Cooperative Deception in Swarms Against a Smart Observer

Stanislas de Charentenay, Alexandre Reiffers-Masson, Gilles Coppin, Caroline Lesueur-Grand, Jacques Petit-Frère

Cooperative Deception in Swarms Against a Smart Observer

Stanislas de Charentenay[1,2] [0009−0009−0282−2968],
Alexandre Reiffers-Masson[1] [0000−0002−4084−1977],
Gilles Coppin[1] [0000−0002−9193−425X], Caroline Lesueur[2] [0009−0000−8557−7180],
and Jacques Petit-Frère[2] [0009−0008−8384−5581]

[1] IMT Atlantique, Lab-STICC, Brest, France
[2] Thales LAS, Elancourt, France

Abstract. In this article, we study cooperative deception in swarms, in the context of a two-player zero-sum game played over a directed acyclic graph. A swarm of agents must navigate to goal destinations while misleading an intelligent adversary that observes only partial, aggregated signals and updates its belief over time to optimize disruptive actions. This interaction is formalized as a dynamic game with one-sided partial observability, capturing both coordinated swarm behavior and adaptive adversarial inference over a finite horizon. To compute the equilibrium strategies, we propose an algorithm based on fictitious play, where best responses are computed via linear programming. To address the exponential complexity of multi-stage planning, we introduce a compression technique that maps observation histories into compact information states, ensuring that our algorithm remains effective with such compressed states. This reduction enables efficient equilibrium computation even in long-horizon settings. Simulations demonstrate how swarm-level deception can strategically reduce adversarial effectiveness and support theoretical results on the algorithm's convergence and complexity.

Keywords: Swarms · Cooperative Deception · Directed Acyclic Graph · Finite-Horizon Two-Player Zero-Sum Partially Observable Stochastic Games · Linear Programming · Fictitious Play.

1 Introduction

Deception and camouflage are increasingly critical tools in adversarial environments, from military operations to cybersecurity. Whether in the physical world or digital networks, misleading an opponent's perception can provide a decisive strategic advantage. In this paper, we focus on *swarm-level deception*, where the misleading behavior emerges not from isolated agents but from coordinated, collective strategies. As adversaries grow more intelligent and adaptive, there is a pressing need to design deception mechanisms that are strategic, scalable, and grounded in rigorous models of adversarial interaction.

Recent work has explored various facets of multi-agent deception. Some focus on tactical coordination, such as robot patrol deception using decoys [10], leader

identity camouflage [4], or UAV trajectory misdirection [9,2]. Others explore strategic deception against rational adversaries, particularly in cybersecurity, where defenders use adaptive honeypots learned via reinforcement learning [7], or multi-stage deception to reduce attacker success over network paths using heuristic methods [8], or signaling games to model attacker-defender interaction with equilibrium analysis [3]. While these works provide valuable insights, they are either limited to single-stage or single-agent settings, or rely on heuristics that do not directly extend to computing optimal deception strategies in structured, multi-stage environments as we do.

We study such a scenario in which a swarm of agents (e.g., UAVs) traverses a spatial environment modeled as a directed acyclic graph (DAG) over multiple stages. At each stage, the swarm's distribution across the graph influences the effectiveness of adversarial actions. A smart adversary observes partial signals and continuously updates its belief about the swarm's configuration. Based on this evolving information, it will attempt to harm the swarm as it travels through the DAG. The swarm can cooperatively manipulate the observations to mislead the adversary's inference, turning deception into a dynamic strategic interaction.

This setting presents unique modeling and computational challenges. First, it requires orchestrating coordinated behavior across multiple agents, which involves planning in large combinatorial spaces. Second, the adversary is not passive: it is an intelligent observer capable of inferring hidden swarm states from partial information and adapting its strategy accordingly. As the consequences of being understood are costly for the swarm, this calls for a model where a rational adversarial reaction is taken into account.

We formalize this interaction as a two-player zero-sum game with one-sided partial observability, where the swarm (Player 1) selects both movement and deception strategies, and the adversary (Player 2) observes partial information on the state of the swarm and reacts accordingly. We focus on a finite horizon game, where the adversary updates its belief at each stage. Prior work has addressed one-sided partially observable dynamic games using approaches such as value iteration [6] and heuristic search methods [11]. However, these techniques are inapplicable due to the fact that they focus on infinite-horizon game. In contrast, we propose an algorithm based on *fictitious play*, where best responses are efficiently computed via linear programming.

Our main contributions are as follows:

- We introduce a novel two-player zero-sum game that models deception in multi-agent navigation over a DAG. The game captures the evolution of the swarm distribution and allows agents to cooperatively manipulate observations to mislead a smart adversary.
- We show that best responses for both players can be formulated as linear programs.
- We develop a learning-based solution approach using fictitious play, where each player iteratively computes a best response to the opponent's mixed strategy. We rigorously show that the *LP-based fictitious play* converges to the minmax equilibrium of our game.

- To improve scalability, we introduce a compression technique that maps the sequence of past observations into a compact information state, which preserves all relevant information for planning. This reduces the dimensionality of the strategy space and enables efficient equilibrium computation in complex games.
- We validate our framework with extensive simulations and experiments that confirm our theoretical results on the algorithm's convergence and computational complexity.

2 Model

We first define in 2.1 a general framework based on a zero-sum one-sided partially observable Markov game without explicitly specifying the swarm's internal state structure. Here, Player 1 and Player 2 will represent the swarm and the smart observer respectively. Then, through a concrete example in 2.2, we illustrate how this general framework can be instantiated to study the coordinated deception strategies of a swarm of UAVs evolving through a DAG.

2.1 State, Action, and Observation Spaces

We study a two-player zero-sum dynamic game over a finite number of stages H, indexed by discrete stages $t = 0, 1, \ldots, H$. At each stage t, the system is in a state $s_t \in S_t$, where S_t is a finite state space. Player 1 selects an action $a_t^1 \in A_t^1$, and Player 2 selects an action $a_t^2 \in A_t^2$, representing the swarm configuration it attempts to guess. We assume that A_t^1 and A_t^2 are finite spaces. Player 2 receives a partial observation $o_t \in O_t$, where O_t is a finite observation space for stage t. The observations are deterministically derived from the current state and Player 1's action. We call observation history at stage t the collection of all observations received by Player 2 until stage t, denoted by $o_{0:t} = (o_{t'})_{0 \leq t' \leq t}$. Note that for the particular case $t = -1$, the observation history $o_{0:-1}$ corresponds to an empty set $o_{0:-1} = \emptyset$. The space of histories will be denoted by $\mathcal{H}_t := \prod_{t'=0}^{t} O_{t'}$.

The state evolves according to a time-dependent Markov transition function:

$$s_{t+1} \sim p_t(\cdot \mid s_t, a_t^1).$$

At $t = 0$, the initial state s_0 is sampled from a known distribution $b_0 \in \Delta(S_0)$, where the notation $\Delta(S_0)$ represents the probability simplex over S_0.

Observations are generated as follows:

$$o_t = f_t^{\text{obs}}(s_t, a_t^1).$$

Observe that the observation for a given s_t and a_t^1 is unique and that the observations are independent of Player 2's actions. Also note that Player 1's action affects both the transition and the observation, whereas Player 2's action affects only the payofff. This will be more concrete in the subsection 2.2.

