Scalable Solutions to Zero-Sum Partially Observable Stochastic Games Through Belief Aggregation with Approximation Guarantees

Kim Hammar and Tansu Alpcan

Department of Electrical and Electronic Engineering
University of Melbourne, Australia
{kim.hammar,tansu.alpcan}@unimelb.edu.au

Abstract

We study the problem of solving one-sided, zero-sum, partially observable stochastic games (POSGs). These games model sequential interactions between two adversaries, where one player has partial observability of the game state. They are applicable to many important domains, such as robotics and cybersecurity. Solving such games is computationally challenging since the solution depends on the first player's belief about the game state, which belongs to a continuous (and often high-dimensional) belief space. In the literature, only a single method has demonstrated reliable performance for solving these types of games, namely Heuristic Search Value Iteration (HSVI). However, this method is restricted to small games. We address this limitation by presenting a new method with similar approximation and convergence guarantees but improved scalability and flexibility, which we call SAB: Shapley iteration with aggregated beliefs. Our method aggregates the belief space into a finite set of representative beliefs and computes their values through Shapley iteration. It then approximates the value function of the POSG through interpolation from these values. We prove that SAB converges and provide a bound on its approximation error. Experiments across several benchmark games show that SAB matches the performance of HSVI on small game instances while also scaling to larger games. Moreover, we find that SAB is up to 79% faster than HSVI at obtaining a near-optimal approximation.

Code — https://github.com/Kim-Hammar/SAB.jl

Introduction

From robotics to cybersecurity, many domains where AI systems operate involve dynamic multi-agent interactions based on competition and partial observability. Such settings naturally call for the tools of game theory, which provide a mathematical framework for modeling strategic behavior. Among the many game models proposed in the literature, one of the most general is the partially observable stochastic game (POSG) (Shapley 1953; Hansen, Bernstein, and Zilberstein 2004). Games of this type capture many features of real-world decision-making: they model dynamics through state transitions over time, partial observability by allowing players to act based on incomplete information, and stochasticity through probabilistic outcomes of actions. This generality allows POSGs to model a broad class of decision-making problems with high practical relevance, such as cybersecurity (Hammar et al. 2025a), online poker (Ganzfried and Sandholm 2015), power grids (Gong et al. 2024), and robotics (So et al. 2023). This game model is also widely used in learning theory as a general framework for studying sequential decision-making; see e.g., (Liu, Szepesvári, and Jin 2022; Delage et al. 2024; Yan et al. 2024).

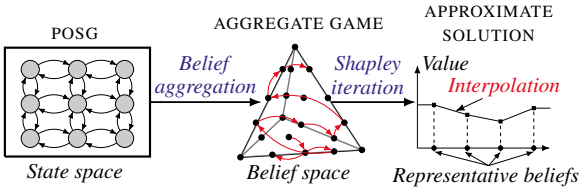

Figure 1: Our method for approximately solving a one-sided partially observable stochastic game (POSG). We transform the POSG into an *aggregate belief game*, which we solve using Shapley iteration. We then use this solution to approximate the value function of the POSG via interpolation.

In this paper, we study a specific subclass of POSGs, namely zero-sum POSGs with *one-sided* partial observability. Such games involve two players: one with partial observability of the game state and one with complete observability. They interact over a sequence of time steps, during which both players select actions that influence the game state and determine the rewards they receive. The asymmetry in information creates a strategic imbalance where the first player must make decisions based on a *belief* about the game state. In contrast, the second player can condition its actions on the actual state. Such asymmetry arises in many practical situations. For example, an air surveillance system equipped with radar may only intermittently detect an aircraft or pick up noise, leading to uncertainty about its position. Meanwhile, the aircraft itself knows its actual location and can maneuver strategically to avoid detection.

From a computational perspective, one-sided POSGs can be seen as a generalization of the partially observable Markov decision process (POMDP) (Goldsmith and Mundhenk 2007). As a result, solving one-sided POSGs is more demanding than solving POMDPs, which is PSPACE-hard (Papadimitriou and Tsitsiklis 1987, Thm. 6). Hence, scal-

able approximation schemes are required when attempting to solve such games in practice. Several approximation schemes have been proposed in the literature. Most of them focus on approximating the game's value function, which encodes the highest expected reward that the first player can guarantee against any opponent strategy. Since the game is zero-sum, this function determines the expected reward in any Nash equilibrium of the game and can thus be used to derive optimal game strategies. However, computing the value function is generally intractable as it depends on the first player's belief about the game state, which belongs to a continuous (and often high-dimensional) belief space.

Approaches proposed in the literature for approximating the value function include heuristic search (Tomášek et al. 2021; Tomášek, Horák, and Bošanský 2024), approximate dynamic programming (Hammar et al. 2025a; Horák et al. 2019), and reinforcement learning (Cai et al. 2024). Among these approaches, the principal method (and the only method with theoretical guarantees) is heuristic search value iteration (HSVI) (Horák, Bošanský, and Pěchouček 2017). However, HSVI is restricted to small games due to its high computational complexity (Horák and Bošanský 2019).

In this paper, we present an alternative method for approximating the value function of one-sided POSGs, which we call SAB: Shapley iteration with aggregated beliefs; see Fig. 1. SAB provides similar theoretical guarantees as HSVI but is more flexible in the sense that its computational cost can be reduced at the expense of increased approximation error. This flexibility allows SAB to approximate the value function of games where HSVI is computationally infeasible.

SAB is based on the conceptual aggregation framework for Markov decision problems formulated by Bertsekas in (Bertsekas 2012, §6.5) and extended to POMDPs in (Li, Hammar, and Bertsekas 2025; Hammar et al. 2025b). The first step of SAB is to aggregate the belief space of the game into a finite set of *representative beliefs*. SAB then uses these beliefs to construct a (computationally tractable) *aggregate belief game*. This construction provides modeling flexibility and allows for the incorporation of domain-specific structure. Next, SAB solves the aggregate game using Shapley iteration (Shapley 1953) to obtain the values of the representative beliefs. Finally, SAB approximates the value function of the POSG through interpolation from these values.

We derive a bound on the approximation error of SAB and prove that it converges. Moreover, computational experiments demonstrate that SAB achieves low approximation error while offering greater scalability and flexibility than HSVI. In summary, we make the following contributions:

- We develop SAB: a new method for approximately solving zero-sum POSGs with one-sided partial observability. Compared to the state-of-the-art, SAB has similar theoretical guarantees but improved scalability and flexibility.
- We derive a bound on the approximation error of SAB and prove that it converges.
- We evaluate SAB on three types of POSGs: stopping, pursuit-evasion, and patrolling games. The results show that SAB matches the approximation error of HSVI while being more scalable and up to 79% more efficient.

