Proceedings of Machine Learning Research vol vvv:1–12, 2024

HSVI-based Online Minimax Strategies for Partially Observable Stochastic Games with Neural Perception Mechanisms

Rui Yan[1] RUI.YAN@CS.OX.AC.UK
Gabriel Santos[1] GABRIEL.SANTOS@CS.OX.AC.UK
Gethin Norman[1,2] GETHIN.NORMAN@GLASGOW.AC.UK
David Parker[1] DAVID.PARKER@CS.OX.AC.UK
Marta Kwiatkowska[1] MARTA.KWIATKOWSKA@CS.OX.AC.UK
[1] *Department of Computer Science, University of Oxford, Oxford, OX1 3QD, UK*
[2] *School of Computing Science, University of Glasgow, Glasgow, G12 8QQ, UK*

Abstract

We consider a variant of continuous-state partially-observable stochastic games with neural perception mechanisms and an asymmetric information structure. One agent has partial information, with the observation function implemented as a neural network, while the other agent is assumed to have full knowledge of the state. We present, for the first time, an efficient online method to compute an ε-minimax strategy profile, which requires only one linear program to be solved for each agent at every stage, instead of a complex estimation of opponent counterfactual values. For the partially-informed agent, we propose a continual resolving approach which uses lower bounds, pre-computed offline with heuristic search value iteration (HSVI), instead of opponent counterfactual values. This inherits the soundness of continual resolving at the cost of pre-computing the bound. For the fully-informed agent, we propose an inferred-belief strategy, where the agent maintains an inferred belief about the belief of the partially-informed agent based on (offline) upper bounds from HSVI, guaranteeing ε-distance to the value of the game at the initial belief known to both agents.

Keywords: Minimax strategies, continual resolving, partially observable stochastic games.

1. Introduction

Partially-observable stochastic games (POSGs) are a modelling formalism that enables strategic reasoning and (near-)optimal synthesis of strategies and equilibria in multi-agent settings with partial observations and uncertainty. *One-sided POSGs* (Horák et al., 2023) are a tractable subclass of two-agent, zero-sum POSGs with an asymmetric information structure, where only one agent has partial information while the other agent is assumed to have full knowledge. This is well suited to autonomous safety- or security-critical settings, such as patrolling or pursuit-evasion games, which require reasoning about worst-case assumptions. Since real-world settings increasingly often utilise neural networks (NNs) for perception tasks such as localisation and object detection, *one-sided neuro-symbolic POSGs (one-sided NS-POSGs)* were introduced (Yan et al., 2023). In this model the agent with partial information observes the environment only through a trained NN classifier, and consequently the game is generalised to *continuous environments*, to align with NN semantics, while observations remain discrete. A point-based NS-HSVI method was developed to approximate values of one-sided NS-POSGs working with (polyhedral decompositions of) the continuous space.

Strategy synthesis for continuous games is more challenging than for the finite-state case (Horák et al., 2023), since continuous-state spaces lead to an infinite number of strategies and discretisation suffers from the curse of dimensionality. Several *offline* methods exist, based on counterfactual regret minimisation, heuristics or reinforcement learning (see Related Work, below). In this paper, we consider *online* methods, which can improve efficiency and adaptability. The best performing online method (Moravčík et al., 2017) continually resolves a local strategy that only keeps track of the agent's belief of its opponent state and a vector of opponent counterfactual values. Apart from Horák et al. (2023), existing continual resolving approaches (Moravčík et al., 2017; Šustr et al., 2019; Schmid et al., 2023) are for extensive form games (EFGs), and cannot be directly applied to POSGs. This is because, although the two formalisms are connected (Kovařík et al., 2022), transitioning between them is not straightforward.

Contributions. We develop a *continual resolving* approach for one-sided NS-POSGs, addressing several challenges. Firstly, existing continual resolving approaches need to estimate the opponent's counterfactual values by solving a subgame at each stage (Moravčík et al., 2017), which would be intractable for continuous games. Instead, for the agent with partial observation (Ag_1), we use the lower bound computed offline by NS-HSVI (Yan et al., 2023), giving a polyhedral bound without solving a subgame. At each stage, we solve a linear program (LP), whose size is linear in the number of states in the current belief rather than the number of states reached, to compute the agent's action choice and update the lower bound. Thus, a *stage strategy* is computed online, in the spirit of continual resolving, for each situation as it arises during execution, instead of storing a complete strategy. Although we require offline computation for NS-HSVI, existing continual resolving approaches need to train deep counterfactual value networks to solve the subgame. Importantly, our NS-HSVI continual resolving does not lose soundness, i.e., ε-exploitability (Burch et al., 2014).

We can use any synthesis method for fully-observable stochastic games to generate an ε-minimax strategy for the fully-informed agent (Ag_2) (Yan et al., 2022). However, solving the fully-observable case would generate a *complete* strategy, which can be costly in terms of memory. We instead propose an online *inferred-belief* strategy by observing that, by using the offline upper bound from NS-HSVI, Ag_2 only needs to keep track of an inferred belief about Ag_1's belief and solve an LP, linear in the number of states in the current belief, to synthesise an action and the next inferred-belief of Ag_2. Since Ag_2 is fully informed, it does not need to store its belief. This allows us to generate a simpler strategy than the complete strategy for Ag_2, which guarantees the value at the initial belief, known to both agents, but cannot optimally employ the suboptimal actions of Ag_1 during play.

Summarising the contribution, we present, for the first time, an efficient online method to compute an ε-minimax strategy profile for one-sided NS-POSGs, a variant of two-player continuous-state POSGs with neural perception mechanisms, by exploiting bounds pre-computed offline by a variant of HSVI. We implement our approach, evaluate it on a pursuit-evasion model inspired by mobile robotics and investigate the synthesised agent strategies.

Related Work. Existing offline strategy synthesis methods for *one-sided* POSGs include a space partition approach (Zheng et al., 2022), a point-based approximate algorithm for continuous observations (Zheng et al., 2023), projection to POMDPs based on factored representations (Carr et al., 2021) and HSVI algorithms for finite (Horák et al., 2023) and continuous-state spaces (Yan et al., 2023). Since POSGs and EFGs are connected through factored-observation stochastic games (Kovařík et al., 2022), we next review relevant methods for two-agent zero-sum EFGs.

Counterfactual Regret Minimization (CFR) (Zinkevich et al., 2007) exploits the fact that the time-averaged strategy profile of regret minimizing algorithms converges to an ε-minimax strategy profile, in two-agent zero-sum EFGs with imperfect information. Since its introduction, a variety of CFR variants have been proposed and successfully applied to games (Lanctot et al., 2009; Lisỳ et al., 2015; Burch et al., 2014). However, since the CFR-based approaches require iterative traversal of the game tree, they become intractable when the tree is large. Additionally, these approaches are *offline* algorithms, returning a complete solution strategy that is difficult to represent and store.

A number of algorithms based on game-theoretic learning models such as reinforcement learning and heuristic search have also been proposed, which are able to compute strategies for two-agent zero-sum games with imperfect information, including (Bosansky et al., 2014; Heinrich et al., 2015; Lanctot et al., 2017; McAleer et al., 2021) and heuristic search value iteration (HSVI) (Yan et al., 2023; Delage et al., 2023), which we utilise in our work. However, these approaches are also offline algorithms and unable to refine strategies at test time.

