac{1}{\gamma^2} P_d^k D T P_d^k, \ \hat{R}_j = R$, and during training of the level-$k$th adversarial agent, $\hat{M}_d^k = M - P_d^k G R^\top G T P_u^k$ and $\hat{R}_d = -\gamma^2$.

We now define the error based on the current estimate of the Q-function that we wish to drive to zero as,

$$e_j^k = \hat{Q}_j^k(x(t), a_j^k(t)) - \hat{Q}_j^k(x(t - T_{IRL}), a_j^k(t - T_{IRL})) + \int_{t-T_{IRL}}^{t} (x^T \hat{M}_j^k x + a_j^k T \hat{R}_j a_j^k) d\tau = \hat{W}_j^k \text{Tr}(U_j^k(t) \otimes U_j^k(t)) - \tilde{W}_j^k \text{Tr}(U_j^k(t - T_{IRL}) \otimes U_j^k(t - T_{IRL})).$$
+ \int_{t-T_{\text{int}}}^{t} (x^T M_k^k x + \alpha^k_j \dot{R}_j a_j^k) \, dt,
\]
as well as the policy error \( \epsilon_{j,a}^k = W_{a,k}^T x + (\hat{Q}_{j,a})^{-1} \hat{Q}_{j,a} x^k \), where the appropriate elements of the \( \hat{Q} \) matrix will be extracted from the critic estimate \( \hat{W}_k \).

Defining the squared error functions for the critic weights \( K_1 = \frac{1}{2} \| \epsilon_{j,a}^k \|^2 \) and the actor weights \( K_2 = \frac{1}{2} \| \epsilon_{j,a}^j \|^2 \), we construct the tuning rules by applying normalized gradient descent [23] as,
\[
\dot{W}_{j}^k = -\alpha^k \frac{\sigma}{(1 + \epsilon_{j,a}^k T^k)^2} \epsilon_{j,a}^k, \quad \dot{W}_{j,a}^k = -\alpha^k x \epsilon_{j,a}^k, \tag{20}
\]
where \( \sigma^k = (U_k^k(t) \otimes U_k^j(t) - (U_k^j(t-T_{\text{IRL}}) \otimes U_k^j(t-T_{\text{IRL}}))) \), and \( \alpha, \alpha_a \in \mathbb{R}^+ \) are tuning gains.

**Lemma 1.** Consider the costs for the defending (9) and the attacking (12) agents. The training of a level-\( k \)-agent is described by the Q-function approximator (18) and her optimal policy by (19). The tuning rules, for the critic is given by (20), and for the actor by (21). Then the closed-loop system of the augmented state \( \psi = [x^T \dot{W}_j^k - W_k^j \ W_{j,a}^k - W_k^{j,a}]^T \) is asymptotically stable at the origin if the critic gain is picked sufficiently larger than the actor gain and,
\[
1 < \alpha_a < \frac{1}{\delta (\lambda (R^{-1})^k (2\lambda (\hat{M}^k_j + Q_{j,a}^k R^{-1} Q_{j,a}^{T,k}) - \lambda (Q_{j,a}^k Q_{j,a}^{T,k})))}, \tag{21}
\]
where \( \delta \in \mathbb{R}^+ \).

**Proof:** The proof is provided in [24].

**B. Estimation of the Adversarial Levels**

In this section, we introduce an estimation algorithm that can be employed by the defender, in order to assess the mean intelligence level of the environment. The required data will be collected through sequential interactions with different attackers over predefined time periods \( T_{\text{int}} \in \mathbb{R}^+ \). To capture the “mean intelligence level,” we fit all arbitrary attack inputs to a reinforcement-like signal. However, we need to reward those elements of the vector that are closer to the appropriate level, i.e., to reinforce the minimum element.