Zero-Sum Partially Observable Stochastic Games with One-Sided Partial Observability

A zero-sum, *one-sided*, partially observable stochastic game (POSG) models the strategic interaction between two players in a dynamic system that evolves in discrete time steps $t = 0, 1, \ldots$ over an infinite time horizon. It is defined as

$$\Gamma = \langle \mathcal{N}, \mathcal{S}, \mathcal{A}_1, \mathcal{A}_2, p_{ss'}, r, \gamma, b_0, p, \mathcal{O} \rangle, \quad (1)$$

where $\mathcal{N} = \{1, 2\}$ is the set of players, $\mathcal{S} = \{1, \ldots, n\}$ is the set of states, $\mathcal{O}$ is the set of observations, and $\mathcal{A}_k$ is the set of actions of Player k, all of which are finite. The initial state is drawn from the *belief* $b_0 = (b_0(1), \ldots, b_0(n))$, where $b_0(s)$ is the probability that the initial state is s. We denote the space of all such beliefs by B, i.e., $b_0 \in B$.

State transitions $s \to s'$ occur according to transition probabilities $p_{ss'}(a^1, a^2)$, where a^k is the action of Player k. Each transition is associated with a bounded and discounted (real-valued) reward $\gamma^t r(s, a^1, a^2)$, where $\gamma \in (0, 1)$ is a discount factor. Additionally, each state transition $s \to s'$ generates an observation o with probability $p(o \mid s', a^1, a^2)$.

Both players have perfect recall and follow *behavior strategies*. Specifically, $\pi_k(a_t^k \mid h_t^k)$ is the probability that Player k takes action a_t^k, where h_t^k is the *history* of Player k at time t. For Player 1, this history is defined as

$$h_t^1 = (b_0, a_0^1, o_1, a_1^1, \ldots, a_{t-1}^1, o_t). \quad (2)$$

By contrast, the history of Player 2 is defined as

$$h_t^2 = (b_0, s_0, a_0^2, a_0^1, o_1, s_1, \ldots, a_{t-1}^2, a_{t-1}^1, s_t, o_t). \quad (3)$$

In other words, Player 1 is uncertain about the state while Player 2 knows the state. Player 1's uncertainty is quantified by the *belief state* $b_t = (b_t(1), \ldots, b_t(n))$, where $b_t(s)$ is the conditional probability that the state is s given the history h_t^1.

The expected reward under strategies (π_1, π_2) is

$$\underset{\pi_1,\pi_2}{V}(b) = \lim_{T\to\infty} \mathbb{E}_{\pi_1,\pi_2} \left\{ \sum_{t=0}^{T} \gamma^t r(s_t, a_t^1, a_t^2) | b_0 = b \right\}, \quad (4)$$

where $\mathbb{E}_{\pi_1,\pi_2}\{\cdot\}$ denotes the expected value when actions are sampled as $a^1 \sim \pi_1(\cdot \mid h_t^1)$ and $a^2 \sim \pi_2(\cdot \mid h_t^2)$.

A strategy π_1 is a *best response* against the strategy π_2 if it maximizes $V_{\pi_1,\pi_2}(b)$ for all beliefs b. Similarly, a strategy π_2 is a best response against π_1 if it minimizes the same quantity. When each player follows a best response, their strategies form a Nash equilibrium (Nash 1951, Eq. 1). We denote the expected reward when the game is played according to such strategies by $V^\star(b)$. This value is defined as

$$V^\star(b) = \max_{\pi_1\in\Pi_1} \min_{\pi_2\in\Pi_2} V_{\pi_1,\pi_2}(b) = \min_{\pi_2\in\Pi_2} \max_{\pi_1\in\Pi_1} V_{\pi_1,\pi_2}(b), \quad (5)$$

where Π_k is the strategy space of Player k. We refer to the function $V^\star$ as the *value function* of the game. As is well-known, this function is guaranteed to exist, as stated below.

Proposition 1. *A value function $V^\star$ that satisfies (5) exists.*

We provide the proof of Prop. 1 in the supplementary material (Hammar and Alpcan 2025); a similar proof is presented in (Hammar 2024, Thm. 3). While this proposition states that the value function exists, it does not provide insight on how to compute it. In the next section, we show that this computation can be formulated as a dynamic programming problem, which provides the basis for our solution method, as explained in the subsequent sections.

A Dynamic Programming Formulation for Computing the Value Function of the Game

Proposition 1 implies that there exists a pair of Nash equilibrium strategies $(\pi_1^\star, \pi_2^\star)$ such that the value function $V^\star$ satisfies the following Shapley equation:

$$
\begin{aligned}
V^\star(b_t) &= \mathop{\mathbb{E}}_{\pi_1^\star,\pi_2^\star}\left\{\sum_{j=0}^{\infty}\gamma^j r(s_{t+j}, a^1_{t+j}, a^2_{t+j}) \mid b_t\right\} \\
&= \mathop{\mathbb{E}}_{\pi_1^\star,\pi_2^\star}\left\{r(s_t, a^1_t, a^2_t)\right\} + \\
&\qquad \gamma \mathop{\mathbb{E}}_{\pi_1^\star,\pi_2^\star}\left\{\sum_{j=0}^{\infty}\gamma^j r(s_{t+j+1}, a^1_{t+j+1}, a^2_{t+j+1}) \mid b_t\right\} \\
&= \mathop{\mathbb{E}}_{\pi_1^\star,\pi_2^\star}\left\{r(s_t, a^1_t, a^2_t) + \gamma V^\star(b_{t+1}) \mid b_t\right\} \qquad (6) \\
&= \max_{\mu_1\in M_1}\min_{\mu_2\in M_2}\mathop{\mathbb{E}}_{\mu_1,\mu_2}\left\{r(s_t, a^1_t, a^2_t) + \gamma V^\star(b_{t+1}) \mid b_t\right\},
\end{aligned}
$$

where μ_1 and μ_2 are *stage strategies*, i.e., strategies for the specific stage of the game that is characterized by the belief state b_t. These strategies belong to the strategy spaces $M_1 = \Delta(\mathcal{A}_1)$ and $M_2 = \mathcal{S} \to \Delta(\mathcal{A}_2)$, where $\Delta(\mathcal{A}_k)$ denotes the set of probability distributions over the set $\mathcal{A}_k$.

