Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 58-30 (2024) 97–102

Strategic Control of Experience-Weighted Attraction Model in Human-AI Interactions

Yuksel Arslantas [*] **Muhammed O. Sayin** [*]

[*] *Electrical and Electronics Engineering Department, Bilkent University, Ankara Türkiye 06800 (e-mail: yuksel.arslantas@bilkent.edu.tr, sayin@ee.bilkent.edu.tr).*

Abstract: This paper investigates a novel control framework for strategic Artificial Intelligence (AI) in human interactions. We leverage the Experience-Weighted Attraction (EWA) model, a widely used method for capturing human learning dynamics. EWA incorporates experiences to influence future choices through "attraction values" assigned to different actions. By treating these attraction values as the system state, we formulate the interaction between the AI and human as a stochastic control problem. This approach allows the AI to strategically influence the human's behavior by manipulating the environment or offering incentives that alter the attraction landscape, even under conditions of partial knowledge about the human agent's learning process. Our framework contributes to the field of human-AI interaction by providing a novel control method driven by the dynamics of human learning through EWA.

Keywords: Dynamic programming, Learning algorithms, Markov decision processes, Stochastic control

1. INTRODUCTION

The increasing popularity of Artificial Intelligence (AI) is transforming human-machine interaction, particularly in cyber-physical systems (CPS) where physical components integrate with computational intelligence. These complex systems require a deep understanding of human-AI interactions to achieve optimal performance and ensure safety (Shi et al., 2011; Humayed et al., 2017). A major challenge lies in the vulnerability of traditional learning algorithms to manipulation by strategically advanced agents like AI (Huang and Zhu, 2019; Vundurthy et al., 2023; Arslantas et al., 2024). This raises a critical question: how much can a strategic AI manipulate, assist, or guide an (human) agent within these complex CPS to achieve a more favorable outcome?

To tackle this problem, we take a control-theoretic approach, examining the repeated play of general-sum normal-form games between a human agent and a strategic AI agent. Our goal is to investigate how the AI agent can strategically influence the human's learning behavior to achieve a more favorable outcome. This outcome could be maximizing the joint payoff for both agents, or it could be maximizing the AI's own payoff at the expense of the human. We model the human agent's behavior using the Experience-Weighted Attraction (EWA) algorithm, which integrates belief learning and reinforcement learning models, making it particularly well-suited for capturing the complexity of human decision-making in strategic interactions. Unlike simpler models that focus solely on beliefs or received payoffs, EWA can account for both aspects, leading to a more comprehensive understanding of human behavior in games. The human agent employs the EWA algorithm with certain preselected parameters that may vary during the game. The AI agent, being strategically sophisticated, is aware of the human agent's use of the EWA algorithm. We show that the AI agent can use this awareness to control the human agent's learning behavior. By modeling the problem as a Markov Decision Process (MDP), the AI agent can maximize the discounted sum of payoffs over an infinite horizon. The main challenge of this approach is the continuous state space of the modeled MDP. To address this, we propose a quantization-based approximation method. This method approximates the MDP with a finite state version, allowing us to apply dynamic programming techniques. Note that we consider a simple AI as a proof of concept. However, by employing function approximation methods, we can explore more complex scenarios with more powerful AI agents.

Our work builds upon the existing research on strategizing against learning agents that has been a recent focus in the literature. Dong and Mu (2022) studied fictitious play in 2×2 games involving AI-human interactions, where the AI adopts fictitious play, demonstrating vulnerability results based on the game matrix's payoff values. Vundurthy et al. (2023) examined alternating fictitious play showing that a strategically sophisticated agent can leverage game and opponent knowledge by solving a linear programming problem to drive the opponent towards a more favorable mixed-strategy profile. Deng et al. (2019) demonstrated that sophisticated agents can secure equilibrium values against a class of no-regret learners.

Recent work by Arslantas et al. (2024) has focused on the vulnerability of Q-learning, proposing a stochastic control-based solution that uses the agents' Q-values as

10.1016/j.ifacol.2025.01.163

the system's state. They studied iteratively played normal-form games among multiple strategic agents and naive Q-learners. Similar to our work, they handle the continuum state space with a quantization-based approximation scheme both analytically and numerically. While Arslantas et al. (2024) and our work share similar approaches in analyzing the vulnerabilities of learning dynamics, our paper addresses more generic learning dynamics that encompass a wider range of learning rules. Furthermore, in contrast to Arslantas et al. (2024), which assumes complete knowledge for the strategic agent, we demonstrate that our results hold even when the perfect knowledge assumption is relaxed, as shown in our numerical results.