Player 2 observes only the sequence of observations $o_{0:t}$. However we assume that Player 1 knows the current state s_t at each stage as well as the sequence of observations from Player 2.

Definition 1. *For each stage t, history $o_{0:t-1} \in \mathcal{H}_{t-1}$ and state s_t, a behavioral strategy for Player 1 is defined as*

$$\pi_t^1(\cdot \mid s_t, o_{0:t-1}) \in \Delta(A_t^1).$$

It assigns probabilities to actions based on the current state and past observations of Player 2. For all stage t and history $o_{0:t} \in \mathcal{H}_t$, a behavioral strategy for Player 2 is defined as

$$\pi_t^2(\cdot \mid o_{0:t}) \in \Delta(A_t^2),$$

and therefore solely depends on the observation history. We denote by $\pi^1 := \pi_{0:H}^1$ and $\pi^2 := \pi_{0:H}^2$ the policies of players over all stages, and by Π^1 and Π^2 their corresponding sets.

The stagewise reward for Player 1 is a function $g_t(s_t, a_t^2)$ and Player 2 incurs a cost of $-g_t(s_t, a_t^2)$ at each stage. In this paper, we focus on the cumulative payoff:

$$J(\pi^1, \pi^2) = \mathbb{E}\left[\sum_{t=0}^{H} g_t(s_t, a_t^2)\right],$$

where the expectation is over the initial state distribution, the transition dynamics, and the randomized strategies of both players.

Definition 2. *For a fixed Player 2 policy π^2, Player 1's best response is:*

$$\mathbf{BR}_1(\pi^2) \in \arg\max_{\pi^1} J(\pi^1, \pi^2).$$

Similarly, Player 2's best response to π^1 is defined as:

$$\mathbf{BR}_2(\pi^1) \in \arg\min_{\pi^2} J(\pi^1, \pi^2).$$

A minmax equilibrium is a pair (π^{1*}, π^{2*}) satisfying:

$$\pi^{1*} = \mathbf{BR}_1(\pi^{2*}), \quad \pi^{2*} = \mathbf{BR}_2(\pi^{1*}).$$

We denote $J^* = J(\pi^{1*}, \pi^{2*})$ as the expected cumulative payoff at equilibrium.

2.2 A Concrete Instance: UAV Swarm Movement over a DAG

As a possible illustration of our model, let us consider a scenario in which Player 1 wants to control a swarm of unmanned aerial vehicles (UAVs) over a finite number of stages $H \in \mathbb{N}$ while Player 2 models an enemy aiming to retrieve a good estimation of the position of its members. The dynamic of the UAVs in the swarm is modeled as movement in a directed acyclic graph (DAG) $G = (\mathcal{X}, E)$, where the nodes $\mathcal{X}$ represent positions of interest and edges $E \subseteq \mathcal{X} \times \mathcal{X}$ represent admissible transitions between positions. Without loss of generality, we assume that the DAG is *layered*: that is, the node set is partitioned into disjoint subsets $\mathcal{X}_0, \mathcal{X}_1, \ldots, \mathcal{X}_H$, where each layer $\mathcal{X}_t$ contains the positions reachable at stages $t \in \{0, \ldots, H\}$. Transitions are only allowed between consecutive layers: if $(x, x') \in E$, then $x \in \mathcal{X}_t$ and $x' \in \mathcal{X}_{t+1}$. As such, we will denote $E_t \subseteq E$ the set of directed edges from $\mathcal{X}_t$ to $\mathcal{X}_{t+1}$. We also denote parent(ξ) and child(ξ) the parent node and child node respectively of directed edge $\xi \in E$.

State Space: At each stage t, the swarm configuration is represented by a function $s_t : \mathcal{X}_t \to \mathbb{N}$, where $s_t(x)$ denotes the number of UAVs at position $x \in \mathcal{X}_t$. The total number of UAVs is bounded by a known constant $n_{\max} \in \mathbb{N}$. The state space is therefore:

$$S_t = \left\{ (s_t(x))_{x \in \mathcal{X}_t} \,\middle|\, \sum_{x \in \mathcal{X}_t} s_t(x) \leq n_{\max} \right\}.$$

In this example, we assume that UAVs are statistically identicals, meaning that if two UAVs are in the same state, the Player 2 will not be able to differentiate them. This is why we focus on this aggregated representation of the states.

Action Space of Player 1: At stage t, the swarm takes a combination of two decisions for each edge $\xi \in E_t$, where $E_t \subseteq E$ denotes the set of directed edges from $\mathcal{X}_t$ to $\mathcal{X}_{t+1}$:

- $n_\xi \in \mathbb{N}$: The number of available UAVs that will follow the path represented by edge ξ,
- $\beta_\xi \in \{0, 1\}$: A choice of deceptive behavior where $\beta_\xi = 1$ hides one UAV in the group (if any), and $\beta_\xi = 0$ exposes all. Although the chosen set of behavior is small here for computation simplicity, we could extend the limit of dissimulated UAVs with higher values of β_ξ.

Observe that n_ξ must satisfy the flow conservation constraint:

$$\sum_{\substack{\xi \in E_t \\ \text{parent}(\xi) = x}} n_\xi = s_t(x), \quad \forall x \in \mathcal{X}_t.$$

We call $a_t^1 = (n_\xi, \beta_\xi)_{\xi \in E_t}$ the action of the swarm at stage t, representing the combination of these two choices for each edge in E_t. The admissible action set is:

$$A_t^1(s_t) = \left\{ (n_\xi, \beta_\xi)_{\xi \in E_t} \text{ subject to } \sum_{\substack{\xi \in E_t \\ \text{parent}(\xi) = x}} n_\xi = s_t(x),\ \forall x \in \mathcal{X}_t \right\}.$$

Transition Function: In this example, the state evolves deterministically. Given $s_t \in S_t$ and $a_t^1 = \{(n_\xi, \beta_\xi)\}_{\xi \in E_t} \in A_t^1(s_t)$, the next state $s_{t+1} \in S_{t+1}$ is computed by:

$$s_{t+1}(x') = \sum_{\substack{\xi \in E_t \\ \text{child}(\xi) = x'}} n_\xi, \quad \forall x' \in \mathcal{X}_{t+1}. \tag{1}$$

This defines a deterministic transition function:

$$p_t(s_{t+1} \mid s_t, a_t^1) = \begin{cases} 1, & \text{if } s_{t+1}, s_t, a_t^1 \text{ satisfy (1)}, \\ 0, & \text{otherwise.} \end{cases}$$

Observation function: Player 2 receives an observation at each stage derived from the swarm action. The observation function is defined as:

$$f_t^{\text{obs}}(s_t, a_t^1) = \{\max(n_\xi - \beta_\xi, 0)\}_{\xi \in E_t}.$$

This reflects that when $\beta_\xi = 1$, one UAV is hidden (if any are present), while with $\beta_\xi = 0$, all UAVs are visible. The observation space is:

$$O_t = \left\{ (o_\xi)_{\xi \in E_t} \,\middle|\, \sum_{\xi \in E_t} o_\xi \leq n_{\max} \right\}.$$

Action Space of Player 2: The adversary is modeled as an estimator that the swarm aims to deceive. At every stage t, the adversary will guess the current state of the swarm, which is its distribution over the reachable positions $\mathcal{X}_t$. As such the action space of the adversary A_t^2 at stage t will be equivalent to the state space S_t.