The Shapley equation in (6) implies that the value function $V^\star$ can be computed through dynamic programming by starting from an initial function V_0 and generating a sequence of successive approximations $V_1, V_2, \ldots$ by repeatedly solving (6) with V_k in place of $V^\star$. However, this computation is computationally intractable due to the continuous belief space B. We present a method for tackling this intractability through *belief aggregation* in the next section.

Our Approach to Approximate the POSG By Constructing an Aggregate Belief Game

To circumvent the intractability of (6), we construct an *aggregate belief game* that is computationally tractable and whose value function can be used to approximate the value function of the original POSG. Towards this construction, we start by specifying a finite subset of the original belief space B, which we denote by $\mathcal{B}$. We refer to the elements of this subset as *representative beliefs* and denote them as (x, y).

We relate these beliefs to the original beliefs via *aggregation probabilities* $\{\phi_{bx} \mid b \in B, x \in \mathcal{B}\}$. For example, we may define ϕ_{bx} to be 1 if and only if x is the nearest representative belief to b according to some distance metric. More generally, these probabilities provide modeling flexibility and can be tailored to the game's structure. Given these probabilities, we construct an *aggregate belief game* with state space $\mathcal{B}$ and transition probabilities defined as

$$
\hat{p}_{xy}(\mu_1,\mu_2) = \sum_{a^1,o}\mu_1(a^1)\hat{p}(o|x,a^1,\mu_2)\phi_{F(x,a^1,\mu_2,o)y}, \quad (7)
$$

where the observation probability $\hat{p}(o \mid x, a^1, \mu_2)$ equals

$$
\sum_{s,s',a^2} x(s)\mu_2(a^2 \mid s)p_{ss'}(a^1,a^2)p(o \mid s', a^1, a^2),
$$

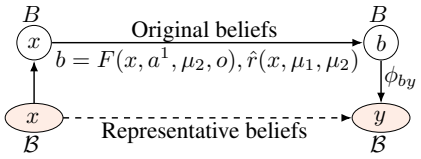

Figure 2: Transition diagram of the aggregate belief game.

and the belief update $b = F(x, a^1, \mu_2, o)$ is calculated as

$$
F(x,a^1,\mu_2,o)(s') = \frac{\sum_{s,a^2} x(s)\mu_2(a^2|s)p(o|s',a^1,a^2)p_{ss'}(a^1,a^2)}{\sum_{a^2,\hat{s},\hat{s}'} p(o|\hat{s}',a^1,a^2)\mu_2(a^2|\hat{s})p_{\hat{s}\hat{s}'}(a^1,a^2)x(\hat{s})} \ \forall s' \in \mathcal{S}.
$$

Similarly, the reward function is defined as

$$
\hat{r}(x,\mu_1,\mu_2) = \sum_{s,a^1,a^2} x(s)\mu_1(a^1)\mu_2(a^2|s)r(s,a^1,a^2). \quad (8)
$$

Remark 1. *We do not require closed-form expressions for the probabilities in (7) or the reward in (8). It suffices to have a simulator that generates beliefs and rewards. In such cases, our method can be implemented in a reinforcement-learning style; see (Hammar and Alpcan 2025) for details.*

The transition probabilities in (7) and the reward function in (8) define a stochastic game whose state space is the set of representative beliefs $\mathcal{B}$; see Fig. 2. We refer to this game as the *aggregate belief game* and denote its value function by $\mathcal{V}^\star : \mathcal{B} \mapsto \mathbb{R}$. For this function, we have the following standard result; see e.g., (Shapley 1953; Filar and Vrieze 1997).

Proposition 2. *Let H be an operator that maps $\mathcal{V} \in \mathbb{R}^{|\mathcal{B}|}$ to $H\mathcal{V}$ with components $(H\mathcal{V})(x)$ calculated as*

$$
\max_{\mu_1\in M_1}\min_{\mu_2\in M_2}\left[\hat{r}(x,\mu_1,\mu_2) + \gamma\sum_{y\in\mathcal{B}}\hat{p}_{xy}(\mu_1,\mu_2)\mathcal{V}(y)\right]. \quad (9)
$$

We have $\|H\mathcal{V} - H\mathcal{V}'\|_\infty \leq \gamma\|\mathcal{V} - \mathcal{V}'\|_\infty$. Further, if $\mathcal{V} \leq \mathcal{V}'$, then $H\mathcal{V} \leq H\mathcal{V}'$. Moreover, the value function $\mathcal{V}^\star$ of the aggregate belief game is the unique fixed point of H.

Shapley gave the original proof of this statement in (Shapley 1953, Thm. 1). The fixed-point property of the operator H enables the computation of the value function $\mathcal{V}^\star$ through repeated application of H. We refer to this iterative procedure as *Shapley iteration*, which provides the basis for our solution method, as described in the next section.

(S)hapley Iteration with (A)ggregated (B)eliefs for Approximating the Value Function

Given the aggregate belief game defined by (7)–(8), the first step of our method is to solve this game through Shapley iteration. We then use the value function of the aggregate game, i.e., $\mathcal{V}^\star$, to approximate the value function of the original POSG (i.e., $V^\star$) through the interpolation formula

$$
\tilde{V}(b) = \sum_{x\in\mathcal{B}}\phi_{bx}\mathcal{V}^\star(x), \qquad \text{for all } b \in B. \quad (10)
$$

Algorithm 1: **S**hapley iteration with **A**ggregated **B**eliefs.

1: **Input:** A one-sided POSG, convergence threshold δ.
2: **Output:** An approximation of $V^\star$; cf. (5).
3: Construct the aggregate game according to (7)–(8).
4: Initialize $\Delta \leftarrow \infty$ and $\mathcal{V}(x) \leftarrow 1$ for all $x \in \mathcal{B}$.
5: **while** $\Delta > \delta$ **do**
6: $\quad \mathcal{V}' \leftarrow H\mathcal{V}$.
7: $\quad \Delta = \|\mathcal{V} - \mathcal{V}'\|_\infty$.
8: $\quad \mathcal{V} \leftarrow \mathcal{V}'$.
9: **end while**
10: Compute $\tilde{V}$ according to (10).
11: **return** $\tilde{V}$.

The resulting value function, $\tilde{V}$, can then be used to derive an ϵ-Nash equilibrium of the POSG; see (Horák et al. 2023) for an analysis of this derivation. The full procedure of our method is outlined in the following pseudocode.