As a result, we apply the softmax function to each element of \( r \), as
\[
\sigma^k = \frac{e^{-r \lambda}}{\sum_{i=1}^{\lambda} e^{-r_i \lambda}}, \tag{23}
\]
where \( \tau \) is the softmax temperature parameter. This will give us, \( \sigma = [\sigma^1 \sigma^2 \ldots \sigma^K]^T \). We model the distribution of the players over the different levels as a Poisson distribution with the following probability mass function, \( p^k = \frac{\lambda^k e^{-\lambda}}{K!} \). This probability shall also define the belief of the defender about the relative proportion of level-\( k \)-adversaries as, \( b^k = \sum_{r=1}^{\lambda} P^r \).

Our goal is to evaluate the parameter \( \lambda \). To this end, we use the following update rule for the mean of the observations,
\[
\lambda^+ = \lambda + (\hat{K}^T \sigma) n \over n + 1, \tag{24}
\]
where \( \hat{K} = [1 \ 2 \ldots \ K] \) is a vector containing the indexes of the levels we have trained and \( n \) is the number of different agents we have interacted with.

**C. Algorithmic Framework for Level Distribution Learning**

In Algorithm 1 that follows, we present the framework to learn the distribution of the different levels of the adversaries.

**Algorithm 1: Intelligence Level Learning**

1: **procedure**
2: Given initial state \( x_0 \), cost weights \( M, R, \gamma \), highest allowable level defined to be \( K \) and time window \( T_{\text{IRL}} \).
3: for \( k = 0, \ldots, K \) do
4: Set \( j := u \) to learn the level-\( k \)-defender policy.
5: Start with an initial guesses for \( W_k^j, W_{a,a}^j \).
6: Propagate the augmented system with states \( \chi = [x^T \ W_u^k \ W_{a,a}^k]^T \), according to (6), (20) and (21) until convergence.
7: Set \( j := d \) to learn the level-\( k \) adversary policy.
8: Start with initial guesses for \( W_u^d, W_{d,a}^d \).
9: Propagate the augmented system with states \( \chi = [x^T \ W_u^k \ W_{a,a}^k]^T \), according to (6), (20) and (21) until convergence. Go to 3.
10: **end for**
11: Define the interaction time with each adversary as \( T_{\text{int}} \), the number of total interactions \( n_{\text{int}} \) and an initial guess for \( \lambda \).
12: for \( t = 1, \ldots, n_{\text{int}} \) do
13: For \( t \in [t_i - T_{\text{int}}, t_i] \), measure the value of \( (22) \).
14: Compute the mean level according to (23).
15: Update \( \lambda \) based on (24). Go to 13 to interact with a different adversary.
16: **end for**
17: **end procedure**

**IV. SIMULATION**

Consider a second order system of the form (6) with, \( F = \begin{bmatrix} -1 & 0.25 \\ -1 & 0 \end{bmatrix}, G = \begin{bmatrix} 1 \\ 0 \end{bmatrix}^T, D = \begin{bmatrix} 1 \\ 0 \end{bmatrix}^T \).

We train up to level-5 intelligence. The system is under attack by 5 agents with level-2, 1 agent with level-1 and
4 agents of level-5. In Figure 1, we can see that after interacting with the adversarial agents in their environment, the defender manages to estimate the distribution of different levels. During transience, it can be seen in Figure 2, that the Poisson parameter converges to the actual mean value of the adversarial agents’ cognitive abilities.

V. CONCLUSION AND FUTURE WORK

This work proposes a framework that introduces non-equilibrium game theory into CPS security. An iterative procedure is presented, that shall allow the formulation of policies chosen by adversaries with different levels of intelligence. It is shown that, for a general non-zero-sum differential game, the best responses stem from Hamilton-Jacobi equations, conditioned by the belief of each player. After highlighting the correspondence between the general problem, and one derived by zero-sum game, which lead to the formulation of Riccati equations, we modify a Q-Learning framework to facilitate learning of policies with arbitrary levels. An algorithmic framework is constructed, that allows the defender to estimate the intelligence level distribution by collecting data from attackers in the environment.

Future efforts will focus on the development of synchronous algorithms, that are able to update the player’s beliefs and their best response Q-learning maps, simultaneously.

REFERENCES