The most related approach is *continual resolving* used in Horák et al. (2023), which is based on DeepStack (Moravčík et al., 2017), although other variants have also been proposed, e.g., (Šustr et al., 2019; Schmid et al., 2023; Brown et al., 2020). Both Horák et al. (2023) and our work are variants of continual resolving for one-sided POSGs, except we consider continuous-state spaces. Under the belief update in Horák et al. (2023), the current state could be missed and the belief might be empty. We assume a uniform stage strategy to fix this issue and ensure that the true state is always in the current belief. The LP size in Horák et al. (2023) is fixed, while the LP in our case varies at each stage because of no prior enumeration of all reachable states. Horák et al. (2023) performs the belief update before Ag_1 taking action and observing, whereas we update the belief using the action and the next observation, which results in a more accurate belief.

2. Background

We briefly review the model of Yan et al. (2023), which generalises *one-sided POSGs* (Horák et al., 2023) to continuous-state spaces and allows neural perception mechanisms. Let $\mathbb{P}(X)$ and $\mathbb{F}(X)$ denote the spaces of probability measures and functions on a Borel space X, respectively.

One-sided NS-POSGs. A *one-sided neuro-symbolic POSG (NS-POSG)* C is a two-player zero-sum infinite-horizon game with discrete actions and observations, where one player (Ag_1) is partially informed and the other (Ag_2) is fully informed. Unlike Horák et al. (2023), the game is played in a closed continuous environment S_E, which Ag_1 perceives only using perception function obs_1 given as a (trained) ReLU NN classifier that maps environment states to so called *percepts*, ranging over a finite set Per_1. The use of classifiers is aligned with, e.g., object detection or vision tasks in autonomous systems. We further assume that Ag_1 has a discrete local state space Loc_1, which is observable to both agents, and that Ag_2 has full knowledge of the environment's state.

A game C comprises agents $\mathsf{Ag}_1 = (S_1, A_1, obs_1, \delta_1)$, $\mathsf{Ag}_2 = (A_2)$ and environment $E = (S_E, \delta_E)$, where $S_1 = Loc_1 \times Per_1$; $A = A_1 \times A_2$ are joint actions; $obs_1 : (Loc_1 \times S_E) \to Per_1$ is Ag_1's perception function (note that we allow NNs to additionally depend on local states); $\delta_1 : (S_1 \times A) \to \mathbb{P}(Loc_1)$ is Ag_1's local transition function; and $\delta_E : (Loc_1 \times S_E \times A) \to \mathbb{P}(S_E)$ is E's finite-branching transition function. We work in the *belief space* $S_B \subseteq \mathbb{P}(S)$, where $S = S_1 \times S_E$, and assume an initial belief b^{init} using the particle-based representation (Porta et al., 2006; Doucet et al., 2001). A belief of Ag_1 is given by $b = (s_1, b_1)$, where $s_1 \in S_1$, $b_1 \in \mathbb{P}(S_E)$ and b_1 is represented by a weighted particle set $\{(s_E^i, \kappa_i)\}_{i=1}^{N_b}$ where $\kappa_i \geq 0$ and $\sum_{i=1}^{N_b} \kappa_i = 1$.

The game starts in a state $s = (s_1, s_E)$, where $s_1 = (loc_1, per_1) \in S_1$, and s is sampled from b^{init}. At each *stage* of the game, both agents concurrently choose one of their actions. If $a = (a_1, a_2) \in A$ is chosen, the local state loc_1 of Ag_1 is updated to $loc_1' \in Loc_1$ via $\delta_1(s_1, a)$, while the environment updates its state to $s_E' \in S_E$ via $\delta_E(loc_1, s_E, a)$. Finally, Ag_1, based on loc_1', generates its percept $per'_1 = obs_1(loc'_1, s'_E)$ at s'_E and C reaches the state $s' = ((loc'_1, per'_1), s'_E)$. The probability of transitioning from s to s' under a is $\delta(s,a)(s') = \delta_1(s_1,a)(loc'_1)\delta_E(loc_1, s_E, a)(s'_E)$.

Strategies. We distinguish between a *history* π (a sequence of states and joint actions, where $\pi(k)$ is the $(k{+}1)$th state, and $\pi[k]$ is the $(k{+}1)$th action) and a (local) *action-observation history (AOH)* for Ag_i (a sequence of its observations and actions). For the fully-informed Ag_2, an AOH is a history. A *strategy* of Ag_i is a mapping $\sigma_i : FPaths_{\mathsf{C},i} \to \mathbb{P}(A_i)$, where $FPaths_{\mathsf{C},i}$ is the set of Ag_i's finite AOHs. A *(strategy) profile* $\sigma = (\sigma_1, \sigma_2)$ is a pair of strategies and we denote by Σ_i and Σ the sets of strategies of Ag_i and profiles. The choices for the players after a history π are given by *stage strategies*: for Ag_1 this is a distribution $u_1 \in \mathbb{P}(A_1)$ and for Ag_2 a function $u_2 : S \to \mathbb{P}(A_2)$, i.e., $u_2 \in \mathbb{P}(A_2 \mid S)$. Given a belief (s_1, b_1), if Ag_1 chooses a_1, *assumes* Ag_2 chooses u_2 and observes s'_1, then the updated belief of Ag_1 via Bayesian inference is denoted $(s'_1, b_1^{s_1,a_1,u_2,s'_1})$.

Objectives and values. We focus on infinite-horizon expected accumulated reward $\mathbb{E}_b^{\sigma}[Y]$ when starting from b under σ, where $Y(\pi) = \sum_{k=0}^{\infty} \beta^k r(\pi(k), \pi[k])$ for an infinite history π, reward structure $r : (S{\times}A) \to \mathbb{R}$ and discount $\beta \in (0,1)$, and Ag_1 and Ag_2 maximise and minimise the expected value. Given $\varepsilon{\geq}0$, a profile $\sigma^\star = (\sigma_1^\star, \sigma_2^\star)$ is an *ε-minimax strategy profile* if for any $b \in S_B$, $\mathbb{E}_b^{\sigma^\star}[Y] \leq \mathbb{E}_b^{\sigma_1^\star,\sigma_2}[Y]{+}\varepsilon$ for all σ_2 and $\mathbb{E}_b^{\sigma^\star}[Y] \geq \mathbb{E}_b^{\sigma_1,\sigma_2^\star}[Y]{-}\varepsilon$ for all σ_1. If $\varepsilon{=}0$, then $\mathbb{E}_b^{\sigma^\star}[Y]$ is the *value* of C, denoted $V^\star$.