Proposition 3. *Our method (*SAB*, Alg. 1) converges for any convergence threshold δ. If $\delta = 0$, then* SAB *converges to the following value function for the* POSG

$$\tilde{V}(b) = \sum_{x \in \mathcal{B}} \phi_{bx} \mathcal{V}^\star(x), \qquad \text{for all } b \in B.$$

Proof. By Prop. 2, we have $\lim_{k\to\infty} H^k\mathcal{V} = \mathcal{V}^\star$ for any value function $\mathcal{V}$. Since $\mathcal{V}^\star$ is a fixed point of H, we have $\|H\mathcal{V}^\star - \mathcal{V}^\star\|_\infty = 0$, which means that the termination condition on line 5 in Alg. 1 is satisfied. The statement of the proposition thus follows from (10) and line 10 in Alg. 1. □

Remark 2. *The computation expressed by Line 6 in Alg. 1 amounts to solving $|\mathcal{B}|$ linear programs, one for each representative belief. See (Hammar and Alpcan 2025) for details.*

Analysis on the Approximation Error of SAB

We refer to the difference between the approximation $\tilde{V}$ [cf. (10)] obtained through SAB and the value function $V^\star$ [cf. (5)] as the *approximation error*. To gain insight into this error, we consider a special case of interest, called *hard aggregation* (Bertsekas 2012; Li and Bertsekas 2025), whereby each belief b aggregates to a single representative belief x, i.e., $\phi_{bx} = 0$ for all representative beliefs x except a single one, denoted by x_b, for which we have $\phi_{bx_b} = 1$. We also require that $\phi_{xx} = 1$ for all representative beliefs $x \in \mathcal{B}$. In this case, the interpolation (10) yields a function $\tilde{V}$ that is piecewise constant. This structure means that the approximation error is determined by how much the value function $V^\star$ varies for beliefs b that aggregate to the same representative belief. The following proposition formalizes this insight.

Proposition 4. *In the case of hard aggregation, we have*

$$|\tilde{V}(b) - V^\star(b)| \le \frac{\epsilon}{1-\gamma}, \qquad \text{for all } b \in B, \tag{11}$$

where ϵ is a finite constant defined by

$$\epsilon = \max_{x \in \mathcal{B}} \sup_{b,b' \in S_x} |V^\star(b) - V^\star(b')|, \tag{12}$$

$$S_x = \{b \mid b \in B, \phi_{bx} = 1\}. \tag{13}$$

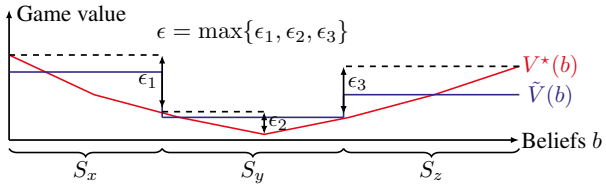

Figure 3: Illustration of the scalar ϵ of Prop. 4. The illustration is based on an approximation with three representative beliefs: $\mathcal{B} = \{x, y, z\}$. The corresponding belief space partitions are: S_x, S_y, and S_z; cf. (13).

Proof. We start by showing that the scalar ϵ in (12) is finite. Prop. 1 implies that $V^\star$ is bounded. As a result, for every $x \in \mathcal{B}$, $\sup_{b,b' \in S_x} |V^\star(b) - V^\star(b')|$ is finite. The finiteness of ϵ follows from the finiteness of the set $\mathcal{B}$.

Next, we consider the vector $\overline{\mathcal{V}}$ defined as

$$\overline{\mathcal{V}}(x) = \inf_{b \in S_x} V^\star(b) + \frac{\epsilon}{1-\gamma}, \qquad x \in \mathcal{B}. \tag{14}$$

Applying the operator H [see (9)] to this vector, we have

$$\begin{aligned}
(H\overline{\mathcal{V}})(x) &= \max_{\mu_1 \in M_1} \min_{\mu_2 \in M_2} \Big[\hat{r}(x, \mu_1, \mu_2) + \gamma \sum_{a^1 \in \mathcal{A}_1} \mu_1(a^1) \cdot \\
&\qquad \sum_{o \in \mathcal{O}} \hat{p}(o \mid x, a^1, \mu_2) \sum_{y \in \mathcal{B}} \phi_{F(x,a^1,\mu_2,o)y} \overline{\mathcal{V}}(y) \Big] \\
&\le \max_{\mu_1 \in M_1} \min_{\mu_2 \in M_2} \Big[\hat{r}(x, \mu_1, \mu_2) + \gamma \sum_{a^1 \in \mathcal{A}_1} \mu_1(a^1) \cdot \\
&\qquad \sum_{o \in \mathcal{O}} \hat{p}(o \mid x, a^1, \mu_2) V^\star(F(x, a^1, \mu_2, o)) \Big] + \frac{\gamma\epsilon}{1-\gamma} \\
&= V^\star(x) + \frac{\gamma\epsilon}{1-\gamma} \le \inf_{b \in S_x} V^\star(b) + \epsilon + \frac{\gamma\epsilon}{1-\gamma} = \overline{\mathcal{V}}(x),
\end{aligned}$$

where the first equality is due to the definition of $\hat{p}_{xy}(\mu_1, \mu_2)$; cf. (7). The first inequality follows from (14) with $F(x, a^1, \mu_2, o)$ in place of b and the fact that $\sum_{a^1,o} \mu_1(a^1)\hat{p}(o \mid x, a^1, \mu_2) = 1$. The second equality holds because $V^\star$ satisfies (6). The last inequality follows from the definition of ϵ and the fact that $x \in S_x$. From this inequality, we obtain $H\overline{\mathcal{V}} \le \overline{\mathcal{V}}$. As a consequence, Prop. 2 implies that the sequence $(\mathcal{V}^k)_{k=0}^\infty$ with $\mathcal{V}^0 = \overline{\mathcal{V}}$ and $\mathcal{V}^{k+1} = H\mathcal{V}^k$ is monotonically decreasing. Moreover, from the fact that H is a contraction [see Prop. 2], we have $\lim_{k\to\infty} \mathcal{V}^k = \mathcal{V}^\star$. Combining these properties with (14), we obtain

$$\mathcal{V}^\star(x) \le \overline{\mathcal{V}}(x) \le V^\star(b) + \frac{\epsilon}{1-\gamma}, \quad \text{for all } b \in S_x, x \in \mathcal{B}.$$

As a result, we have

$$\tilde{V}(b) = \sum_{x \in \mathcal{B}} \phi_{bx} \mathcal{V}^\star(x) = \mathcal{V}^\star(x_b) \le V^\star(b) + \frac{\epsilon}{1-\gamma}.$$

The converse inequality can be derived in an analogous way by considering the vector $\underline{\mathcal{V}}$ with components

$$\underline{\mathcal{V}}(x) = \sup_{b \in S_x} V^\star(b) - \frac{\epsilon}{1-\gamma}, \qquad x \in \mathcal{B}.$$

□

The meaning of Prop. 4 is that the error of the value function approximation $\tilde{V}$ [cf. (10)] is small if the belief space partitions S_x [cf. (13)] conforms to the value function $V^\star$ in the sense that $V^\star$ is approximately constant for beliefs that belong to the same partition S_x, as illustrated in Fig. 3.