Payoff: In this scenario, we evaluate the performance of the swarm deception at stage t with the distance between the adversary estimation a_t^2 and the true state s_t. As such, the stage-wise reward of the swarm is:

$$g_t(s_t, a_t^2) = \sum_{x \in \mathcal{X}_t} \|s_t(x) - a_t^2(x)\|.$$

3 LP-based Fictitious Play Algorithm

In this section, an algorithm to find the equilibrium in our game is given. This algorithm is based on the Fictitious Play algorithm (Brown, 1951; Robinson, 1951) that iteratively compute approximate Nash equilibrium by maintaining averages of best-response policies. In our case, we show that the best responses of the players can be formulated as solutions of linear programs.

3.1 Players Best-Response: linear programs formulation

We can formulate the best responses as solution of linear programs, where the goal is to maximize (resp. minimize) the expected cumulated payoff J.

Let us first define, for all stage t, $s_t \in S_t$, $o_{0:t-1} \in \mathcal{H}_{t-1}$ and $a_t^1 \in A^1$ the occupancy measure y^1 corresponding to policy π^1, as

$$y^1(t, s_t, o_{0:t-1}, a_t^1) := \mathbb{P}_{\pi^1}(s_t, o_{0:t-1}, a_t^1).$$

We note that y^1 depends exclusively on the Player 1 policy π^1, which is only possible because Player 2 has no influence on the state transition p. We call $\mathcal{Y}^1$ the set of occupancy measures corresponding to Player 1 policies. This is key in our work as we can formulate the cumulated cost as a bilinear expression of y^1 and π^2:

$$J(\pi^1, \pi^2) = \sum_{t=0}^{H} \sum_{s_t, o_{0:t-1}, a_t^1, a_t^2} y^1(t, s_t, o_{0:t-1}, a_t^1) \cdot \pi_t^2(a_t^2 \mid o_{0:t-1}, f_t^{\text{obs}}(s_t, a_t^1)) \cdot g_t(s_t, a_t^2).$$

As such, we can introduce notations that use the occupancy measure instead of the Player 1 policy. For the expected cumulated payoff we introduce $\hat{J}(y^1, \pi^2) := J(\pi^1, \pi^2)$. For best responses we denote $\hat{\mathbf{BR}}_2(y^1) := \mathbf{BR}_2(\pi^1)$ and $\hat{\mathbf{BR}}_2(\pi^2) := \arg\max_{y^1} \hat{J}(y^1, \pi^2)$. In the following theorem, we show that we can optimize the occupancy measure with a linear program to compute the best response policy of Player 1.

Theorem 1 (Player 1 Best-Response LP). *Let π_2 be a fixed policy of Player 2. We define the following LP problem:*

$$\max_{y^1} \sum_{t=0}^{H} \sum_{s_t, o_{0:t-1}, a_t^1, a_t^2} y^1(t, s_t, o_{0:t-1}, a_t^1) \cdot \pi_t^2(a_t^2 \mid o_{0:t-1}, f_t^{obs}(s_t, a_t^1)) \cdot g_t(s_t, a_t^2) \tag{P1}$$

$$s.t. \sum_{a_0^1} y^1(0, s_0, \emptyset, a_0^1) = b_0(s_0), \quad \forall s_0 \in S_0 \tag{C1}$$

$$\sum_{a_{t+1}^1} y^1(t+1, s_{t+1}, o_{0:t}, a_{t+1}^1) = \sum_{s_t, o_{0:t-1}, a_t^1} p_t(s_{t+1} \mid s_t, a_t^1) \cdot \mathbb{I}[o_t = f_t^{obs}(s_t, a_t^1)] \cdot y^1(t, s_t, o_{0:t-1}, a_t^1),$$

$$\forall t \in [\![0, H-1]\!], \forall s_t \in S_t, \forall o_{0:t-1} \in \mathcal{H}_{t-1} \tag{C2}$$

$$\sum_{s_t, o_{0:t-1}, a_t^1} y^1(t, s_t, o_{0:t-1}, a_t^1) = 1, \quad \forall t \in [\![0, H]\!] \tag{C3}$$

$$y^1(t, s_t, o_{0:t-1}, a_t^1) \geq 0, \quad \forall t \in [\![0, H]\!], \forall (s_t, o_{0:t-1}, a_t^1) \in S_t \times \mathcal{H}_{t-1} \times A_t^1 \tag{C4}$$

Let y^{1*} be a solution to (P1)*. We define π^{1*} such that for $t \in [\![1, H]\!]$:*

$$\pi_t^{1*}(a_t^1 \mid s_t, o_{0:t-1}) = \begin{cases} \dfrac{y^{1*}(t, s_t, o_{0:t-1}, a_t^1)}{\sum_{\tilde{a}_t^1} y^{1*}(t, s_t, o_{0:t-1}, \tilde{a}_t^1)} & \text{if } \sum\limits_{\tilde{a}_t^1} y^{1*}(t, s_t, o_{0:t-1}, \tilde{a}_t^1) > 0, \\ \delta_t(a_t^1) & \text{otherwise,} \end{cases} \tag{2}$$

where $\delta_t \in \Delta(A_t^1)$ is an arbitrary distribution. Then the following are true:

1. *π^{1*} is a valid Player 1 policy and y^{1*} is its corresponding occupancy measure*
2. *The occupancy measure $y^{1*} = \hat{\mathbf{BR}}_1(\pi^2)$ is a best response to π^2.*

Proof. See in appendix 7.2.

Interpretation of the Linear Program: We will now give an intuition on why the linear program defined in theorem 1 can be used to derive the best response of Player 1. The objective function (P1) is equal the expected cumulative reward, when Player 2's policy is known. The constraints in the linear program ensure that y^1 is a valid occupancy measure. Indeed, constraints (C1) and (C2) ensure consistency with the initial distribution, and flow conservation over time respectively. The two of them ensure that y^1 evolves according to the game's transition and observation dynamics. Finally, (C3) and (C4) ensure that y^1 defines a valid probability distribution. Once the occupancy measure is known, the optimal policy can be derived using (2), obtained through Bayes' theorem.

Player 2 Best-Response: Since Player 2 actions does not influence the state transitions, we can optimize Player 2's policy directly rather than using occupancy measures.

Theorem 2 (Player 2 Best-Response LP). *Let π^1 a fixed Player 1 policy and y^1 its corresponding occupancy measure. We define the following LP problem:*

$$\min_{\pi^2} \sum_{t=1}^{H} \sum_{s_t, o_{0:t-1}, a_t^1, a_t^2} y^1(t, s_t, o_{0:t-1}, a_t^1) \cdot \pi_t^2(a_t^2 \mid o_{0:t-1}, f_t^{obs}(s_t, a_t^1)) \cdot g_t(s_t, a_t^2) \tag{P2}$$

$$s.t. \sum_{a_t^2} \pi_t^2(a_t^2 \mid o_{0:t}) = 1, \quad \forall t \in [\![0, H]\!], \forall o_{0:t} \in \mathcal{H}_t \tag{C5}$$

$$\pi_t^2(a_t^2 \mid o_{0:t}) \geq 0, \quad \forall t \in [\![0, H]\!], \forall o_{0:t} \in \mathcal{H}_t, \forall a_t^2 \in A_t^2 \tag{C6}$$

Let π^{2*} a solution to (P2)*, then it is a best response to occupancy measure y^1:*

$$\pi^{2*} = \hat{\mathbf{BR}}_2(y^1)$$

Proof. See in appendix 7.3

Difference with Linear Program of Player 1: In the case of the Player 2, its actions have no impact on the state transitions. As such, the expected cumulated reward is linear in the policy π^2 of Player 2 and we can directly optimize the policy in our Linear Program. Therefore in this case the objective function (P2) still represents the expected reward, while the constraints (C5) and (C6) ensure that π^2 is a valid policy of the Player 2.