One-sided NS-HSVI. HSVI is an anytime algorithm that approximates the value $V^\star$ via *lower* and *upper bound* functions, updated through heuristically generated beliefs. One-sided NS-HSVI (Yan et al., 2023) works with the continuous-state space of a one-sided NS-POSG using a generalisation of α-functions, similar to Porta et al. (2006), except it uses *polyhedral* representations induced from NNs instead of Gaussian mixtures. For $\varepsilon > 0$, one-sided NS-HSVI returns lower and upper bound functions $V_{lb}^{\Gamma}, V_{ub}^{\Upsilon} \in \mathbb{F}(S_B)$ to approximate $V^\star$ such that $V_{lb}^{\Gamma}(b) \leq V^\star(b) \leq V_{ub}^{\Upsilon}(b)$ for all $b \in S_B$ and $V_{ub}^{\Upsilon}(b^{init}) - V_{lb}^{\Gamma}(b^{init}) \leq \varepsilon$. Given $f : S \to \mathbb{R}$ and belief (s_1, b_1), let $\langle f, (s_1, b_1)\rangle = \int_{s_E \in S_E} f(s_1, s_E) b_1(s_E) \mathrm{d}s_E$. The lower bound V_{lb}^{Γ} is represented via a finite set $\Gamma \subseteq \mathbb{F}(S)$ of *piecewise-constant (PWC)* α-functions such that $V_{lb}^{\Gamma}(s_1, b_1) = \max_{\alpha \in \Gamma} \langle \alpha, (s_1, b_1)\rangle$. The upper bound V_{ub}^{Υ} is represented by a finite set of belief-value pairs $\Upsilon \subseteq S_B {\times} \mathbb{R}$ and computed via an LP.

3. NS-HVSI Continual resolving

Continual resolving, e.g., (Moravčík et al., 2017), is an online method for computing an ε-minimax strategy in two-player, zero-sum imperfect information EFGs; it keeps track of an agent's belief of its opponent state and opponent counterfactual values to build and solve a subgame to synthesise choices, without building a complete strategy. It is *sound*, in computing an ε-minimax strategy, but can be expensive as it needs to estimate opponent counterfactual values by traversing the game tree.

We now present a novel variant of continual resolving, which utilises the *lower bound* function V_{lb}^{Γ} computed by one-sided NS-HSVI to synthesise an ε-minimax strategy for Ag_1 that achieves the desired ε distance to the value function at the initial belief. The method is efficient as it only requires solving a single LP at each stage. We first introduce the following minimax operator.

Definition 1 (Minimax) *The minimax operator $T : \mathbb{F}(S_B) {\to} \mathbb{F}(S_B)$ is given by:*

$$\begin{aligned}[TV](s_1, b_1) = {} & \max\nolimits_{u_1 \in \mathbb{P}(A_1)} \min\nolimits_{u_2 \in \mathbb{P}(A_2 \mid S)} \mathbb{E}_{(s_1,b_1),u_1,u_2}[r(s,a)] \\ & + \beta \textstyle\sum_{(a_1,s'_1) \in A_1 \times S_1} P(a_1, s'_1 \mid (s_1, b_1), u_1, u_2) V(s'_1, b_1^{s_1,a_1,u_2,s'_1})\end{aligned} \tag{1}$$

Algorithm 1 NS-HSVI continual resolving for Ag_1's strategy via the lower bound

```
Input: $(s_1^{init}, b_1^{init})$, a finite set of PWC functions $\Gamma \subseteq \mathbb{F}(S)$ from one-sided NS-HSVI
$Resolve_1((s_1^{init}, b_1^{init}), \alpha^{init})$ where $\alpha^{init} = \arg\max_{\alpha\in\Gamma}\langle \alpha, (s_1^{init}, b_1^{init})\rangle$
function $Resolve_1((s_1, b_1), \alpha_1)$
    $(\overline{v}^\star, \overline{\lambda}_1^\star, \overline{p}_1^\star) \leftarrow$ solve the LP (2) at $(s_1, b_1)$
    $u_1^{lb}(a_1) \leftarrow p^{\star a_1}$ for all $a_1 \in A_1$
    sample and play $a_1 \sim u_1^{lb}$
    $s_1' \leftarrow$ observed $\mathsf{Ag}_1$'s agent state
    $\alpha^{\star a_1, s_1'} \leftarrow \sum_{\alpha\in\Gamma}(\lambda_\alpha^{\star a_1, s_1'}/p^{\star a_1})\alpha$,    $u_2^{lb} \leftarrow$ an assumed stage strategy for $\mathsf{Ag}_2$
    $Resolve_1((s_1', b_1^{s_1, a_1, u_2^{lb}, s_1'}), \alpha^{\star a_1, s_1'})$
```

for $V \in \mathbb{F}(S_B)$ and $(s_1, b_1) \in S_B$, where $\mathbb{E}_{(s_1,b_1),u_1,u_2}[r(s,a)]$ is the expected value of r.

NS-HSVI continual resolving. Motivated by (Horák et al., 2023, Section 9.2), our online game-playing algorithm *NS-HSVI continual resolving*, see Algorithm 1, generates a strategy for Ag_1, denoted σ_1^{lb}, by using the HSVI lower bound instead of opponent counterfactual values used in Moravčík et al. (2017). Since we have pre-computed V_{lb}^{Γ} offline, our NS-HSVI continual resolving only keeps track of a belief (s_1, b_1) and a PWC function α_1 in the the convex hull, $\mathrm{Conv}(\Gamma)$, of Γ. Importantly, Γ can be used to compute an action to play and update the tracking information at each stage (a belief and a PWC function) by solving the LP presented below, thus avoiding the need to estimate the opponent counterfactual values.

Definition 2 (Stage strategy) *For $((s_1, b_1), \alpha_1) \in S_B \times \mathrm{Conv}(\Gamma)$ where b_1 is represented by $\{(s_E^i, \kappa_i)\}_{i=1}^{N_b}$, a stage strategy u_1^{lb} for Ag_1 is such that $u_1^{lb}(a_1) = p^{\star a_1}$ for $a_1 \in A_1$, where $(v_{s_E^i}^\star)_{i=1}^{N_b}$, $(\lambda_\alpha^{\star a_1, s_1'})_{(a_1, s_1')\in A_1\times S_1, \alpha\in\Gamma}$ and $(p^{\star a_1})_{a_1\in A_1}$ is a solution to the following LP:*

$$
\begin{aligned}
&\text{maximise } \textstyle\sum_{i=1}^{N_b} \kappa_i v_{s_E^i} \text{ subject to}\\
&v_{s_E^i} \le \textstyle\sum_{a_1\in A_1} p^{a_1} r((s_1, s_E^i), (a_1, a_2)) + \beta \textstyle\sum_{(a_1, s_1')\in A_1\times S_1, s_E'\in S_E}\\
&\qquad \cdot\, \delta((s_1, s_E^i), (a_1, a_2))(s_1', s_E') \textstyle\sum_{\alpha\in\Gamma} \lambda_\alpha^{a_1, s_1'} \alpha(s_1', s_E') \quad \text{for } 1 \le i \le N_b \text{ and } a_2 \in A_2\\
&v_{s_E^i} \ge \alpha_1(s_1, s_E^i) \quad \text{for } 1 \le i \le N_b\\
&\lambda_\alpha^{a_1, s_1'} \ge 0 \quad \text{for } a_1 \in A_1,\ s_1' \in S_1 \text{ and } \alpha \in \Gamma\\
&p^{a_1} = \textstyle\sum_{\alpha\in\Gamma} \lambda_\alpha^{a_1, s_1'} \quad \text{for } a_1 \in A_1 \text{ and } s_1' \in S_1\\
&\textstyle\sum_{a_1\in A_1} p^{a_1} = 1.
\end{aligned}
\tag{2}
$$

Compared with the LP in Yan et al. (2023) for solving $[TV_{lb}^{\Gamma}](s_1, b_1)$, the LP (2) includes the additional constraints $v_{s_E^i} \ge \alpha_1(s_1, s_E^i)$ for $1 \le i \le N_b$ (Horák et al., 2023, Section 9.2) to refine the minimax stage strategy in $[TV_{lb}^{\Gamma}](s_1, b_1)$, such that the lower bound from α_1 can be kept as the game evolves, since multiple minimax stage strategies for Ag_1 may exist and some of them may deviate from α_1.