Additionally, there is significant literature on the falsification of reward functions in reinforcement learning (Huang and Zhu, 2019, 2021; Zhang et al., 2020). However, the underlying game structure, which prohibits the manipulation of rewards and potentially aligned objectives, distinguishes our vulnerability analysis of learning dynamics from this body of work.

These studies in the vulnerability of learning dynamics literature rely on specific assumptions about the type of the opponent's learning rule. Our work addresses this limitation by employing EWA model, introduced in the seminal work of Camerer and Hua Ho (1999) for human behavior in cyber-physical human systems (CPHS). Depending on the choice of EWA parameters, EWA encompasses many widely used and studied learning algorithms from the class of belief learning and reinforcement learning as illustrated in Figure 1, such as fictitious play, logit learning or Q-learning (Pangallo et al., 2022). This versatility makes EWA more suitable for modeling human behavior compared to using belief learning and reinforcement learning separately. Therefore, EWA provides a comprehensive description of human agents across a wide range of games (Pangallo et al., 2022; Gracia-Lázaro et al., 2012). To the best of our knowledge, our work is the first to leverage EWA in the context of human-AI interaction within CPHS.

The paper is organized as follows: In Section 2, we present the interaction between human and strategic AI agents as a game and introduce their respective behavioral rules: EWA for the human agent and a stochastic control approach with a quantization-based solution for the AI agent. In Section 3 and 4, we demonstrate our results in various types of games and conclude the paper, respectively.

2. STRATEGIC AI AND HUMAN INTERACTION

Consider a normal-form game $\mathcal{G}$ played over stages k, defined by the tuple $\langle \mathcal{A}, \mathcal{B}, u, \overline{u} \rangle$. In this game, $\mathcal{A}$ and $\mathcal{B}$ are the finite action sets for the strategic AI agent and human agents, respectively. The payoff functions $u : \mathcal{A} \times \mathcal{B} \to \mathbb{R}$ and $\overline{u} : \mathcal{A} \times \mathcal{B} \to \mathbb{R}$ represent the payoffs received by the AI agent and the human agent.

2.1 Human Agent

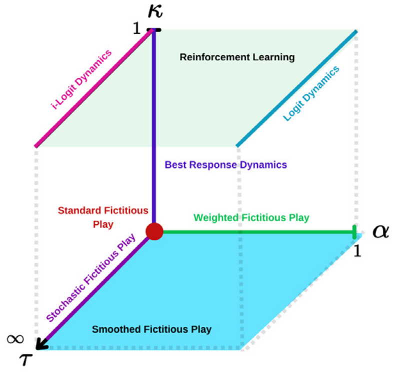

Fig. 1. An illustration of the EWA learning dynamics based on the parameters τ, α, κ, and δ. Belief-based learning dynamics, represented with a solid line, correspond to the case where $\delta = 1$. Payoff-based learning dynamics (i.e., reinforcement learning) are depicted as a shaded area for $\delta = 0$.

The human agent follows the EWA learning algorithm, which updates two variables at each stage. The first variable, N_k, represents the *experience* and is updated as follows

$$N_k = (1-\alpha)(1-\kappa)N_{k-1} + 1, \tag{1}$$

where $\alpha \in [0,1]$, $\kappa \in [0,1]$ and the initial experience vector $N_0 = 0$. The term $(1-\alpha)(1-\kappa)N_{k-1}$ discounts the past experiences, and the increment by one ensures that the experience always increases. The second variable, $Z_k(b)$ demonstrates the *attraction* of the agent for action b and is updated as

$$Z_k(b) = \frac{(1-\alpha)N_{k-1}Z_{k-1}(b) + \left[\delta + (1-\delta)\mathbb{I}_{\{b=b_k\}}\right]\overline{u}(\cdot, b)}{N_k}, \tag{2}$$

where $\mathbb{I}_{\{b=b_k\}}$ denotes the indicator function and $\delta \in [0,1]$ is attraction controller parameter determining the weight of the updates. For example, when $\delta = 0$, only the attractions for the played actions are updated. Conversely, when $\delta = 1$, all actions are updated with equal weight. The initial attraction vector $Z_0(b)$ can be chosen arbitrarily.