3.2 LP-Based Fictitious Play Algorithm

Using the previous results, we present the pseudo-code of Algorithm 1. This algorithm is based on fictitious play with best responses computed using previously defined linear programs.

Algorithm 1 LP-Based Fictitious Play

Initialize counter $k = 0$
Initialize occupancy measure $y^{1,(0)}$ and Player 2 policy $\pi^{2,(0)}$
while not converged **do**
  Compute $\hat{\mathbf{BR}}_1(\pi^{2,(k)})$ using linear program (P1)
  Compute $\hat{\mathbf{BR}}_2(y^{1,(k)})$ using linear program (P2)
  Update $y^{1,(k+1)} = \left(1 - \frac{1}{k+1}\right) y^{1,(k)} + \frac{1}{k+1}\hat{\mathbf{BR}}_1(\pi^{2,(k)})$
  Update $\pi^{2,(k+1)} = \left(1 - \frac{1}{k+1}\right) \pi^{2,(k)} + \frac{1}{k+1}\hat{\mathbf{BR}}_2(y^{1,(k)})$
  $k \leftarrow k + 1$
end while
Compute $\pi^{1,(k)}$ from $y^{1,(k)}$ with expression (2)
return $\pi^{1,(k)}$, $\pi^{2,(k)}$

The algorithm terminates when a convergence criterion is met. In our experiments, we evaluate the convergence with the difference in the players best-responses:

$$\hat{J}(\hat{\mathbf{BR}}_1(\pi^{2,(k)}), \pi^{2,(k)}) - \hat{J}(y^{1,(k)}, \hat{\mathbf{BR}}_2(y^{1,(k)})). \tag{3}$$

The resulting policies approximate a Nash equilibrium of the game.

3.3 Convergence of the Algorithm

The following theorem proves the convergence of the proposed LP-based Fictitious Play Algorithm in continuous time.

Theorem 3 (Fictitious play convergence in occupancy form). *Let* $y^1 : \mathbb{R}_+ \to \mathcal{Y}^1$ $\pi^2 : \mathbb{R}_+ \to \Pi^2$ *be two functions that are solutions of the following equation:*

$$\begin{aligned} \dot{y}^1(\tau) &\in \hat{\mathbf{BR}}_1(\pi^2(\tau)) - y^1(\tau), \\ \dot{\pi}^2(\tau) &\in \hat{\mathbf{BR}}_2(y^1(\tau)) - \pi^2(\tau), \\ y^1(0) \in \mathcal{Y}^1, &\quad \pi^2(0) \in \Pi^2. \end{aligned}$$

Then the value difference defined as:

$$v(\tau) = \hat{J}\Big(\hat{\mathbf{BR}}_1(\pi^2(\tau)), \pi^2\Big) - \hat{J}\Big(y^1(\tau), \hat{\mathbf{BR}}_2(y^1(\tau))\Big),$$

satisfies:

$$v(\tau) \leq v(0)e^{-\tau}$$

Remark 1. To obtain the convergence of the algorithm 1. in a finite number of steps and not in continuous time, one needs simply to use the theory of stochastic differential inclusion [1]. More precisely, one need to apply Theorem 3.6 with Proposition 2.1 and Corollary 3.27 from [1].

Although Theorem 3 proves the that the convergence speed of algorithm 1 converges exponentially in terms of iterations, the time taken for one iteration still depends on the complexity of our linear programs.

4 History compression

4.1 History compression model

We now introduce a compressed representation of observation histories, in order to reduce the size of the policy and optimization space. At stage t, rather than conditioning actions on the full observation history $o_{0:t}$, players may instead use a compressed information state $z_t \in \mathcal{Z}_t$, where $\mathcal{Z}_t$ is the set of the new compressed information states would be much smaller that $\mathcal{H}_t$. Although the compression reduces how strong and smart Player 2 is, the aim is to find compression techniques that keep the relevant information and cuts the redundant parts of the observation history. The compressed states at stage t would be defined recursively from a predetermined compression function φ_t:

$$z_{t+1} = \varphi_t(z_t, o_t),$$

with initial value $z_{-1} = \emptyset$. We call φ-compressed policy of Player 1 a sequence of mappings $\tilde{\pi}^1_t : S_t \times Z_{t-1} \to \Delta(A^1_t)$, and φ-compressed policy of Player 2 a sequence of mappings $\tilde{\pi}^2_t : Z_t \to \Delta(A^2_t)$. Similarly to the case without compression, we use occupancy measures corresponding to φ-compressed policies of Player 1 in the LP formulation:

$$\tilde{y}^1(t, s_t, z_{t-1}, a^1_t) = \mathbb{P}(s_t, z_{t-1}, a^1_t \mid \tilde{\pi}^1).$$

Theorem 4 (Player 1 Best-Response LP with Compression). *Let $\tilde{\pi}^2$ be a fixed policy of Player 2. The best response of Player 1 is given by the solution to the following linear program:*

$$\max_{\tilde{y}^1} \sum_{t=0}^{H} \sum_{s_t, z_{t-1}, a^1_t, a^2_t} g_t(s_t, a^2_t) \cdot \tilde{\pi}^2_t(a^2_t \mid \varphi_t(z_{t-1}, f^{obs}_t(a^1_t))) \cdot \tilde{y}^1(t, s_t, z_{t-1}, a^1_t) \tag{P3}$$

$$s.t. \quad \sum_{a^1_0 \in A^1_0} \tilde{y}^1(0, s_0, \emptyset, a^1_0) = b_0(s_0), \quad \forall s_0 \in S_0 \tag{C7}$$

$$\sum_{a^1_{t+1}} \tilde{y}^1(t+1, s_{t+1}, z_t, a^1_{t+1}) = \sum_{s_t, z_{t-1}, a^1_t} p_t(s_{t+1} \mid s_t, a^1_t) \cdot \mathbb{I}[\varphi_t(z_{t-1}, f^{obs}_t(a^1_t)) = z_t] \cdot \tilde{y}^1(t, s_t, z_{t-1}, a^1_t),$$

$$\forall t \in [\![0, H-1]\!],\ \forall s_{t+1} \in S_{t+1},\ \forall z_t \in Z_t \tag{C8}$$

$$\sum_{s_t, z_{t-1}, a^1_t} \tilde{y}^1(t, s_t, z_{t-1}, a^1_t) = 1, \quad \forall t \in [\![0, H]\!] \tag{C9}$$

$$\tilde{y}^1(t, s_t, z_{t-1}, a^1_t) \geq 0, \quad \forall t \in [\![0, H]\!], \forall (s_t, z_{t-1}, a^1_t) \in S_t \times Z_{t-1} \times A^1_t \tag{C10}$$

Let $\tilde{y}^{1*}$ be a solution to (P3)*. We define $\tilde{\pi}^{1*}$ as:*

$$\tilde{\pi}_t^{1*}(a_t^1 \mid s_t, z_{t-1}) = \begin{cases} \dfrac{\tilde{y}^{1*}(t, s_t, z_{t-1}, a_t^1)}{\sum_{\tilde{a}_t^1} \tilde{y}^{1*}(t, s_t, z_{t-1}, \tilde{a}_t^1)} & \text{if } \sum_{\tilde{a}_t^1} \tilde{y}^{1*}(t, s_t, z_{t-1}, \tilde{a}_t^1) > 0, \\ \delta_t(a_t^1) & \text{otherwise,} \end{cases} \tag{4}$$

where $\delta_t \in \Delta(A_t^1)$ is an arbitrary distribution. Then the following are true:

1. *$\tilde{\pi}^{1*}$ is a valid Player 1 policy and $\tilde{y}^{1*}$ is its corresponding occupancy measure*
2. *The occupancy measure $\tilde{y}^{1*} = \hat{\mathbf{BR}}_1(\tilde{\pi}^2)$ is a best response to $\tilde{\pi}^2$.*

Proof. See Appendix 7.4.