We illustrate how the strategy σ_1^{lb} is obtained in Fig. 1 (left), where the red and orange circles indicate the current state, with the size of the interior solid circle representing the probability.

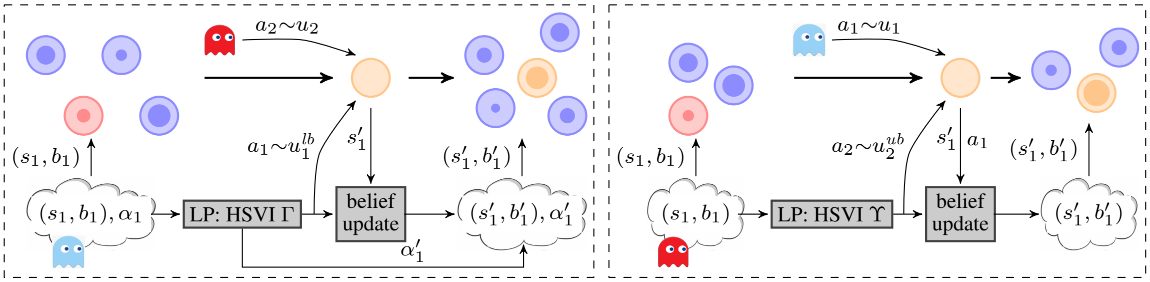

Figure 1: Left: NS-HSVI continual resolving for the partially-informed agent Ag_1 (blue). Right: inferred-belief strategy synthesis for the fully-informed agent Ag_2 (red).

When resolving the game locally $((s_1, b_1), \alpha_1)$, Ag_1 (blue) plays an action a_1 sampled from a stage strategy u_1^{lb} computed via (2). Simultaneously, Ag_2 (red) plays an action a_2 sampled from a stage strategy u_2, which is unknown to Ag_1. The game moves to the next state and consequently Ag_1 observes a new agent state $s_1' \in S_1$. Based on a_1, s_1' and an *assumed* stage strategy u_2^{lb} for Ag_2, see lines 7–8 of Algorithm 1, Ag_1 updates its belief to $(s_1', b_1^{s_1, a_1, u_2^{lb}, s_1'})$ via Bayesian inference, and generates $\alpha^{\star a_1, s_1'} \in \mathrm{Conv}(\Gamma)$ from a solution to (2), which forms a new pair for the next resolving.

Remark 1 *There are two key properties of Algorithm 1. First, LP (2) admits at least one solution as V_{lb}^{Γ} is computed by the one-sided NS-HSVI. Second, the current state has to be in the support of the current belief, i.e.,* Ag_1 *does not lose track of the current state. Such a belief is called a proper belief and this is ensured by assuming the uniform stage strategy for* Ag_2.

Lemma 1 (Existence and proper belief) *For the NS-HSVI continual resolving at $((s_1, b_1), \alpha_1)$, the LP (2) admits at least one solution, and if the current state is (s_1, s_E), then $b_1(s_E) > 0$.*

Proof The optimal value $V^{\star}$ has lower and upper bounds $L = \min_{s \in S, a \in A} r(s, a)/(1 - \beta)$ and $U = \max_{s \in S, a \in A} r(s, a)/(1 - \beta)$. Let $V_{lb}^{\Gamma'}$ be the lower bound of one-sided NS-HSVI (Yan et al., 2023). Existence follows if (2) admits one solution after any point-based update. The key is to check the feasibility of the first two constraints for $v_{s_E^i}$ of (2). Initially $\Gamma' = \{\alpha_0\}$ with $\alpha_0(s) = L$ for $s \in S$, and since $\alpha_1 = \alpha_0$ it is straightforward to verify that (2) admits at least one solution.

For the inductive step, we assume that (2) admits at least one solution for $\alpha \in \Gamma$ and $\alpha_1 \in \mathrm{Conv}(\Gamma)$. The point-based update computes a new set $\Gamma' = \Gamma \cup \{\alpha^{\star}\}$ of PWC functions, where (2) admits at least one solution for $\alpha_1 = \alpha^{\star}$. Thus, (2) admits at least one solution for $\alpha \in \Gamma$ and $\alpha_1 \in \mathrm{Conv}(\Gamma) \cup \{\alpha^{\star}\}$. Following the proof of (Horák et al., 2023, Lemma 9.6), we can show that (2) admits at least one solution for $\alpha \in \Gamma'$ and $\alpha_1 \in \mathrm{Conv}(\mathrm{Conv}(\Gamma) \cup \{\alpha^{\star}\})$, i.e., $\alpha_1 \in \mathrm{Conv}(\Gamma')$.

To show the belief is proper, since the initial state is sampled from (s_1^{init}, b_1^{init}), which is known to Ag_1, and the uniform stage strategy for Ag_2 is assumed, then the result follows. ■

We next show that our NS-HSVI continual resolving can inherit the soundness of continual resolving, i.e., it can compute an ε-minimax strategy for Ag_1.

Theorem 1 (ε-minimax strategy for Ag_1) *The strategy σ_1^{lb} in Algorithm 1 is an ε-minimax strategy for* Ag_1 *at (s_1^{init}, b_1^{init}), i.e., $\mathbb{E}_{(s_1^{init}, b_1^{init})}^{\sigma_1^{lb}, \sigma_2}[Y] \geq V^{\star}(s_1^{init}, b_1^{init}) - \varepsilon$ for all $\sigma_2 \in \Sigma_2$.*

Proof We adapt the proof presented for discrete one-sided POSGs in (Horák et al., 2023, Proposition 9.7). Let $\mathbb{E}_s^{\sigma}[Y]$ denote the expected value of Y when starting from $s \in S$ under $\sigma \in \Sigma$.