Fact 1. After some stages of the game, N_k tends to converge toward its fixed-point value $N^* = 1/(1-(1-\alpha)(1-\kappa))$ provided that $(1-\alpha)(1-\kappa) < 1$ (Pangallo et al., 2022).

Based on Fact 1, we can unify the *experience* and *attraction* updates in the following form

$$\begin{aligned} Z_k(b) = {} & (1-\alpha)Z_{k-1}(b) \\ & + [1-(1-\alpha)(1-\kappa)]\left[\delta + (1-\delta)\mathbb{I}_{\{b=b_k\}}\right]\overline{u}(\cdot, b). \end{aligned} \tag{3}$$

The human agent, given his attraction function $Z_k(b)$, responds according to the softmax function:

$$\sigma(Z_k)(b) = \frac{\exp(Z_k(b)/\tau)}{\sum_{\tilde{b}\in\mathcal{B}} \exp(Z_k(\tilde{b})/\tau)} \quad \forall b \in \mathcal{B}, \tag{4}$$

for some temperature parameter $\tau > 0$ controlling the exploration and $\sigma(Z_k)(b) \in (0,1]$ denotes the probability that the agent takes action $b \in \mathcal{B}$.

2.2 Strategic AI Agent

The goal of the strategic AI agent is to maximize the discounted sum of payoffs it can collect over the infinite horizon, i.e.,

$$\max_{\{a_k\}_{k=0}^{\infty}} \mathrm{E}\left[\sum_{k=0}^{\infty} \gamma^k u(a_k, b_k)\right], \tag{5}$$

where the expectation is taken with respect to the randomness on the action a_k and $b_k \sim \sigma(Z_k)$, and $\gamma \in (0,1)$ is some discount factor.

The strategic AI agent, equipped with knowledge of the underlying game structure and the human agent's behavior evolving according to EWA, utilizes this information to solve (5). Specifically, the strategic AI agent observes the game history $\{a_0, b_0, \ldots, a_{k-1}, b_{k-1}\}$ and uses an internal attraction function tracker to monitor the human agent's attraction function. It can then reformulate the problem as a *fully observable* MDP, where the state space encompasses all possible Z-vectors from (3). For strategic AI agent the objective (5) becomes to find the best policy for MDP $\mathcal{M}$ characterized by the tuple $\langle Z, \mathcal{A}, r, p, \gamma\rangle$ where $Z \subset \mathbb{R}^{|\mathcal{B}|}$ is the compact set of states, $\mathcal{A}$ is the action set of actions as in $\mathcal{G}$, the reward function $r : Z \times \mathcal{A} \to \mathbb{R}$ is given by $r(z, a) = \mathrm{E}[u(a, b)]$ that $b \sim \sigma(Z)$, $p(\cdot|\cdot)$ is the transition kernel for Z-vectors evolving according to (3) and $\gamma \in (0,1)$ is discount factor. Therefore, we can rewrite the goal of the strategic AI agent as follows

$$\max_{\{a_k\}_{k=0}^{\infty}} \mathrm{E}\left[\sum_{k=0}^{\infty} \gamma^k r(z_k, a_k)\right], \tag{6}$$

where the expectation is taken with respect to randomness on (z_k, a_k).

Fact 2. Since the set of all possible Z-vectors is a Polish space, being a compact subset of $\mathbb{R}^{|\mathcal{B}|}$, and the action set $\mathcal{A}$ is finite and state-invariant, there exists an optimal stationary policy $\pi : Z \times \mathcal{A} \to [0,1]$ for $\mathcal{M}$ (Puterman, 2014, Theorem 6.2.12).

Although there exists an optimal stationary policy, finding this policy for a strategic AI agent is challenging due to the continuum of the state space. For instance, when the EWA parameters are set to $\alpha > 0$, $\tau > 0$, $\delta = 1$, and $\kappa = 0$, the human agent follows smoothed fictitious play, and the Z-vector becomes a probability measure for the belief of the opponent's strategy, i.e., $Z = \prod_{|\mathcal{B}|}[0,1]$. Conversely, when $\delta = 0$, the human agent adopts reinforcement learning dynamics. For a specific type of algorithm, such as Q-learning, the Z-vector takes values within a continuum interval, i.e., $Z = \prod_{b\in\mathcal{B}}\left[Q^b_{\min}, Q^b_{\max}\right]$. Due to this continuum state space, the strategic AI agent cannot directly use dynamic programming methods such as value or policy iteration. These methods rely on Bellman equations that involve taking the maximum or expectation over the next states, which becomes intractable with infinitely many possible Z-vectors. Therefore, the strategic AI agent needs to approximate the state space to a finite set.