Asymptotic Optimality of SAB

A special case of interest is when the aggregation probabilities are defined according to the nearest-neighbor mapping

$$\phi_{bx} = 1 \text{ if and only if } x \in \arg\min_{y \in \mathcal{B}} \|b - y\|_\infty. \tag{15}$$

In this case, a natural way to define the set of representative beliefs $\mathcal{B}$ is through uniform discretization as

$$\mathcal{B} = \left\{ x \;\middle|\; x \in B, x(i) = \frac{\beta_i}{\rho}, \sum_{i=1}^{n} \beta_i = \rho, \beta_i \in \{0, \dots, \rho\} \right\}, \tag{16}$$

where $\rho \in \{1, 2, \dots\}$ can be interpreted as the *discretization resolution* (Hammar and Li 2025). When increasing this resolution, the approximation obtained via (10) converges to the value function $V^\star$, as stated in the following proposition.

Proposition 5. *If the set of representative beliefs $\mathcal{B}$ are defined according to* (16) *and the aggregation probabilities ϕ_{bx} are defined according to* (15)*, then*

$$\lim_{\rho \to \infty} |\tilde{V}(b) - V^\star(b)| = 0, \qquad \textit{for all } b \in B.$$

Proof. It can be shown that the value function $V^\star : B \mapsto \mathbb{R}$ is uniformly continuous; see e.g., (Horák et al. 2023, Prop. 5.8). Fix an arbitrary scalar $\alpha > 0$. By uniform continuity, there exists a scalar $\delta > 0$ such that

$$\|b - b'\|_\infty < \delta \implies |V^\star(b) - V^\star(b')| < \alpha \tag{17}$$

for all $b, b' \in B$.

Equation (15) implies that if $b \in S_x$ [cf. (13)], then

$$\|b - x\|_\infty = \min_{x \in \mathcal{B}} \|b - x\|_\infty.$$

Equation (16) implies that each belief coordinate $x(i)$ equals $\frac{\beta_i}{\rho}$ for some $\beta_i \in \{0, \dots, \rho\}$. As a consequence,

$$\max_{b, b' \in S_x} \|b - b'\|_\infty \leq \frac{2n}{\rho}, \qquad \text{for every } x \in \mathcal{B}.$$

Choose any ρ such that $\frac{2n}{\rho} < \delta$. By (17), we have

$$|V^\star(b) - V^\star(b')| < \alpha, \quad \text{for all } b, b' \in S_x,\ x \in \mathcal{B}.$$

Because $\alpha > 0$ is arbitrary and there exists a large enough ρ such that $\frac{1}{\rho} < \delta$ for any $\delta > 0$, we have

$$\lim_{\rho \to \infty} \max_{x \in \mathcal{B}} \max_{b, b' \in S_x} |V^\star(b) - V^\star(b')| = 0.$$

Hence, the constant ϵ in Prop. 4 diminishes as $\rho \to \infty$. Invoking the error bound in Prop. 4 completes the proof. □

Experimental Evaluation

In this section, we present an experimental evaluation of our method (SAB) and compare it to the state-of-the-art method for computing value functions of one-sided POSGs, namely HSVI (Horák, Bošanský, and Pěchouček 2017).

Example Games for the Evaluation

We conduct the experimental evaluation using three different POSGs from the game-theoretic literature. Each game is parameterized by a value N, which controls the size of the game, as described below.

Stopping game This POSG was introduced in (Hammar et al. 2025a; Hammar and Stadler 2024). The game models an intrusion response use case on a networked system with N components. Player 1 represents the system operator and Player 2 represents an attacker. The game has $N + 1$ states, where each state represents the number of components that are compromised by the attacker. Each player has two actions: STOP ($a^k = 1$) and CONTINUE ($a^k = 0$). Action $a^1 = 1$ recovers all compromised components and transitions the game to the state $s = 0$. Conversely, action $a^2 = 1$ represents an attempt to compromise a component, which causes the state s to be incremented by 1 with probability $p_{\mathrm{A}} = 0.2$. The actions $a^1 = 0$ and $a^2 = 0$ mean to wait.

The observation space of the game is $\mathcal{O} = \{0, 1, \dots, 9\}$, where each observation $o \in \mathcal{O}$ represents the number of security alerts. These alerts are distributed as $\mathrm{BetaBin}(\alpha = 0.7, \beta = 3)$ if $s = 0$ and as $\mathrm{BetaBin}(\alpha = 1, \beta = 0.7)$ otherwise. Finally, the reward function is defined as

$$r(s, a^1, a^2) = s^{5/4}(1 - a^1) + a^1(1 - 2\delta_{s0}),$$

where $\delta_{ij} = 0$ if $i \neq j$ and $\delta_{ij} = 1$ if $i = j$.

Pursuit-evasion game A variant of this POSG was introduced in (Horák and Bošanský 2016). The game models a pursuer (Player 1) trying to capture an evader (Player 2) on a $3 \times N$ grid. The state represents the evader's position. The action of the pursuer determines which grid cell to search, while the action of the evader decides which cell to move to next. The evader can only move to cells that are adjacent to its current position. If the pursuer searches the evader's current position, the evader is caught with probability $p_{\mathrm{C}} = 0.5$. In that case, the reward is 1. In all other cases, the reward is 0. The evader's next position after being caught is drawn uniformly at random from all positions on the grid.

Patrolling game A variant of this POSG was introduced in (Vorobeychik et al. 2014). A similar game was also studied in (Basilico, De Nittis, and Gatti 2016). The game models a patroller (Player 1) that seeks to defend a set of targets from an attacker (Player 2). The game is played on a graph where nodes represent targets and edges represent paths between them. The attacker aims to remain undetected at a target for $t_{\mathrm{A}} = 3$ consecutive time steps to carry out a successful attack. The patroller's goal is to intercept the attacker before this occurs. The game state captures the attacker's current target and the number of consecutive steps spent there. At each step, the attacker chooses to move along an edge or remain at its current location, while the patroller selects a target to visit. A successful attack yields a reward of -1. Detection of the attacker yields a reward of 1. In all other cases, the reward is 0. After detection, the attacker's next position is sampled uniformly at random. The graph is an Erdős-Rényi graph with parameter $p = 0.2$ and N nodes.