Consider any $b = (s_1, b_1) \in S_B$ and $\alpha_1 \in \mathrm{Conv}(\Gamma)$. We assume that Ag_1 follows $\mathit{Resolve}_1$ in Algorithm 1 at the first t stages and then follows the uniform stage strategy. We denote such a strategy by $\sigma_1^{b,\alpha_1,t}$. Lemma 1 guarantees that $\mathit{Resolve}_1$ can run for t stages. We next prove that, for any s_E with $b_1(s_E) > 0$, the expected value of Y from (s_1, s_E) under $\sigma_1^{b,\alpha_1,t}$ has the bound by α_1:

$$\mathbb{E}_{(s_1,s_E)}^{\sigma_1^{b,\alpha_1,t},\sigma_2}[Y] \geq \alpha_1(s_1, s_E) - \beta^t(U - L) \quad \text{for all } \sigma_2 \in \Sigma_2. \tag{3}$$

We prove (3) by induction on $t \in \mathbb{N}$. For t=0, (3) holds as U and L are trivial lower and upper bounds, and $\alpha_1 \in \mathrm{Conv}(\Gamma)$ and $\alpha(s) \leq U$ for all $s \in S$ and $\alpha \in \Gamma$. We assume (3) holds for the first t stages and any $b' = (s_1', b_1') \in S_B$. The strategy $\sigma_1^{b,\alpha_1,t+1}$ implies that, at (s_1, b_1), Ag_1 takes u_1^{lb} (line 4) and then follows the strategy $\sigma_1^{b',\alpha',t}$ if a_1 is taken and s_1' is observed, where $b' = (s_1', b_1')$, $b_1' = b_1^{s_1,a_1,u_2^{lb},s_1'}$ and $\alpha' = \alpha^{\star a_1,s_1'}$. Letting $u_2 \in \mathbb{P}(A_2 \mid S)$ be the stage strategy of Ag_2 at b given by σ_2, using $b_1(s_E) > 0$ by Lemma 1, the left side of (3) by replacing t with $t+1$ equals:

$$\begin{aligned}
&\mathbb{E}_{u_1^{lb},u_2}[r((s_1,s_E),a)] + \beta\textstyle\sum_{(a_1,a_2)\in A \wedge (s_1',s_E')\in S} u_1^{lb}(a_1)u_2(a_2 \mid s_1, s_E) \\
&\qquad \cdot \delta((s_1,s_E),(a_1,a_2))(s_1',s_E')\mathbb{E}_{(s_1',s_E')}^{\sigma_1^{b',\alpha',t},\sigma_2}[Y] && \text{by definition of } u_1^{lb},\ u_2 \text{ and } \delta \\
&\geq \min\nolimits_{a_2\in A_2} \Big(\textstyle\sum_{a_1\in A_1} u_1^{lb}(a_1)r((s_1,s_E),(a_1,a_2)) + \beta\sum_{a_1\in A_1 \wedge (s_1',s_E')\in S} u_1^{lb}(a_1) \\
&\qquad \cdot \delta((s_1,s_E),(a_1,a_2))(s_1',s_E')\mathbb{E}_{(s_1',s_E')}^{\sigma_1^{b',\alpha',t},\sigma_2}[Y]\Big) && \text{by linearity in } u_2 \\
&\geq \min\nolimits_{a_2\in A_2} \big(\textstyle\sum_{a_1\in A_1} u_1^{lb}(a_1)r((s_1,s_E),(a_1,a_2)) + \beta\sum_{a_1\in A_1 \wedge (s_1',s_E')\in S} u_1^{lb}(a_1) \\
&\qquad \cdot \delta((s_1,s_E),(a_1,a_2))(s_1',s_E')(\alpha^{\star a_1,s_1'}(s_1',s_E') - \beta^t(U-L))\big) && \text{by induction} \\
&= \min\nolimits_{a_2\in A_2} \big(\textstyle\sum_{a_1\in A_1} u_1^{lb}(a_1)r((s_1,s_E),(a_1,a_2)) + \beta\sum_{a_1\in A_1 \wedge (s_1',s_E')\in S} u_1^{lb}(a_1) \\
&\qquad \cdot \delta((s_1,s_E),(a_1,a_2))(s_1',s_E')\alpha^{\star a_1,s_1'}(s_1',s_E')\big) - \beta^{t+1}(U-L) && \text{rearranging} \\
&\geq v_{s_E}^\star - \beta^{t+1}(U-L) && \text{since } (\overline{v}^\star, \overline{\lambda}_1^\star, \overline{p}_1^\star) \text{ is a solution to (2) (first constraint)} \\
&\geq \alpha_1(s_1,s_E) - \beta^{t+1}(U-L) && \text{since } (\overline{v}^\star, \overline{\lambda}_1^\star, \overline{p}_1^\star) \text{ is a solution to (2) (second constraint)}
\end{aligned}$$

and hence (3) holds. Letting $\sigma_1^{lb} = \lim_{t\to\infty} \sigma_1^{b^{init},\alpha^{init},t}$ by definition:

$$\begin{aligned}
\mathbb{E}_{b^{init}}^{\sigma_1^{lb},\sigma_2}[Y] &= \textstyle\int_{s_E \in S_E} b_1^{init}(s_E)\mathbb{E}_{(s_1^{init},s_E)}^{\sigma_1^{lb},\sigma_2}[Y]\mathrm{d}s_E \\
&\geq \langle \alpha^{init}, (s_1^{init}, b_1^{init})\rangle && \text{by (3) and definition of } \langle\cdot,\cdot\rangle \\
&= V_{lb}^{\Gamma}(b^{init}) && \text{by line 1 of Algorithm 1} \\
&\geq V^\star(b^{init}) - \varepsilon && \text{since } V_{lb}^{\Gamma} \text{ is returned by one-sided NS-HSVI}
\end{aligned}$$

which completes the proof. ■

4. Inferred-Belief Strategy Synthesis

We complement our variant of continual resolving with strategy synthesis for Ag_2, which utilises the *upper bound* function V_{ub}^{Υ} pre-computed offline and keeps track of an *inferred* belief about what Ag_1 believes, which could differ from Ag_1's true belief. Any offline method for fully-observable stochastic games could instead be used, with the associated high computational and storage cost of

ALGORITHM 2 Inferred-belief strategy synthesis for Ag$_2$ via the upper bound

```
Input: $(s_1^{init}, b_1^{init})$, a finite set of belief-value pairs $\Upsilon$ by one-sided NS-HSVI
$Resolve_2(s_1^{init}, b_1^{init})$
function $Resolve_2(s_1, b_1)$
    $u_2^{ub} \leftarrow$ Ag$_2$'s minimax strategy in $[TV_{ub}^{\Upsilon}](s_1, b_1)$,    $(s_1, s_E) \leftarrow$ current observed state
    sample and play $a_2 \sim u_2^{ub}(\cdot \mid s_1, s_E)$
    $(a_1, s_1') \leftarrow$ Ag$_1$'s action and updated agent state
    $Resolve_2(s_1', b_1^{s_1,a_1,u_2^{ub},s_1'})$
```

generating a complete strategy. Instead, we present an efficient *online* algorithm, where only one LP is solved at each stage, to synthesise an ε-minimax strategy for Ag$_2$. Since we have pre-computed the offline upper bound, this strategy can guarantee the minimax value from the initial belief, which is known to both agents, but cannot optimally employ the suboptimal actions of Ag$_1$ during play.