To approximate the MDP with a finite state space, the strategic AI agent can employ various methods from the literature. For example, it can leverage the linear programming solution of MDPs by selecting a basis to reduce the dimension of states, thus reducing the state space to a finite one (De Farias and Van Roy, 2003, 2004). Similarly, it can use approximate value iteration or policy iteration that utilizes similar preselected basis function methods (Tsitsiklis and Van Roy, 1996; Farahmand et al., 2010). Furthermore, it can employ any universal approximating architecture for the value function, as in neuro-dynamic programming, to find the optimal policy (Bertsekas and Tsitsiklis, 1996).

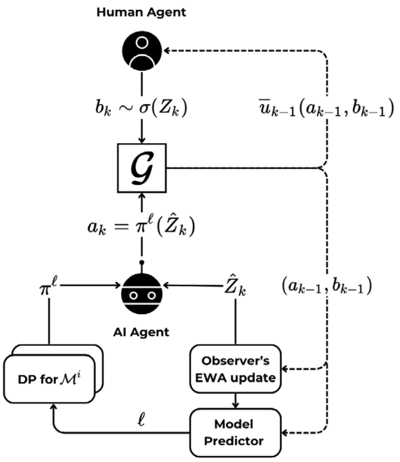

Fig. 2. The illustration demonstrates how the strategic AI agent and the human agent take actions during the repeated play of $\mathcal{G}$. The dashed lines indicate observations from the previous stage.

Algorithm 1 Strategic AI Agent

```
input: $\pi^i$ and $Z_0$
for each stage $k = 0, 1, \ldots$ do
    observe $(a_{k-1}, b_{k-1})$
    \\ Track the state
    if $k > 0$ then update $\hat{Z}^i_{k-1}$ to $\hat{Z}^i_k$ according to (3)
    \\ Predict the model
    if $k > 0$ then update $\lambda_{k-1}$ according to (9)
    find the best model type $\ell$
    take action $a_k \sim \pi^\ell(z_k)$
    receive reward $r_k = r(z_k, a_k)$
end for
```

Fig. 3. An algorithmic representation demonstrate how the strategic AI agent take actions in the repeated play of $\mathcal{G}$.

2.3 Approximation of the MDP

In this work, we propose a quantization-based approximation to tackle the continuum state space. Consider a quantization mapping $\Phi : Z \to \tilde{Z}$, where $\tilde{Z} \subset Z$. The set $Z_{\tilde{z}} := \{z \in Z : \Phi(z) = \tilde{z}\}$ satisfies $Z_{\tilde{z}} \cap Z_{\tilde{z}'} = \varnothing$ for all $\tilde{z} \neq \tilde{z}'$, and $\bigcup_{\tilde{z}\in\tilde{Z}} Z_{\tilde{z}} = Z$.

The strategic AI agent can use this quantization-based approach to approximate $\mathcal{M}$ as $\widetilde{\mathcal{M}}$, characterized by the tuple $\langle \tilde{Z}, \mathcal{A}, r_{\tilde{Z}}, \tilde{p}, \gamma\rangle$. Here, $\tilde{Z}$ is the finite range space of the quantizer Φ and the state space of the approximated MDP $\widetilde{\mathcal{M}}$. $\mathcal{A}$ is the set of finite actions as described in $\mathcal{G}$. The reward function $r_{\tilde{Z}} : \tilde{Z} \times \mathcal{A} \to \mathbb{R}$ is given by $r_{\tilde{Z}}(\tilde{z}, a) = \mathrm{E}[u(a, b)]$ where $b \sim \sigma(\tilde{Z})$. The transition probabilities are defined by

$$\tilde{p}(\tilde{z}_+|\tilde{z}, a) = \int_{Z_{\tilde{z}_+}} p(dz_+|\tilde{z}, a). \tag{7}$$