The computation of the optimal policy $\tilde{\pi}^{1*}$ follows the same formula as in the full history case, by simply substituting $o_{0:t}$ with the compressed state z_{t-1} and the compressed history z_{t-1}, resulting in a more compact representation that requires significantly less memory than policies based on the full observation history.

We now describe how to compute the best response of Player 2 to a fixed policy $\tilde{\pi}^1$ of Player 1. As in the previous formulation, observation histories are compressed into states $z_t \in Z_t$ via a recursive function φ_t. Player 2 chooses actions based on these compressed states.

Theorem 5 (Player 2 Best-Response LP with Compression). *Let $\tilde{\pi}^1$ be a fixed policy for Player 1, and let $\tilde{y}^1$ be its corresponding occupancy measure. Then the best-response of Player 2 is the solution to the following linear program:*

$$\min_{\tilde{\pi}^2} \sum_{t=0}^{H} \sum_{s_t, z_{t-1}, a_t^1, a_t^2} g_t(s_t, a_t^2) \cdot \tilde{\pi}_t^2(a_t^2 \mid \varphi_t(z_{t-1}, f_t^{obs}(a_t^1))) \cdot \tilde{y}^1(t, s_t, z_{t-1}, a_t^1) \tag{P4}$$

$$s.t. \sum_{a_t^2} \tilde{\pi}_t^2(a_t^2 \mid z_t) = 1, \quad \forall t \in [\![0, H]\!],\ \forall z_t \in Z_t \tag{C11}$$

$$\tilde{\pi}_t^2(a_t^2 \mid z_t) \geq 0, \quad \forall t \in [\![0, H]\!],\ \forall z_t \in Z_t,\ \forall a_t^2 \in A_t^2 \tag{C12}$$

Let $\tilde{\pi}^{2*}$ be a solution to (P4)*, then it is a best response to occupancy measure $\tilde{y}^1$:*

$$\tilde{\pi}^{2*} = \hat{\mathbf{BR}}_2(\tilde{y}^1)$$

Proof. The proof is identical to the case with no compression described in Appendix 7.3.

4.2 Sliding Window Compression

A natural and simple instance of the history compression model is the *sliding window compression*, where the compressed state retains only the last k observations.

Let $k \in \mathbb{N}$ be the window size. We define the compressed space for stage $t \geq k$ as:

$$Z_t := O_{t-k+1} \times \cdots \times O_t,$$

with the convention that for $t < k$, we define $Z_t := O_1 \times \cdots \times O_t$, i.e., the window grows with time until it reaches size k. The initial compressed state is $z_{-1} = \emptyset$.

The compression update function φ_t operates as:

$$z_{t+1} = \varphi_t(z_t, o_{t+1}) = \begin{cases} (z_t, o_{t+1}) & \text{if } t < k-1, \\ (o_{t-k+2}, \ldots, o_{t+1}) & \text{if } t \geq k-1, \end{cases}$$

meaning that z_{t+1} always stores the most recent k observations at stage $t+1$.

5 Numerical Results

5.1 Scenario and Qualitative Behavior

To illustrate the behavior of the proposed algorithm, we consider a simplified instance of the swarm deception game based on the model described in Section 2.2. In this simulation scenario, we assume that the swarm evolves through a line. At each stage, the swarm always has a single position and path to take at each stage t, and no repartition choice. As such the action space will be $A_t = \{0, 1\}$, where 1 represents the choice to hide one UAV on that edge and 0 to show all the swarm. The adversary receives an observation affected by the hiding mechanism, in the form of the number of UAVs in the swarm that are not hidden, and attempts to estimate the true distribution of UAVs across positions. The DAG studied in this example is a particular case as the state and action space at every stage stays small. Although the size of the observation space still explodes with the number of stages, this example will stay tractable.

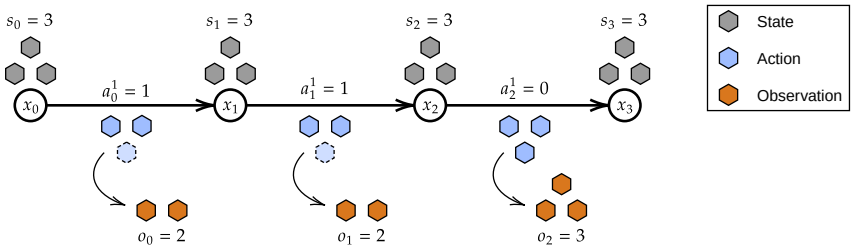

Fig. 1. Example of swarm evolution throughout a simple line graph used in the simulations.

In our experiments, we simulate this game with a maximum swarm size $n_{\max} = 5$. In this case, the size of state and action space at each stages t will be

$|S_t| = 6$ and $|A_t| = 2$. The observation is deterministic and computed from the swarm action using the function $f_t^{\text{obs}}(a_t^1) = \max(n_\xi - \beta_\xi, 0)$, as described earlier. The size of the observation history $\mathcal{H}_t$ at stage t will be $|\mathcal{H}_t| = 6^t$. At each stage, the adversary (Player 2) selects an action $a_t^2 \in S_t$, which is interpreted as its estimate of the true swarm distribution. The payoff is defined as the negative ℓ_1-distance between the true swarm configuration and the adversary's estimate. An example of a game instance in the described scenario is illustrated in diagram 1 with a horizon $H = 3$, with an initial state of 3 UAVs in the swarm.

The objective of our simulations are twofold: to quantify the impact of history compression on the computational cost associated with solving best-response linear programs, and to evaluate the convergence behavior of the suggested algorithm in terms of iterations.

5.2 Impact of History Compression on the Linear Programs Runtime

We begin our numerical analysis by evaluating how different observation compression schemes affect the computational cost of solving the best-response linear programs for each player. Specifically, we compare the LP runtimes under three settings: no compression, sliding window of size 2, and sliding window of size 1. The experiment is conducted for increasing values of the game horizon H, ranging from 2 to 6.