Inferred-belief strategy synthesis. Our inferred-belief strategy synthesis for Ag$_2$ (Algorithm 2) returns a strategy σ_2^{ub} based on V_{ub}^{Υ}. The main idea of σ_2^{ub} is that Ag$_2$ keeps a belief (s_1, b_1), about Ag$_1$'s belief at the current stage, and then computes an action via $Resolve_2$ based on (s_1, b_1) and V_{ub}^{Υ}. This belief (s_1, b_1) is *inferred*, as Ag$_2$ has no access to what Ag$_1$ actually believes about the state, except the initial belief which is common knowledge. However, since Ag$_2$ is fully-informed, it can simulate an inferred belief update of Ag$_1$, i.e., its belief about Ag$_1$'s next belief.

We illustrate the obtained strategy σ_2^{ub} in Fig. 1 (right). If (s_1, b_1) is what Ag$_2$ believes about Ag$_1$'s belief and (s_1, s_E) is the current state observed by Ag$_2$, then Ag$_2$ chooses a_2 sampled from the stage strategy u_2^{ub} conditioned on (s_1, s_E) in $[TV_{ub}^{\Upsilon}](s_1, b_1)$ computed via an LP (Yan et al., 2023). At the same time, Ag$_1$ takes $a_1 \in A_1$ sampled from a stage strategy u_1, where Ag$_2$ does not know u_1. Then, the game moves to the next state and thus Ag$_1$ observes $s_1' \in S_1$. Based on a_1, s_1' and u_2^{ub}, Ag$_2$ updates its belief about Ag$_1$'s belief via Bayesian inference to $(s_1', b_1^{s_1,a_1,u_2^{ub},s_1'})$.

We next show that the inferred-belief strategy is sound, i.e., the inferred-belief carries enough information to generate an ε-minimax strategy for Ag$_2$.

Lemma 2 (Monotonicity) *If V_{ub}^{Υ} is an upper bound generated during the one-sided NS-HSVI, then $[TV_{ub}^{\Upsilon}](s_1, b_1) \leq V_{ub}^{\Upsilon}(s_1, b_1)$ for all $(s_1, b_1) \in S_B$.*

Proof For a given set Υ of belief-value pairs, we first show $[TV_{ub}^{\Upsilon}]$ is convex and continuous. From (Yan et al., 2023, Theorem 6 and 7) we have that $[TV_{ub}^{\Upsilon}](s_1, b_1) = \sup_{\alpha \in \Gamma^{\Upsilon}} \langle \alpha, (s_1, b_1) \rangle$, where $\Gamma^{\Upsilon} \subseteq \mathbb{F}(S)$ and $L \leq \alpha(s) \leq U$ for all $s \in S$ and $\alpha \in \Gamma^{\Upsilon}$. By (Horák et al., 2023, Proposition 4.9), it follows that $[TV_{ub}^{\Upsilon}](s_1, \cdot)$ is convex. For $b_1, b_1' \in \mathbb{P}(S_E)$ and $\alpha \in \Gamma^{\Upsilon}$, using (Yan et al., 2023, Theorem 1), we have $|\langle \alpha, (s_1, b_1) \rangle - \langle \alpha, (s_1, b_1') \rangle| \leq K_{ub}(b_1, b_1')$, where K_{ub} measures belief difference, and hence $|[TV_{ub}^{\Upsilon}](s_1, b_1) - [TV_{ub}^{\Upsilon}](s_1, b_1')| \leq K_{ub}(b_1, b_1')$.

Let $V_{ub}^{\Upsilon^t}$ and I^t be the upper bound and the associated index set after the t-th point-based update, respectively. Similarly to (Horák et al., 2023, Lemma 9.11), we can now prove the monotonicity of $[TV_{ub}^{\Upsilon^t}]$ by induction on $t \in \mathbb{N}$. For t=0, since $\Upsilon^0 = \{((s_1^i, b_1^i), U) \in S_B \times \mathbb{R} \mid i \in I^0\}$ for some initial index set I^0. For any $(s_1, b_1) \in S_B$, using (1), we have $[TV_{ub}^{\Upsilon^0}](s_1, b_1) \leq \max_{s \in S \wedge a \in A} r(s, a) + \beta U = (1 - \beta)U + \beta U = U = V_{ub}^{\Upsilon^0}(s_1, b_1)$.

For the inductive step, we assume that $[TV_{ub}^{\Upsilon^t}](s_1', b_1') \leq V_{ub}^{\Upsilon^t}(s_1', b_1')$ for all $(s_1', b_1') \in S_B$. Thus $y_i \geq V_{ub}^{\Upsilon^t}(s_1^i, b_1^i) \geq [TV_{ub}^{\Upsilon^t}](s_1^i, b_1^i)$ for $i \in I^t$. Let $(s_1, b_1) \in S_B$ be the belief for the

$(t{+}1)$-th point-based update. By lines 8 and 9 of (Yan et al., 2023, Algorithm 1), we have $y^\star = [TV_{ub}^{\Upsilon^t}](s_1, b_1)$ and $\Upsilon^{t+1} = \Upsilon^t \cup \{((s_1, b_1), y^\star)\}$. Using (Yan et al., 2023, Lemma 4), we have $V_{ub}^{\Upsilon^t}(s_1', b_1') \geq V_{ub}^{\Upsilon^{t+1}}(s_1', b_1')$ for all $(s_1', b_1') \in S_B$, from which $[TV_{ub}^{\Upsilon^t}](s_1^i, b_1^i) \geq [TV_{ub}^{\Upsilon^{t+1}}](s_1^i, b_1^i)$ for any $i \in I^{t+1}$. Therefore we have that $y^i \geq [TV_{ub}^{\Upsilon^t}](s_1^i, b_1^i) \geq [TV_{ub}^{\Upsilon^{t+1}}](s_1^i, b_1^i)$ for any $i \in I^{t+1}$. Now, for any $(s_1', b_1') \in S_B$, if $(\lambda_i^\star)_{i \in I_{s_1'}^{t+1}}$ is a solution for $V_{ub}^{\Upsilon^{t+1}}(s_1', b_1')$, then by construction:

$$
\begin{aligned}
& V_{ub}^{\Upsilon^{t+1}}(s_1', b_1') = \textstyle\sum_{i \in I_{s_1'}^{t+1}} \lambda_i^\star y_i + K_{ub}(b_1', \textstyle\sum_{i \in I_{s_1'}^{t+1}} \lambda_i^\star b_1^i) \\
& \geq \textstyle\sum_{i \in I_{s_1'}^{t+1}} \lambda_i^\star [TV_{ub}^{\Upsilon^{t+1}}](s_1^i, b_1^i) + K_{ub}(b_1', \textstyle\sum_{i \in I_{s_1'}^{t+1}} \lambda_i^\star b_1^i) && \text{by induction} \\
& \geq [TV_{ub}^{\Upsilon^{t+1}}](s_1', \textstyle\sum_{i \in I_{s_1'}^{t+1}} \lambda_i^\star b_1^i) + K_{ub}(b_1', \textstyle\sum_{i \in I_{s_1'}^{t+1}} \lambda_i^\star b_1^i) && \text{since } [TV_{ub}^{\Upsilon^{t+1}}] \text{ is convex} \\
& \geq [TV_{ub}^{\Upsilon^{t+1}}](s_1', b_1') && \text{since } [TV_{ub}^{\Upsilon^{t+1}}] \text{ is } K_{ub}\text{-continuous}
\end{aligned}
$$

and hence by induction $[TV_{ub}^{\Upsilon}]$ is monotone as required. ■

Theorem 2 (ε-minimax strategy for Ag_2) *The strategy σ_2^{ub} in Algorithm 2 is an ε-minimax strategy for Ag_2 at (s_1^{init}, b_1^{init}), i.e., $\mathbb{E}_{(s_1^{init}, b_1^{init})}^{\sigma_1, \sigma_2^{ub}}[Y] \leq V^\star(s_1^{init}, b_1^{init}) + \varepsilon$ for all $\sigma_1 \in \Sigma_1$.*