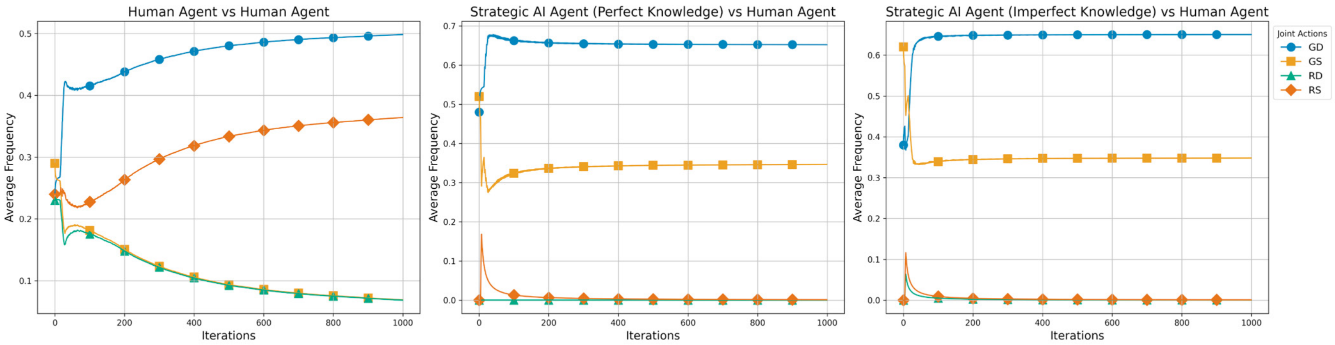

Fig. 4. The evolution of the empirical averages of the action profiles for the human agent vs human agent and strategic AI agent vs human agent in coordination game for perfect and imperfect knowledge.

We can rewrite the goal of the strategic AI agent in finite state space approximation as

$$\max_{\{a_k\}_{k=0}^{\infty}} \mathrm{E}\left[\sum_{k=0}^{\infty} \gamma^k r_{\tilde{Z}}(\tilde{z}_k, a_k)\right], \tag{8}$$

where $\tilde{z}_k = \Phi(z_k)$ is the quantized state at stage k. Hence, the strategic AI agent can adopt dynamic programming methods to obtain the optimal stationary policy.

Remark 1. The error between the value functions of $\mathcal{M}$ and $\widetilde{\mathcal{M}}$ can be quantified as in (Arslantas et al., 2024, Proposition 3) thanks to the quantization adopted by strategic agent and softmax response of the human agent. By increasing the quantization level, or alternatively, adopting another MDP approximation structure the strategic agent can increase its performance.

2.4 Model Prediction

In real-world scenarios, the AI agent might not have complete knowledge about the specific EWA parameters the human agent employs. Here, we consider a situation where the AI agent is aware of a finite set of possible EWA types that the human agent could adopt, but is unsure of the exact one.

To address this uncertainty, the AI agent pre-computes the optimal policy, π^i for each approximate MDP $\{\widetilde{\mathcal{M}}^i = \langle \tilde{Z}^i, \mathcal{A}, r_{\tilde{Z}^i}, \tilde{p}^i, \gamma\rangle\}_{i=1}^N$ modeling different EWA types. The strategic AI agent maintains a belief vector, $\lambda_k \in \Delta(N)$ at each stage k where $\Delta(N)$ represents the N-dimensional simplex. This vector represents the AI agent's current belief about the probability of each EWA type. The element λ_k^i denotes the probability assigned to the human agent having type i at stage k.

The strategic AI agent then uses its internal attraction function to track the state, $\hat{Z}_k^i$, and the mixed strategy of the human agent, β_k^i, for each EWA type i at stage k. After observing the action taken by the human agent, b_k, the strategic AI agent updates its belief as follows

$$\lambda_{k+1} = \frac{\lambda_k^T \beta_k}{\|\lambda_k^T \beta_k\|}, \tag{9}$$

where $\beta_k \in \mathbb{R}^N$ is defined as $\beta_k := \{\beta_k^i(b_k)\}_{i=1}^N$ and $\beta_k^i(b_k)$ is the probability of taking action b_k under the mixed strategy β_k^i. Once the belief vector is updated, the AI agent selects a policy to play in the next stage. It chooses the policy that is optimal for the most likely EWA type based on the updated belief. Simply the strategic AI agent plays according to the strategy that best aligns with the human agent it believes is most probable, i.e., $\pi_k = \pi^\ell$ where $\ell = \operatorname{argmax}_j \lambda_k^j$.