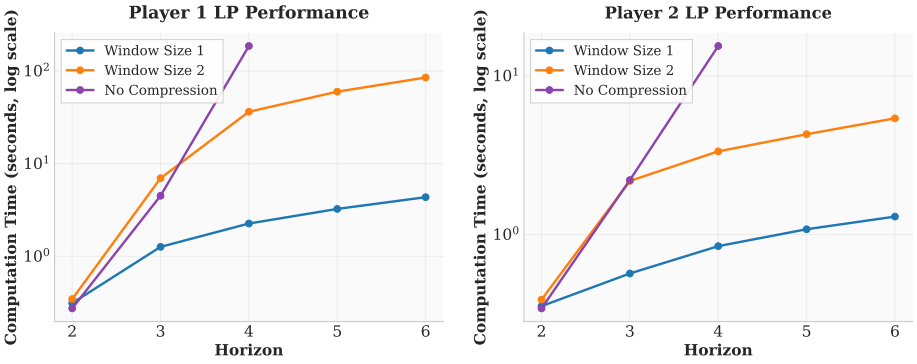

Fig. 2. Computation time comparison across window sizes for Player 1 and Player 2 Linear Programs

The results are presented in Figure 2. For each player, we compute the time required to solve the LP for the different horizon values. The plots reveal the exponential growth in computation time when no compression is applied as expected. In the case of no compression, for the Player 1 LP, we witness an explosion in computation time from 4.5 seconds for a horizon of 3 to 186.4 seconds for a horizon of 4. In contrast, sliding window compression offers a significant

improvement in scalability. Notably, a window size of 1 achieves the best performance, reducing runtime by several orders of magnitude as the horizon increases with a computation time of 4.6 seconds for a horizon of 6 in the Player 1 LP.

We are facing with the following trade-off: Not using compression allows for richer policy representations, however it quickly becomes computationally infeasible as the time horizon grows. In a futur work, we will optimize the compression to tackle this tradeoff.

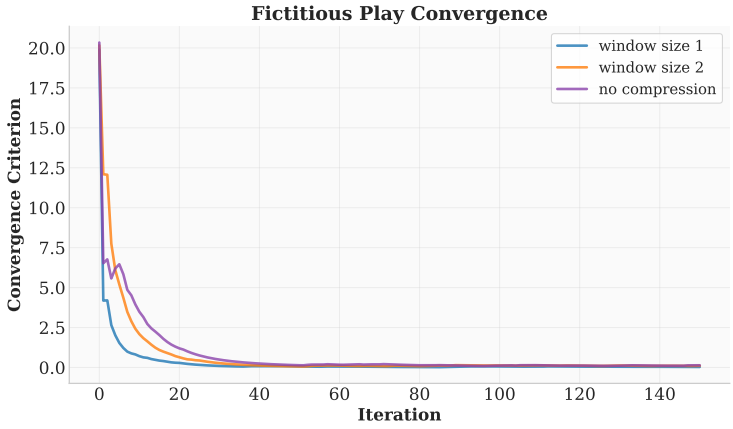

Fig. 3. Evolution of the convergence criterion across fictitious play iterations.

5.3 Convergence of Fictitious Play Algorithm

We now evaluate the convergence behavior of the fictitious play algorithm under the different observation compression schemes. The convergence criterion is defined in (3.2) as the performance gap between the current strategy of one player and the best response of the opponent. At equilibrium, this gap vanishes, indicating that both players have reached mutually optimal strategies.

Figure 3 shows the evolution of the convergence criterion over 100 fictitious play iterations, for the three compression settings described previously. In all cases, the error decreases exponentially toward zero, confirming the results of Theorem 3 in this setting.

These results support the theoretical guarantees of fictitious play, even in partially observable settings with compressed information. While compression may restrict policy expressiveness, it does not prevent convergence in our experiments.

6 Conclusion

In this article, we introduced a new formal game-theoretic framework to model deception strategies for a multi-agent system navigating through a directed acyclic graph (DAG). Specifically, we defined a two-player zero-sum dynamic game that captures the strategic conflict between a swarm aiming to conceal its true distribution in the DAG and an adversary attempting to estimate this distribution through partial observations.

To efficiently find equilibrium strategies in this complex setting, we proposed an algorithm based on fictitious play, utilizing linear programming to compute best-response strategies for both players. Recognizing the computational challenge posed by the exponential growth of observation histories with the number of stages, we further introduced a version of our algorithm with compression of the history, significantly mitigating computational complexity.

Our numerical experiments demonstrated the efficiency of this compression scheme, even in extended horizon. Additionally, the theoretical and numerical results confirmed the convergence and practical applicability of our fictitious play-based approach.

Future research directions include the optimization of the compression function, in order to minimize the amount of relevant information lost with the dimension reduction of the observation history. Other approximations techniques, such as the use of neural networks, are to be explored to compute efficiently the best responses in setting with very large state and action spaces. Regarding the application of this algorithm on more concrete examples, further research could be conducted on constructing a DAG approximation of a real-case continuous environment using path optimization methods.

Acknowledgments. The authors would like to thank the Agence de l'Innovation de Défense (AID) for their financial support.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Benaïm, M., Hofbauer, J., Sorin, S.: Stochastic approximations and differential inclusions. SIAM Journal on Control and Optimization **44**(1), 328–348 (2005)
2. Bildik, E., Tsourdos, A., Perrusquía, A., Inalhan, G.: Decoys deployment for missile interception: A multi-agent reinforcement learning approach. Aerospace **11**(8), 684 (2024)
3. Carroll, T.E., Grosu, D.: A game theoretic investigation of deception in network security. Security and Communication Networks **4**(10), 1162–1172 (2011)
4. Deka, A., Luo, W., Li, H., Lewis, M., Sycara, K.: Hiding leader's identity in leader-follower navigation through multi-agent reinforcement learning. In: 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 4769–4776. IEEE (2021)

5. Hofbauer, J., Sorin, S.: Best response dynamics for continuous zero-sum games. Discrete and Continuous Dynamical Systems Series B **6**(1), 215 (2006)
6. Horák, K., Bošanskỳ, B., Kovařík, V., Kiekintveld, C.: Solving zero-sum one-sided partially observable stochastic games. Artificial Intelligence **316**, 103838 (2023)
7. Huang, L., Zhu, Q.: Adaptive honeypot engagement through reinforcement learning of semi-markov decision processes. In: Decision and Game Theory for Security. GameSec 2019. Lecture Notes in Computer Science, vol 11836. Springer, Cham (2019)
8. Huang, L., Zhu, Q.: Farsighted risk mitigation of lateral movement using dynamic cognitive honeypots. In: International conference on decision and game theory for security. pp. 125–146. Springer (2020)
9. Li, Y., Shi, C., Yan, M., Zhou, J.: Mission planning and trajectory optimization in uav swarm for track deception against radar network. Remote Sensing **16**(18), 3490 (2024)
10. Talmor, N., Agmon, N.: On the power and limitations of deception in multi-robot adversarial patrolling. In: IJCAI. pp. 430–436 (2017)
11. Tomášek, P., Horák, K., Aradhye, A., Bošanskỳ, B., Chatterjee, K.: Solving partially observable stochastic shortest-path games. In: IJCAI-21 Proc. of the 30th International Joint Conference on Artificial Intelligence. pp. 4182–4189 (2021)

7 Appendix

7.1 Preliminaries

Definition 3 (Information State Space of Player 1). *Let $H \in \mathbb{N}$ be the time horizon of the game. For each stage $t \in [\![0, H]\!]$, we define the information state space $\mathcal{I}_t$ as a finite set of all possible realizations of the data available to Player 1 at stage t.*

An information state $\iota_t \in \mathcal{I}_t$ represents the variable on which Player 1 behavioral strategy at stage t is conditioned on. In our case, we have $\iota_t = (s_t, o_{0:t-1})$ in the regular case and $\iota_t = (s_t, z_{t-1})$ in the case with compression. This definition will enable us to prove results for both cases simultaneously.