Proof Consider $b{=}(s_1, b_1) \in S_B$. We assume that Ag_2 follows $\mathit{Resolve}_2$ in Algorithm 2 for the first t stages and then follows the uniform strategy, and denote this strategy by $\sigma_2^{b,t}$. We prove by induction on $t \in \mathbb{N}$ that the expected value of Y from b under $\sigma_2^{b,t}$ has the following upper bound:

$$
\mathbb{E}_b^{\sigma_1, \sigma_2^{b,t}}[Y] \leq V_{ub}^{\Upsilon}(b) + \beta^t (U - L) \qquad \text{for all } \sigma_1 \in \Sigma_1. \tag{4}
$$

For $t = 0$, the strategy $\sigma_2^{b,0}$ implies that Ag_2 adopts the uniform strategy, and therefore (4) directly follows as U and L are lower and upper bounds.

For the inductive step, we assume (4) holds for the first t stages. The strategy $\sigma_2^{b,t+1}$ implies that at $b = (s_1, b_1)$, Ag_2 takes u_2^{ub} (line 4) and then if a_1 is taken and s_1' is observed, follows the strategy $\sigma_2^{b',t}$, where $b' = (s_1', b_1')$ and $b_1' = b_1^{s_1, a_1, u_2^{ub}, s_1'}$. Letting $u_1 \in \mathbb{P}(A_1)$ be Ag_1's stage strategy at b given by any σ_1, the left-hand side of (4) by replacing t with $t + 1$ equals:

$$
\begin{aligned}
& \mathbb{E}_{b, u_1, u_2^{ub}}[r(s,a)] + \beta \textstyle\sum_{(a_1, s_1') \in A_1 \times S_1} P(a_1, s_1' \mid b, u_1, u_2^{ub}) \mathbb{E}_{b'}^{\sigma_1, \sigma_2^{b',t}}[Y] \\
& \leq \mathbb{E}_{b, u_1, u_2^{ub}}[r(s,a)] + \beta \textstyle\sum_{(a_1, s_1') \in A_1 \times S_1} P(a_1, s_1' \mid b, u_1, u_2^{ub}) (V_{ub}^{\Upsilon}(b') + \beta^t (U - L)) && \text{by induction} \\
& \leq \mathbb{E}_{b, u_1^{ub}, u_2^{ub}}[r(s,a)] + \beta \textstyle\sum_{(a_1, s_1') \in A_1 \times S_1} P(a_1, s_1' \mid b, u_1^{ub}, u_2^{ub}) V_{ub}^{\Upsilon}(b') + \beta^{t+1} (U - L) \\
& && \text{rearranging and since } u_1^{ub} \text{ is a minimax strategy} \\
& = [TV_{ub}^{\Upsilon}](b) + \beta^{t+1}(U - L) && \text{by definition of } [TV_{ub}^{\Upsilon}] \\
& \leq V_{ub}^{\Upsilon}(b) + \beta^{t+1}(U - L) && \text{by Lemma 2}
\end{aligned}
$$

as required. Now, letting $\sigma_2^{ub} = \lim_{t \to \infty} \sigma_2^{b^{init}, t}$, from (4) we have:

$$
\mathbb{E}_{b^{init}}^{\sigma_1, \sigma_2^{ub}}[Y] \leq V_{ub}^{\Upsilon}(b^{init}) \leq V^\star(b^{init}) + \varepsilon
$$

where the last inequality follows from the fact that V_{ub}^{Υ} is returned by one-sided NS-HSVI. ■

Corollary 1 (ε-minimax strategy profile) *The profile $(\sigma_1^{lb}, \sigma_2^{ub})$ is an ε-minimax strategy profile.*

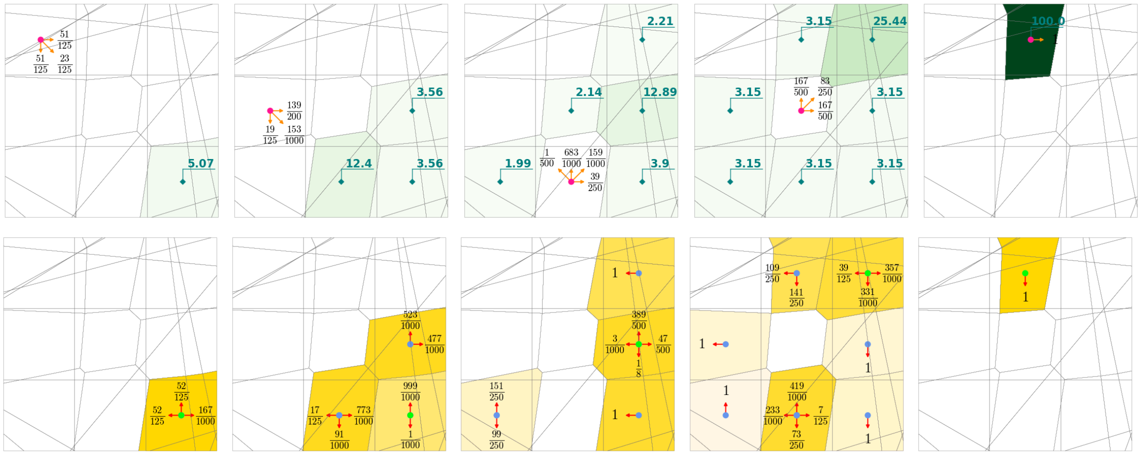

Figure 2: Snippets of a synthesised strategy for the pursuer and evader (from left to right).