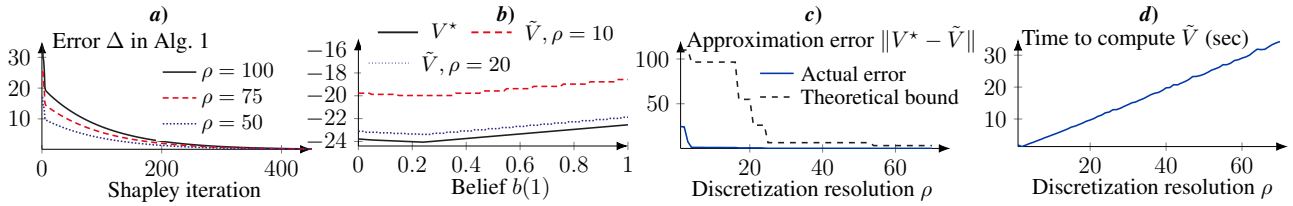

Figure 4: Numerical illustrations of Props. 3–5 based on the stopping game with $N = 1$. Plot ***a)*** shows the convergence of SAB (Alg. 1) for varying resolutions ρ [cf. (16)]; plot ***b)*** visually compares the approximation obtained through SAB with the value function; plot ***c)*** compares the approximation error with the theoretical bound in Prop. 4; and plot ***d)*** shows the compute time.

Experimental Setup

We run all experiments on an M4 PRO chip and instantiate all games with the discount factor $\gamma = 0.99$. We use the convergence threshold $\delta = 0.01$ for both SAB and HSVI. If convergence does not occur within 2 hours, we use the smallest value of δ reached after 2 hours. Unless stated otherwise, we instantiate SAB with the set of representative beliefs generated via (16) and the aggregation probabilities defined by the nearest-neighbor mapping (15). Since both SAB and HSVI are deterministic algorithms, we report their results based on a single evaluation. Further details about our experimental setup are provided in (Hammar and Alpcan 2025).

Numerical Illustrations of Our Theoretical Results

Before presenting the comparison between our method (SAB) and HSVI, we start by numerically illustrating Props. 3–5. To this end, we apply SAB to the stopping game with $N = 1$. This game instantiation has only two states, which allows us to efficiently compute the value function $V^\star$ and visually compare it with the approximate value function $\tilde{V}$ [cf. (10)] obtained through SAB; see Fig. 4.

We observe in Fig. 4.a that SAB converges, as expected from Prop. 3. Moreover, Figs. 4.b-c show that the approximation error of $\tilde{V}$ decreases when the discretization resolution ρ in (16) increases, as asserted in Prop. 5. We also note in Fig. 4.c that the error bound stated in Prop. 4 is not tight but becomes increasingly accurate when the resolution ρ increases. However, increasing the resolution ρ also increases the computation time of SAB, as shown in Fig. 4.d. Hence, the discretization resolution ρ governs a trade-off between computational expedience and approximation error.

Comparison Between SAB and HSVI

Figures 5–6 present a comparison between our method (SAB) and HSVI across the POSGs described above. We instantiate each game with $N \in \{1, 2, 3, 4\}$. These values of N allow us to compute the value function $V^\star$ [cf. (5)] through HSVI and compute the approximation error $\|V^\star - \tilde{V}\|_\infty$, where $\tilde{V}$ is the value function produced by SAB.

The main trend observed across all evaluations is that SAB consistently produces value functions with low approximation error much faster than HSVI. In contrast, HSVI converges more slowly but eventually achieves slightly lower approximation errors when given sufficient computation time. Specifically, we find that SAB is 79% faster than HSVI (on average) in reaching an approximation error of 10 or less; see Fig. 6. These results suggest the following criterion for selecting between the two methods: if rapid computation of a reasonably accurate value function is required, then SAB is preferable. Conversely, when an exact value function is needed and ample time is available, then HSVI is preferable.

In addition to being more efficient, SAB is more flexible than HSVI by allowing to tailor the aggregation to game-specific structures. While we have not exploited this flexibility in the preceding evaluation to enable a fair comparison with HSVI, it can significantly reduce the computation time of SAB in certain cases, as demonstrated in the next section.

Evaluation of SAB on a Large Game

To demonstrate the flexibility of SAB, we apply it to an instance of the stopping game with $N = 500$ components and $|\mathcal{O}| = 10,000$ observations. Due to the high-dimensional belief space and large observation space in this game, it is computationally infeasible to apply HSVI. However, SAB can be applied to this game by adapting the set of representative beliefs $\mathcal{B}$ to reduce computational complexity at the expense of increased approximation error. In particular, for this experiment, we construct the set $\mathcal{B}$ by selecting 100 beliefs via discretization of a region of the belief space where only the states 0 or 1 have non-zero probability. We chose this region based on the structure of the game, which predominantly involves uncertainty between these two states.

Because computing the exact value function $V^\star$ is intractable for this game, we evaluate the quality of SAB's approximation using the *approximate exploitability* metric (Timbers et al. 2022), which is defined as

$$\eta = V_{\hat{\pi}_1, \pi_2}(b_0) - V_{\pi_1, \hat{\pi}_2}(b_0), \tag{18}$$

where b_0 is the initial belief, (π_1, π_2) are the strategies obtained via SAB using (9), $\hat{\pi}_1$ is an approximate best response against the strategy π_2, and $\hat{\pi}_2$ is an approximate best response against the strategy π_1. These approximate best responses can be efficiently obtained through reinforcement learning techniques. The closer the exploitability becomes to 0, the closer (π_1, π_2) is likely to be to a Nash equilibrium.

Since HSVI is impractical to apply to this game, we compare our method with neural fictitious self-play (NFSP), which is a reinforcement learning algorithm for approximating equilibria of games with partial observability (Heinrich and Silver 2016). We use the OPENSPIEL implementation of