3. ILLUSTRATIVE EXAMPLES

We examine the performance of a strategic AI agent in CPHS under three distinct scenarios: coordination, anti-coordination and zero-sum games. We further illustrate the case where the AI agent does not have perfect knowledge about the EWA parameters of the human agent and constructs a belief for a finite set of types. As a proof of concept, we choose two possible EWA types. To model the human agent, we set the EWA parameters to $\alpha^1 = 0.1$, $\tau^1 = 0.01$, $\kappa^1 = 0.5$, and $\delta^1 = 0.5$. We specifically choose $\kappa^1 = 0.5$ and $\delta^1 = 0.5$ to ensure that the EWA model does not align strictly with belief or reinforcement learning, thus effectively representing the human agent. For the second type of EWA we determine the parameters as $\alpha^2 = 0.1$, $\tau^2 = 0.01$, $\kappa^2 = 0.6$, and $\delta^2 = 0.4$. We choose the forgetting factor and exploration rate for both of the types according to the payoffs in the game such that the naive agent gives importance to the recent actions and explore sufficiently. For the strategic AI agent, we set $\gamma = 0.8$. Furthermore, we quantize each Z-vector to 100 uniform intervals. In all scenarios, we first illustrate the cases where both agents adopt the same EWA learning dynamics, i.e., human agent vs. human agent. After establishing these benchmarks, we demonstrate the case where the strategic AI agent employs dynamic programming with perfect and imperfect knowledge.

Table 1. Coordination game between AI agent (row player) and human agent (column player)

	D	S
G	$(0.8, 0.8)$	$(0.2, 0.1)$
R	$(0.1, 0.2)$	$(0.5, 0.5)$

Coordination Game: Consider a traffic intersection scenario where an AI agent controls the traffic light and a single driver is present. The AI agent has two actions: green (G) and red (R), to control the flow of traffic, while

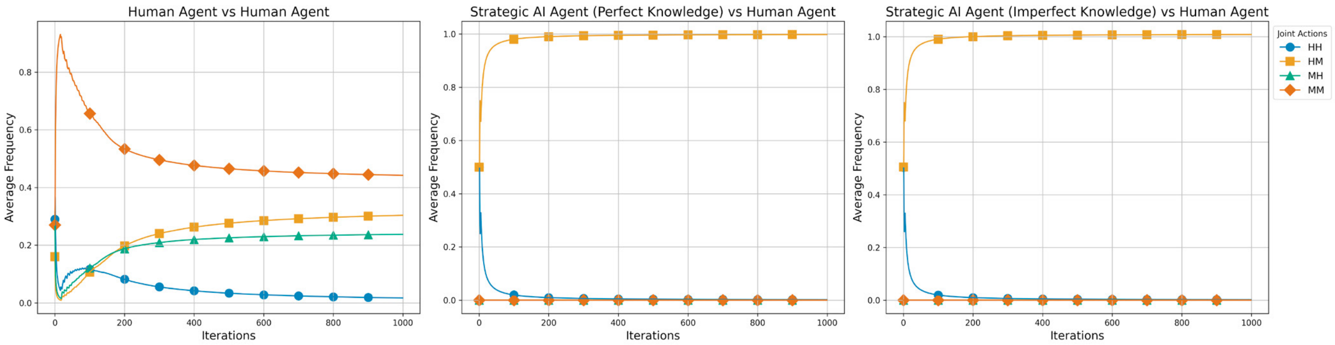

Fig. 5. The evolution of the empirical averages of the action profiles for the human agent vs human agent and strategic AI agent vs human agent in anti-coordination game for perfect and imperfect knowledge.