The next lemma is used to prove theorem 1 and 4.

Lemma 1. *Let $\{y^1(t, \iota_t, a_t^1)\}_{t=0}^{H-1} \subset \mathbb{R}_{\geq 0}$ be a collection of nonnegative variables over information states $\iota_t \in \mathcal{I}_t$ and actions $a_t^1 \in A_t^1$, satisfying:*

(L1) Initial distribution:

$$\sum_{a_0^1} y^1(0, \iota_0, a_0^1) = \mathbb{P}(\iota_0), \quad \forall \iota_0 \in \mathcal{I}_0,$$

(L2) Flow consistency:

$$\sum_{a_{t+1}^1} y^1(t+1, \iota_{t+1}, a_{t+1}^1) = \sum_{\iota_t, a_t^1} \mathbb{P}(\iota_{t+1} \mid \iota_t, a_t^1) \cdot y^1(t, \iota_t, a_t^1) \quad \forall t \in [\![0, H-1]\!].$$

(L3) Normalization and nonnegativity:

$$\sum_{\iota_t, a_t^1} y^1(t, \iota_t, a_t^1) = 1, \quad \text{and } y^1(t, \iota_t, a_t^1) \geq 0 \quad \forall t \in [\![0, H]\!], \forall (\iota_t, a_t^1) \in \mathcal{I}_t \times A_t^1.$$

Let us define π^1 such that for any stage t, $(\iota_t, a_t^1) \in \mathcal{I}_t \times A_t^1$:

$$\pi_t^1(a_t^1 \mid \iota_t) = \begin{cases} \dfrac{y^1(t, \iota_t, a_t^1)}{\sum_{\tilde{a}_t^1} y^1(t, \iota_t, \tilde{a}_t^1)} & \text{if } \sum_{\tilde{a}_t^1} y^1(t, \iota_t, \tilde{a}_t^1) > 0, \\ \delta_t(a_t^1) & \text{otherwise,} \end{cases}$$

for an arbitrary $\delta_t \in \Delta(A_t^1)$.

Then, the following are true:

1. *π^1 is a valid behavioral policy of Player 1*
2. *For all t, $y^1(t, \iota_t, a_t^1)$ is the occupancy measure of π^1, i.e. it is the joint distribution over (ι_t, a_t^1) induced by following policy π^1 equals:*

$$\mathbb{P}_{\pi^1}(\iota_t, a_t^1) = y^1(t, \iota_t, a_t^1), \quad \forall \iota_t \in \mathcal{I}_t,\ a_t^1 \in A_t^1.$$

Proof. We first prove that π^1 is a Player 1 behavioral policy. For any stage t and $\iota_t \in \mathcal{I}_t$, we show that $\pi_t^1(\cdot \mid \iota_t) \in \Delta(A_t^1)$. If $\sum_{a_t^1} y^1(t, \iota_t, a_t^1) = 0$, then by definition $\pi_t^1(\cdot \mid \iota_t) \in \Delta(A_t^1)$. Else, we have from (L3) and definition of π^1, $\pi_t^1(a_t^1 \mid \iota_t) > 0$ for all $a_t^1 \in A_t^1$. Furthermore, we have:

$$\sum_{a_t^1} \pi_t^1(a_t^1 \mid \iota_t) = \sum_{a_t^1} \frac{y^1(t, \iota_t, a_t^1)}{\sum_{\tilde{a}_t^1} y^1(t, \iota_t, \tilde{a}_t^1)} = 1.$$

As such, $\pi_t^1(\cdot \mid \iota_t) \in \Delta(A_t^1)$. Now, we prove by induction on t that the joint distribution over (ι_t, a_t^1) induced by following π^1 is exactly $y^1(t, \iota_t, a_t^1)$.

Base case: At $t = 0$, from (L1) we have:

$$\sum_{a_0^1} y^1(0, \iota_0, a_0^1) = \mathbb{P}(\iota_0).$$

If $\mathbb{P}(\iota_0) > 0$, we define:

$$\pi_0^1(a_0^1 \mid \iota_0) = \frac{y^1(0, \iota_0, a_0^1)}{\mathbb{P}(\iota_0)},$$

so:

$$\mathbb{P}_{\pi^1}(\iota_0, a_0^1) = y^1(0, \iota_0, a_0^1).$$

If $\mathbb{P}(\iota_0) = 0$, then $y^1(0, \iota_0, a_0^1) = 0$ for all a_0^1, and π_0^1 yields:

$$\mathbb{P}_{\pi^1}(\iota_0, a_0^1) = {\pi^1}_0(a_0^1 \mid \iota_0) \cdot 0 = 0 = y^1(0, \iota_0, a_0^1).$$

Inductive step: Suppose that for some $t \in [\![0, H-1]\!]$,

$$\mathbb{P}_{\pi^1}(\iota_t, a_t^1) = y^1(t, \iota_t, a_t^1), \quad \forall (\iota_t, a_t^1) \in \mathcal{I}_t \times A_t^1.$$

We now compute the joint distribution at time $t+1$:

$$\begin{aligned}\mathbb{P}_{\pi^1}(\iota_{t+1}, a^1_{t+1}) &= {\pi^1}_{t+1}(a^1_{t+1} \mid \iota_{t+1}) \cdot \mathbb{P}_{\pi^1}(\iota_{t+1}) \\ &= {\pi^1}_{t+1}(a^1_{t+1} \mid \iota_{t+1}) \cdot \sum_{\iota_t, a^1_t} \mathbb{P}(\iota_{t+1} \mid \iota_t, a^1_t) \cdot \mathbb{P}_{\pi^1}(\iota_t, a^1_t) \\ &= {\pi^1}_{t+1}(a^1_{t+1} \mid \iota_{t+1}) \cdot \sum_{\iota_t, a^1_t} \mathbb{P}(\iota_{t+1} \mid \iota_t, a^1_t) \cdot y^1(t, \iota_t, a^1_t).\end{aligned}$$

From (L2) we get:

$$\mathbb{P}_{\pi^1}(\iota_{t+1}, a^1_{t+1}) = {\pi^1}_{t+1}(a^1_{t+1} \mid \iota_{t+1}) \cdot \sum_{a^1_{t+1}} y^1(t+1, \iota_{t+1}, a^1_{t+1}).$$

If $\sum_{a^1_{t+1}} y^1(t+1, \iota_{t+1}, a^1_{t+1}) > 0$, then by definition of π^1:

$${\pi^1}_{t+1}(a^1_{t+1} \mid \iota_{t+1}) \cdot \sum_{a^1_{t+1}} y^1(t+1, \iota_{t+1}, a^1_{t+1}) = y^1(t+1, \iota_{t+1}, a^1_{t+1}).$$

Thus:

$$\mathbb{P}_{\pi^1}(\iota_{t+1}, a^1_{t+1}) = y^1(t+1, \iota_{t+1}, a^1_{t+1}).$$

If instead $\sum_{a_{t+1}} y^1(t+1, \iota_{t+1}, a^1_{t+1}) = 0$, then $\mathbb{P}_{\pi^1}(\iota_{t+1}, a^1_{t+1}) = 0$. From nonnegativity constraint (L3), we also have $y^1(t+1, \iota_{t+1}, a^1_{t+1}) = 0$ for all a^1_{t+1}. Thus:

$$\mathbb{P}_{\pi^1}(\iota_{t+1}, a^1_{t+1}) = 0 = y^1(t+1, \iota_{t+1}, a^1_{t+1}).$$

As such, we prove that $\forall t$, $\mathbb{P}_{\pi^1}(\iota_t, a^1_t) = y^1(t, \iota_t, a^1_t)$.