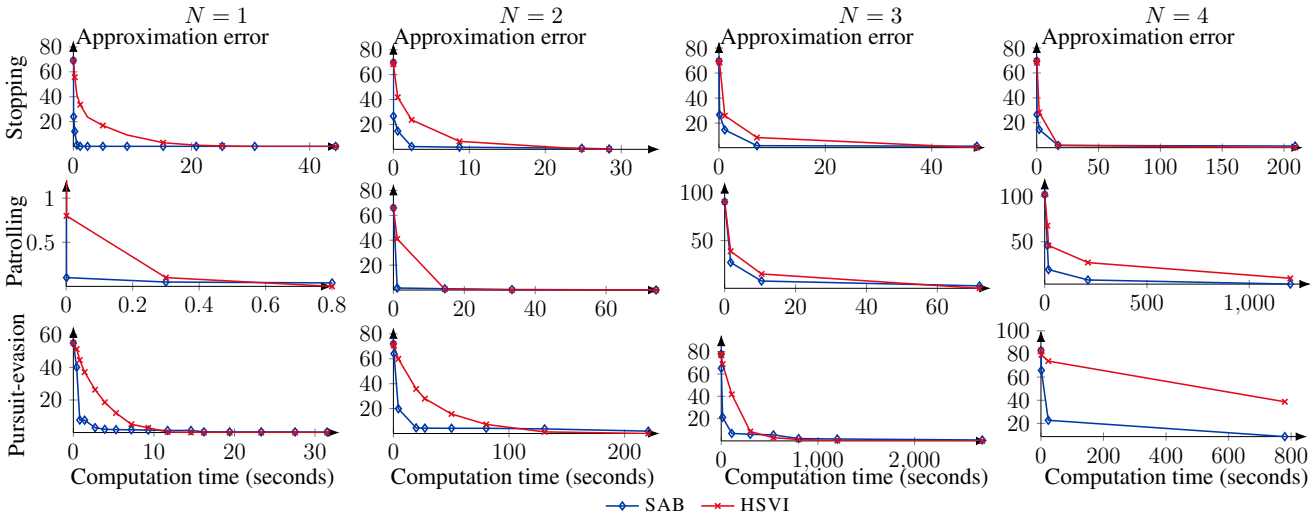

Figure 5: Comparison between our method (SAB) and the current state-of-the-art method (HSVI) on three example POSGs from the game-theoretic literature. Rows relate to different games. Columns relate to the parameter N, which controls the size of the game instantiation. The x-axes indicate computation time and the y-axes indicate the approximation error $\|V^\star - \tilde{V}\|$.

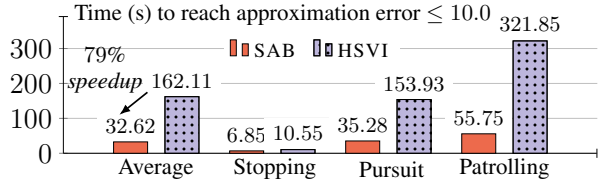

Figure 6: Time to reach an approximation error of 10.0 or less across the evaluation games with $N \in \{1, 2, 3, 4\}$.

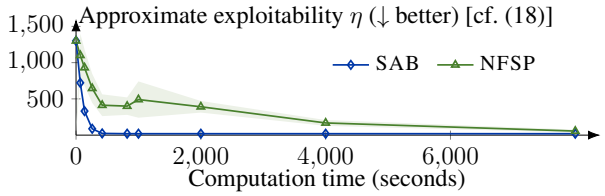

Figure 7: Comparison between our method (SAB) and a reinforcement learning method (NFSP) on a large instance of the stopping game. The x-axis shows the computation time and the y-axis shows the approximate exploitability; cf. (18). Curves show the mean value from evaluations with 5 random seeds; shaded areas indicate standard deviations.

NFSP (Lanctot et al. 2020); see (Hammar and Alpcan 2025) for details. The main difference between SAB and NFSP is that NFSP does not provide any theoretical guarantees.

The evaluation results are shown in Fig. 7. We observe that SAB produces approximations with low approximate exploitability faster than NFSP. We explain this speedup by SAB's ability to exploit the structure of the game and incorporate domain knowledge in the belief aggregation.

Remark 3. *Due to the infinite horizon, extensive-form game algorithms (e.g., CFR (Zinkevich et al. 2007)), do not apply.*

Discussion of the Evaluation Results

The key properties of SAB, as evidenced by our theoretical analysis and experimental results, are as follows.

- *Reliable.* SAB provides similar theoretical performance guarantees as HSVI and achieves almost as low approximation error on small game instances.
- *Efficient.* SAB attains approximations with low error up to 79% faster than HSVI on small game instances.
- *Scalable and flexible.* SAB's belief aggregation mechanism enables it to scale to large game instances and incorporate game-specific structure.
- *Extensible.* While a key strength of SAB lies in its theoretical foundation, it can also be integrated with neural network approximations for identifying representative beliefs and performing the interpolation (10).

Conclusion

We present SAB: Shapley iteration with aggregated beliefs, a new method for approximately solving zero-sum POSGs with one-sided partial observability. By approximating the POSG with an aggregate belief game that can be efficiently solved using Shapley iteration, SAB offers a scalable and flexible alternative to existing methods, such as HSVI. Our theoretical analysis establishes a bound on the approximation error of SAB and a convergence guarantee. We evaluate SAB on three different types of games. The empirical results show that SAB maintains low approximation error while improving scalability and flexibility in comparison with HSVI. These findings suggest that SAB is a practical and reliable method for approximately solving one-sided POSGs. Future work will focus on integrating deep reinforcement learning techniques to further improve the scalability of SAB.

Acknowledgments

This research is supported by the Swedish Research Council under contract 2024-06436. The authors are grateful to Prof. Branislav Bosanský for sharing the code of HSVI. The authors also thank Dr. Yuchao Li and Prof. Dimitri Bertsekas at Arizona State University for their valuable feedback on an early draft of this paper.

References

Basilico, N.; De Nittis, G.; and Gatti, N. 2016. A Security Game Combining Patrolling and Alarm-Triggered Responses Under Spatial and Detection Uncertainties. *Proceedings of the AAAI Conference on Artificial Intelligence*, 30(1).

Bertsekas, D. 2012. *Dynamic Programming and Optimal Control: Vol. II*. Athena Scientific Belmont, 4th edition.

Cai, Y.; Liu, X.; Oikonomou, A.; and Zhang, K. 2024. Provable Partially Observable Reinforcement Learning with Privileged Information. In Globerson, A.; Mackey, L.; Belgrave, D.; Fan, A.; Paquet, U.; Tomczak, J.; and Zhang, C., eds., *Advances in Neural Information Processing Systems*, volume 37, 63790–63857. Curran Associates, Inc.

Delage, A.; Buffet, O.; Dibangoye, J. S.; and Saffidine, A. 2024. HSVI Can Solve Zero-Sum Partially Observable Stochastic Games. *Dynamic Games and Applications*, 14(4): 751–805.

Filar, J.; and Vrieze, K. 1997. *Competitive Markov Decision Processes*. Springer.

Ganzfried, S.; and Sandholm, T. 2015. Endgame Solving in Large Imperfect-Information Games. In *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems*, AAMAS '15, 37–45. Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems. ISBN 9781450334136.