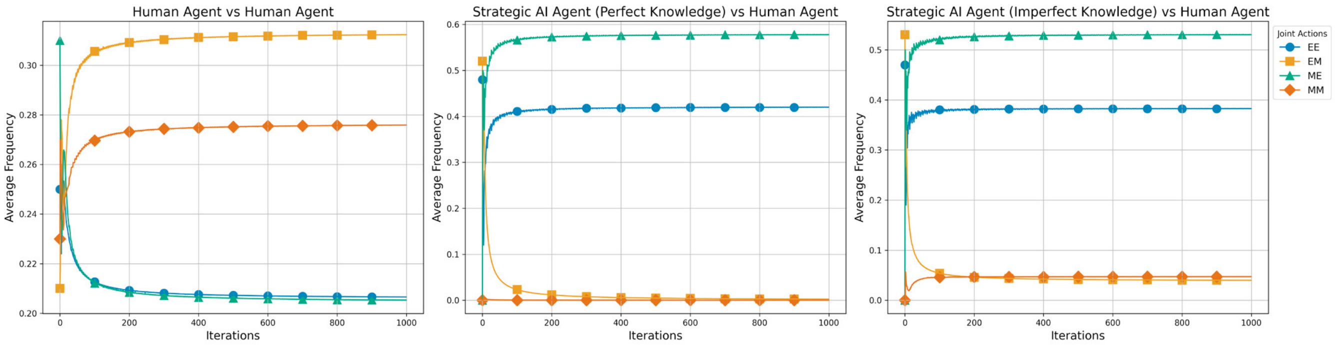

Fig. 6. The evolution of the empirical averages of the action profiles for the human agent vs human agent and strategic AI agent vs human agent in zero-sum game for perfect and imperfect knowledge.

the driver also has two actions: drive (D) or stop (S). The payoffs, as illustrated in Table 1, represent the effectiveness and safety of the different actions, encouraging both agents to find an optimal strategy for coordination.

When the agents follow EWA learning dynamics, they more frequently choose the joint action profile where the traffic light is green and the driver drives. However, when the strategic AI agent leverages its complete knowledge, this frequency increases as illustrated in Figure 4. Furthermore, the strategic AI drives the human agent toward a more favorable joint action where the traffic light is green but the driver stops, enhancing either traffic flow or safety. We observed an increase in the utility achieved by the strategic agent, rising from 0.3295 to 0.4387 under perfect knowledge, and to 0.3678 under imperfect knowledge. Similarly, the human agent's utility increased from 0.3268 to 0.4081 and 0.3291, respectively.

Table 2. Anti-coordination game between AI agent (row player) and human agent (column player)

	H	M
H	$(-1,-1)$	$(1,-0.5)$
M	$(-0.5,1)$	$(0.5,0.5)$

Anti-coordination Game: Imagine a market competition scenario where a strategic AI agent and a human agent represent competing companies, each deciding on their strategies for a new product launch. These companies have two possible actions: targeting the high-end market (H), which consists of fewer customers but yields higher profits, or targeting the mass market (M), which has a larger customer base but offers lower profits. The matrix game illustrating this scenario is shown in Table 2.

When the agents follow EWA dynamics, they tend to target the mass market, as depicted in Figure 5. However, when a strategic AI agent is involved, it can manipulate the joint actions such that it targets the high-end market to achieve more favorable profits while driving the human agent to target the mass market. In addition, the strategic AI agent's utility increases from 0.2097 to 0.3944 and 0.3934 in perfect and imperfect knowledge, respectively. Conversely, the human agent's utility decreases from 0.2391 to -0.5014 and -0.5016.

Table 3. Zero-sum game between AI agent (row player) and human agent (column player)

	E	M
E	$(-1,1)$	$(1,-1)$
M	$(1,-1)$	$(-1,1)$

Zero-sum Game: Consider a cybersecurity scenario involving a strategic AI agent (attacker) attempting to infiltrate a system and a human agent (defender) trying to protect it. The attacker has two options: sending deceptive emails to trick users into revealing sensitive information (E) or using malicious software to gain unauthorized access (M). The defender can counter these actions by strengthening email filters (E) or improving the malware detection system (M). This cybersecurity problem can be characterized as a zero-sum game, as depicted in Table 3.

In this game scenario, the strategic AI agent can leverage its knowledge to increase the frequency of mismatched joint actions. This result is illustrated in Figure 6 where the AI agent chooses M and the human agent chooses E more frequently. Furthermore, the strategic AI agent's achieved utility increases from 0.0162 to 0.3599 and 0.2509 for the perfect knowledge and imperfect knowledge cases, respectively.

4. DISCUSSION

We investigated how a strategic AI agent can strategize against a human agent modeled using EWA, assuming the strategic AI agent understands the human agent's learning dynamics and the underlying game structure. We tackled this problem from a control-theoretical perspective, considering the human agent's algorithm as a dynamical system with states represented by the human agent's attraction functions. The strategic AI agent's objective was formulated as an MDP with a continuous state space. To manage this complexity, we introduced a quantization-based approximation to reduce the state space dimensionality, allowing the MDP to be solved through dynamic programming. We also investigated the cases where the strategic AI agent has imperfect knowledge about the human agent and proposed a model prediction framework against the finitely many EWA types. Our numerical results showed that the strategic AI agent can manipulate the human agent to achieve favorable outcomes for itself in both perfect and imperfect knowledge cases.