7.2 Proof of Theorem 1

Let y^{1*} be an optimal solution of the linear program (P1). We construct π^{1*} from y^{1*} via:

$$\pi^{1*}_t(a^1_t \mid s_t, o_{0:t-1}) = \begin{cases} \dfrac{y^{1*}(t, s_t, o_{0:t-1}, a^1_t)}{\sum_{\tilde{a}^1_t} y^{1*}(t, s_t, o_{0:t-1}, \tilde{a}^1_t)} & \text{if } \sum_{\tilde{a}^1_t} y^{1*}(t, s_t, o_{0:t-1}, \tilde{a}^1_t) > 0, \\ \delta_t(a^1_t) & \text{otherwise,} \end{cases}$$

where $\delta_t \in \Delta(A^1_t)$ is an arbitrary distribution.

We first verify that π^{1*} is a valid behavioral strategy. We apply Lemma 1 with information state $\iota_t = (s_t, o_{0:t-1})$. In this case:

$$\begin{aligned}\mathbb{P}(\iota_{t+1} \mid \iota_t, a^1_t) &= \mathbb{P}(s_{t+1}, o_{0:t} \mid s_t, o_{0:t-1}, a^1_t) \\ &= \mathbb{P}(s_{t+1} \mid s_t, o_{0:t-1}, a^1_t) \cdot \mathbb{P}(o_{0:t} \mid s_{t+1}, s_t, o_{0:t-1}, a^1_t) \\ &= p_t(s_{t+1} \mid s_t, a^1_t) \cdot \mathbb{I}[o_t = f^{\text{obs}}_t(s_t, a^1_t)].\end{aligned}$$

As such the LP constraints C1–C4 satisfy the hypotheses (L1)–(L3) of the lemma, and we conclude that:

– π^{1*} is a valid behavioral strategy of Player 1;
– y^{1*} is the occupancy measure of π^{1*}

Hence, by definition of the LP objective:

$$y^{1*} = \arg\max_{y^1} \hat{J}(y^1, \pi^2),$$

which implies $y^{1*} \in \mathbf{BR}(\pi^2)$.

7.3 Proof of Theorem 2

Let π^{2*} be an optimal solution of the linear program (P2). The constraints (C5) and (C6) ensure that π^{2*} is a valid policy of Player 2. Furthermore, the objective function of the linear program is equal to the expected cumulated reward $\hat{J}(y^1, \pi^{2*})$ as such:

$$\pi^{2*} = \arg\min_{\pi^2} \hat{J}(y^1, \pi^2) = \hat{\mathbf{BR}}_2(y^1)$$

7.4 Proof of Theorem 4

Let $\tilde{y}^{1*}$ be an optimal solution of the linear program (P3). We construct $\tilde{\pi}^{1*}$ from $\tilde{y}^{1*}$ via:

$$\tilde{\pi}_t^{1*}(a_t^1 \mid s_t, z_{t-1}) = \begin{cases} \dfrac{\tilde{y}^{1*}(t, s_t, z_{t-1}, a_t^1)}{\sum_{\tilde{a}_t^1} \tilde{y}^{1*}(t, s_t, z_{t-1}, \tilde{a}_t^1)} & \text{if } \sum_{\tilde{a}_t^1} \tilde{y}^{1*}(t, s_t, z_{t-1}, \tilde{a}_t^1) > 0, \\ \delta_t(a_t^1) & \text{otherwise,} \end{cases}$$

where $\delta_t \in \Delta(A_t^1)$ is an arbitrary distribution.

We now verify that $\tilde{\pi}^{1*}$ is a valid φ-compressed behavioral strategy. We apply Lemma 1 with information state $\iota_t = (s_t, z_{t-1})$ and occupancy function $y^1(t, \iota_t, a_t) = y^{1*}(t, s_t, z_{t-1}, a_t^1)$. In this case:

$$\begin{aligned} \mathbb{P}(\iota_{t+1} \mid \iota_t, a_t) &= \mathbb{P}(s_{t+1}, z_{t+1} \mid s_t, z_t, a_t^1) \\ &= \mathbb{P}(s_{t+1} \mid s_t, z_t, a_t^1) \cdot \mathbb{P}(z_{t+1} \mid s_{t+1}, s_t, z_t, a_t^1) \\ &= p_t(s_{t+1} \mid s_t, a_t^1) \cdot \mathbb{I}[\varphi_t(z_{t-1}, f_t^{\text{obs}}(a_t^1)) = z_t]. \end{aligned}$$

As such the LP constraints C7–C8 satisfy the hypotheses (L1)–(L3) of the lemma, and we conclude that:

– $\tilde{\pi}^{1*}$ is a valid φ-compressed behavioral strategy of Player 1;
– $\tilde{y}^{1*}$ is the occupancy measure of $\tilde{\pi}^{1*}$

Hence, by definition of the LP objective:

$$\tilde{y}^{1*} = \arg\max_{\tilde{y}^1} \hat{J}(\tilde{y}^1, \tilde{\pi}^2),$$

which implies $\tilde{y}^{1*} \in \hat{\mathbf{BR}}(\tilde{\pi}^2)$.

7.5 Proof of Theorem 3 (Convergence of Fictitious Play)

This proof is based on the work of Hofbauer and Sorin ([5]). We first prove that the sets of occupancy measures of Player 1 policies and Player 2 policies $\mathcal{Y}^1$ and Π^2 are compact convex sets, and that $\hat{J}$ is a saddle function, i.e. convex in y^1 and concave in π^2. We will then apply the main result to prove the theorem.

Proposition 1 (Convexity of variable sets). *$\mathcal{Y}^1$ and Π^2 are both compact convex sets.*

Proof. The set $\mathcal{Y}^1 \subset \prod_t \Delta(S_t \times \mathcal{H}_t \times A_t^1)$ a subset to a product of simplex sets and defined by linear constraints C1-C4. As such, it is itself compact convex. Similarly, the set $\Pi^2 = \prod_t \Delta(A_t^2)^{\mathcal{H}_t}$ is a product of simplex sets. This makes both set compact convex.

Proposition 2 (Saddle function property of LP objective function). *Let $\hat{J}$ be the function expressing the expected cumulated payoff for Player 2 policy π^2 and any occupancy measure of Player 1 y^1. $\hat{J}$ is a saddle function: concave in y^1 for fixed π^2, and convex in π^2 for fixed y^1.*

Proof. $\hat{J}$ bilinear in (y^1, π^2). As such, for fixed π^2, $\hat{J}$ is linear in y^1, hence concave. For fixed y^1, $\hat{J}$ is linear in π^2, hence convex.

As such, the propositions 1 and 2 satisfy the hypothesis and allow to apply the main theorem in [5]. Defining v as:

$$v(\tau) = \left[\max_{y^1} \hat{J}(y^1, \pi^2(\tau))\right] - \left[\min_{\pi^2} \hat{J}(y^1(\tau), \pi^2)\right]$$

we get result:

$$v(\tau) \leq v(0)e^{-\tau}$$