5. Experiments

We evaluated our method on a variant of a *pursuit-evasion* game (Horák et al., 2023), inspired by mobile robotics (Chung et al., 2011; Isler and Karnad, 2008); see the appendix of Yan et al. (2023) for more detail. The game involves a *pursuer*, whose aim is to capture an *evader*. The pursuer is equipped with a ReLU NN classifier, which takes the location (coordinates) of the pursuer as input and outputs one of the 9 abstract grid cells, each consisting of multiple polytopes, with the initial decomposition obtained by computing the preimage of the NN (Matoba and Fleuret, 2020). The pursuer therefore observes which cell it is in, but not its exact location, and knows neither the exact location nor cell of the evader. The evader is fully informed and knows the exact locations of both agents. The evader is captured when both agents are in the same cell. We set a discount of 0.7, reward 100 for capture and timeout of $2h$. The (offline) lower and upper bounds for the value of the initial belief are 5.0699 and 6.0665, respectively. We synthesise strategies, which demonstrate that the pursuer can eventually capture the evader with positive probability. Fig. 2 shows stage strategies and lower bounds of states in the belief of the pursuer (top), and evader's strategies and inferred beliefs (bottom), at different stages. Lower bounds are coloured green and inferred beliefs yellow. The agent positions are highlighted (pink dot for pursuer and light green for evader). The belief of the pursuer and inferred-belief of the evader do not always coincide, e.g., in the third column, the state with bound 2.14 is in the pursuer's belief, but not the inferred belief. We observe that the pursuer's strategy selects the moves according to the magnitude of the bound, e.g., in the fourth column, the pursuer moves up or right, since the top right evader position has the highest bound.

6. Conclusions

We have developed an efficient online method to synthesise strategies for a variant of one-sided continuous-state POSGs with discrete observations and validated it on a pursuit-evasion game, in which the partially-informed agent uses a neural network for perception. We have shown that combining continual resolving, inferred beliefs and HSVI bounds computed offline can generate an ε-minimax strategy profile online. For future work, we will consider aggressive assumed stage strategies for the fully-informed agent, since uniform strategies may lead to a large number of states in the belief and consequently large LPs to solve.

Acknowledgements. This project was funded by the ERC under the European Union's Horizon 2020 research and innovation programme (FUN2MODEL, grant agreement No.834115).

References

Branislav Bosansky, Christopher Kiekintveld, Viliam Lisy, and Michal Pechoucek. An exact double-oracle algorithm for zero-sum extensive-form games with imperfect information. *Journal of Artificial Intelligence Research*, 51:829–866, 2014.

Noam Brown, Anton Bakhtin, Adam Lerer, and Qucheng Gong. Combining deep reinforcement learning and search for imperfect-information games. In *Advances in Neural Information Processing Systems*, pages 17057–17069, 2020.

Neil Burch, Michael Johanson, and Michael Bowling. Solving imperfect information games using decomposition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 28(1), pages 602–608, 2014.

Steven Carr, Nils Jansen, Sudarshanan Bharadwaj, Matthijs TJ Spaan, and Ufuk Topcu. Safe policies for factored partially observable stochastic games. In *Robotics: Science and System XVII*, 2021.

Timothy H Chung, Geoffrey A Hollinger, and Volkan Isler. Search and pursuit-evasion in mobile robotics: A survey. *Autonomous robots*, 31:299–316, 2011.

Aurélien Delage, Olivier Buffet, Jilles S Dibangoye, and Abdallah Saffidine. HSVI can solve zero-sum partially observable stochastic games. *Dynamic Games and Applications*, pages 1–55, 2023.

Arnaud Doucet, Nando De Freitas, and Neil James Gordon, editors. *Sequential Monte Carlo methods in practice*, volume 1(2), 2001. Springer.

Johannes Heinrich, Marc Lanctot, and David Silver. Fictitious self-play in extensive-form games. In *International Conference on Machine Learning*, pages 805–813. PMLR, 2015.

Karel Horák, Branislav Bošanskỳ, Vojtěch Kovařík, and Christopher Kiekintveld. Solving zero-sum one-sided partially observable stochastic games. *Artificial Intelligence*, 316:103838, 2023.

Volkan Isler and Nikhil Karnad. The role of information in the cop-robber game. *Theoretical Computer Science*, 399(3):179–190, 2008.

Vojtěch Kovařík, Martin Schmid, Neil Burch, Michael Bowling, and Viliam Lisỳ. Rethinking formal models of partially observable multiagent decision making. *Artificial Intelligence*, 303:103645, 2022.

Marc Lanctot, Kevin Waugh, Martin Zinkevich, and Michael Bowling. Monte Carlo sampling for regret minimization in extensive games. In *Advances in Neural Information Processing Systems*, volume 22, 2009.

Marc Lanctot, Vinicius Zambaldi, Audrūnas Gruslys, Angeliki Lazaridou, Karl Tuyls, Julien Pérolat, David Silver, and Thore Graepel. A unified game-theoretic approach to multiagent reinforcement learning. In *Proceedings of the International Conference on Neural Information Processing Systems*, page 4193–4206, 2017.

Viliam Lisỳ, Marc Lanctot, and Michael H Bowling. Online Monte Carlo counterfactual regret minimization for search in imperfect information games. In *Proceedings of the International Conference on Autonomous Agents and Multiagent Systems*, pages 27–36, 2015.

Kyle Matoba and François Fleuret. Computing preimages of deep neural networks with applications to safety, 2020. openreview.netforum?id=FN7_BUOG78e.

Stephen McAleer, John B Lanier, Kevin A Wang, Pierre Baldi, and Roy Fox. XDO: A double oracle algorithm for extensive-form games. volume 34, pages 23128–23139, 2021.

Matej Moravčík, Martin Schmid, Neil Burch, Viliam Lisỳ, Dustin Morrill, Nolan Bard, Trevor Davis, Kevin Waugh, Michael Johanson, and Michael Bowling. Deepstack: Expert-level artificial intelligence in heads-up no-limit poker. *Science*, 356(6337):508–513, 2017.

Josep M Porta, Nikos Vlassis, Matthijs TJ Spaan, and Pascal Poupart. Point-based value iteration for continuous POMDPs. *Journal of Machine Learning Research*, 7:2329–2367, 2006.

Martin Schmid, Matej Moravčík, Neil Burch, Rudolf Kadlec, Josh Davidson, Kevin Waugh, Nolan Bard, Finbarr Timbers, Marc Lanctot, G Zacharias Holland, et al. Student of games: A unified learning algorithm for both perfect and imperfect information games. *Science Advances*, 9(46): eadg3256, 2023.

Michal Šustr, Vojtěch Kovařík, and Viliam Lisý. Monte Carlo continual resolving for online strategy computation in imperfect information games. In *Proceedings of the International Conference on Autonomous Agents and MultiAgent Systems*, page 224–232, 2019.

Rui Yan, Gabriel Santos, Gethin Norman, David Parker, and Marta Kwiatkowska. Strategy synthesis for zero-sum neuro-symbolic concurrent stochastic games (extended version). arXiv.2202.06255, 2022.

Rui Yan, Gabriel Santos, Gethin Norman, David Parker, and Marta Kwiatkowska. Partially observable stochastic games with neural perception mechanisms. arXiv.2310.11566, 2023.

Wei Zheng, Taeho Jung, and Hai Lin. The Stackelberg equilibrium for one-sided zero-sum partially observable stochastic games. *Automatica*, 140:110231, 2022.

Wei Zheng, Taeho Jung, and Hai Lin. Continuous-observation one-sided two-player zero-sum partially observable stochastic game with public actions. *IEEE Transactions on Automatic Control*, pages 1–15, 2023.

Martin Zinkevich, Michael Johanson, Michael Bowling, and Carmelo Piccione. Regret minimization in games with incomplete information. In *Advances in Neural Information Processing Systems*, volume 20. Curran Associates, Inc., 2007.