This research lays the groundwork for understanding the vulnerabilities of the human agents when confronted by strategic AI agents in CPHS from a control-theoretical perspective. This insight can help us design more robust algorithms for practical applications or improve the performance of CPHS. Future research directions include *(i)* limiting the capabilities of strategic AI agents, *(ii)* investigating how strategic AI agents without perfect knowledge can learn the dynamics of human agents, *(iii)* exploring alternative approximation methods, *(iv)* applying these approaches to more complex real-world scenarios, and *(v)* designing more robust learning dynamics against strategic agents to robustify the human behavior in CPHS.

ACKNOWLEDGEMENTS

This work was supported by The Scientific and Technological Research Council of Türkiye (TUBITAK) BIDEB 2232-B International Fellowship for Early Stage Researchers under Grant Number 121C124.

REFERENCES

Arslantas, Y., Yuceel, E., and Sayin, M.O. (2024). Strategizing against Q-learners: A control-theoretical approach. *IEEE Control Systems Letters*, 8, 1733–1738.

Bertsekas, D. and Tsitsiklis, J.N. (1996). *Neuro-dynamic Programming*. Athena Scientific.

Camerer, C. and Hua Ho, T. (1999). Experience-weighted attraction learning in normal form games. *Econometrica*, 67(4), 827–874.

De Farias, D.P. and Van Roy, B. (2003). The linear programming approach to approximate dynamic programming. *Operations Research*, 51(6), 850–865.

De Farias, D.P. and Van Roy, B. (2004). On constraint sampling in the linear programming approach to approximate dynamic programming. *Mathematics of Operations Research*, 29(3), 462–478.

Deng, Y., Schneider, J., and Sivan, B. (2019). Strategizing against no-regret learners. In *Advances in Neural Information Processing Systems*, volume 32.

Dong, H. and Mu, Y. (2022). The optimal strategy against fictitious play in infinitely repeated games. In *Proceedings of the 41st Chinese Control Conference*.

Farahmand, A.m., Szepesvári, C., and Munos, R. (2010). Error propagation for approximate policy and value iteration. *Advances in Neural Information Processing Systems*, 23.

Gracia-Lázaro, C., Ferrer, A., Ruiz, G., Tarancón, A., Cuesta, J.A., Sánchez, A., and Moreno, Y. (2012). Heterogeneous networks do not promote cooperation when humans play a prisoner's dilemma. *Proceedings of the National Academy of Sciences*, 109(32), 12922–12926.

Huang, Y. and Zhu, Q. (2019). Deceptive reinforcement learning under adversarial manipulations on cost signals. In T. Alpcan, Y. Vorobeychik, J. Baras, and G. Dan (eds.), *International Conference on Decision and Game Theory for Security*, volume 11836 of *Lecture Notes in Computer Science*. Springer, Cham.

Huang, Y. and Zhu, Q. (2021). Manipulating reinforcement learning: Stealthy attacks on cost signals. In *Game Theory and Machine Learning for Cyber Security*, 367–388. John Wiley & Sons.

Humayed, A., Lin, J., Li, F., and Luo, B. (2017). Cyber-physical systems security—a survey. *IEEE Internet of Things Journal*, 4(6), 1802–1831.

Pangallo, M., Sanders, J.B., Galla, T., and Farmer, J.D. (2022). Towards a taxonomy of learning dynamics in 2× 2 games. *Games and Economic Behavior*, 132, 1–21.

Puterman, M.L. (2014). *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons.

Shi, J., Wan, J., Yan, H., and Suo, H. (2011). A survey of cyber-physical systems. In *International Conference on Wireless Communications and Signal Processing*, 1–6. IEEE.

Tsitsiklis, J.N. and Van Roy, B. (1996). Feature-based methods for large scale dynamic programming. *Machine Learning*, 22(1), 59–94.

Vundurthy, B., Kanellopoulos, A., Gupta, V., and Vamvoudakis, K.G. (2023). Intelligent players in a fictitious play framework. *IEEE Transactions on Automatic Control*.

Zhang, X., Ma, Y., Singla, A., and Zhu, X. (2020). Adaptive reward-poisoning attacks against reinforcement learning. In *International Conference on Machine Learning*, 11225–11234. PMLR.