Goldsmith, J.; and Mundhenk, M. 2007. Competition Adds Complexity. In Platt, J.; Koller, D.; Singer, Y.; and Roweis, S., eds., *Advances in Neural Information Processing Systems*, volume 20. Curran Associates, Inc.

Gong, J.; Yu, N.; Han, F.; Tang, B.; Wu, H.; and Ge, Y. 2024. Energy Scheduling Optimization for Microgrids Based on Partially Observable Markov Game. *IEEE Transactions on Artificial Intelligence*, 5(11): 5371–5380.

Hammar, K. 2024. *Optimal Security Response to Network Intrusions in IT Systems*. Ph.D. thesis, KTH Royal Instistute of Technology.

Hammar, K.; and Alpcan, T. 2025. *Supplementary material - Scalable Solutions to Zero-Sum Partially Observable Stochastic Games through Belief Aggregation with Approximation Guarantees*. https://doi.org/10.5281/zenodo.17584615.

Hammar, K.; and Li, T. 2025. Online Incident Response Planning under Model Misspecification through Bayesian Learning and Belief Quantization. In *Proceedings of the 2025 Workshop on Artificial Intelligence and Security*, AISec '25. New York, NY, USA: Association for Computing Machinery. Preprint: https://arxiv.org/abs/2508.14385.

Hammar, K.; Li, T.; Stadler, R.; and Zhu, Q. 2025a. Adaptive Security Response Strategies Through Conjectural Online Learning. *IEEE Transactions on Information Forensics and Security*, 20: 4055–4070.

Hammar, K.; Li, Y.; Alpcan, T.; Lupu, E. C.; and Bertsekas, D. 2025b. Adaptive Network Security Policies via Belief Aggregation and Rollout. https://arxiv.org/abs/2507.15163, arXiv:2507.15163.

Hammar, K.; and Stadler, R. 2024. Learning Near-Optimal Intrusion Responses Against Dynamic Attackers. *IEEE Transactions on Network and Service Management*, 21(1): 1158–1177.

Hansen, E. A.; Bernstein, D. S.; and Zilberstein, S. 2004. Dynamic programming for partially observable stochastic games. In *Proceedings of the 19th National Conference on Artifical Intelligence*, AAAI'04, 709–715. AAAI Press. ISBN 0262511835.

Heinrich, J.; and Silver, D. 2016. Deep Reinforcement Learning from Self-Play in Imperfect-Information Games. arXiv:1603.01121.

Horák, K.; and Bošanský, B. 2016. A Point-Based Approximate Algorithm for One-Sided Partially Observable Pursuit-Evasion Games. In Zhu, Q.; Alpcan, T.; Panaousis, E.; Tambe, M.; and Casey, W., eds., *Decision and Game Theory for Security*, 435–454. Cham: Springer International Publishing. ISBN 978-3-319-47413-7.

Horák, K.; and Bošanský, B. 2019. Solving Partially Observable Stochastic Games with Public Observations. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01): 2029–2036.

Horák, K.; Bošanský, B.; Kiekintveld, C.; and Kamhoua, C. 2019. Compact Representation of Value Function in Partially Observable Stochastic Games. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, 350–356. International Joint Conferences on Artificial Intelligence Organization.

Horák, K.; Bošanský, B.; Kovařík, V.; and Kiekintveld, C. 2023. Solving zero-sum one-sided partially observable stochastic games. *Artificial Intelligence*, 316: 103838.

Horák, K.; Bošanský, B.; and Pěchouček, M. 2017. Heuristic Search Value Iteration for One-Sided Partially Observable Stochastic Games. *Proceedings of the AAAI Conference on Artificial Intelligence*, 31(1).

Lanctot, M.; Lockhart, E.; Lespiau, J.-B.; Zambaldi, V.; Upadhyay, S.; Pérolat, J.; Srinivasan, S.; Timbers, F.; Tuyls, K.; Omidshafiei, S.; Hennes, D.; Morrill, D.; Muller, P.; Ewalds, T.; Faulkner, R.; Kramár, J.; Vylder, B. D.; Saeta, B.; Bradbury, J.; Ding, D.; Borgeaud, S.; Lai, M.; Schrittwieser, J.; Anthony, T.; Hughes, E.; Danihelka, I.; and Ryan-Davis, J. 2020. OpenSpiel: A Framework for Reinforcement Learning in Games. arXiv:1908.09453.

Li, Y.; and Bertsekas, D. 2025. An Error Bound for Aggregation in Approximate Dynamic Programming. arXiv:2507.01324.

Li, Y.; Hammar, K.; and Bertsekas, D. 2025. Feature-Based Belief Aggregation for Partially Observable Markov

Decision Problems. https://arxiv.org/abs/2507.04646, arXiv:2507.04646.

Liu, Q.; Szepesvári, C.; and Jin, C. 2022. Sample-efficient reinforcement learning of partially observable Markov games. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22. Red Hook, NY, USA: Curran Associates Inc. ISBN 9781713871088.

Nash, J. F. 1951. Non-cooperative games. *Annals of Mathematics*, 54: 286–295.

Papadimitriou, C. H.; and Tsitsiklis, J. N. 1987. The Complexity of Markov Decision Processes. *Math. Oper. Res.*, 12: 441–450.

Shapley, L. S. 1953. Stochastic Games. *Proceedings of the National Academy of Sciences*, 39(10): 1095–1100.

So, O.; Drews, P.; Balch, T.; Dimitrov, V.; Rosman, G.; and Theodorou, E. A. 2023. MPOGames: Efficient Multimodal Partially Observable Dynamic Games. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 3189–3196.

Timbers, F.; Bard, N.; Lockhart, E.; Lanctot, M.; Schmid, M.; Burch, N.; Schrittwieser, J.; Hubert, T.; and Bowling, M. 2022. Approximate Exploitability: Learning a Best Response. In Raedt, L. D., ed., *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, 3487–3493. International Joint Conferences on Artificial Intelligence Organization. Main Track.

Tomášek, P.; Horák, K.; Aradhye, A.; Bošanský, B.; and Chatterjee, K. 2021. Solving Partially Observable Stochastic Shortest-Path Games. In Zhou, Z.-H., ed., *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, 4182–4189. International Joint Conferences on Artificial Intelligence Organization. Main Track.

Tomášek, P.; Horák, K.; and Bošanský, B. 2024. Iterative algorithms for solving one-sided partially observable stochastic shortest path games. *International Journal of Approximate Reasoning*, 175: 109297.

Vorobeychik, Y.; An, B.; Tambe, M.; and Singh, S. 2014. Computing Solutions in Infinite-Horizon Discounted Adversarial Patrolling Games. *Proceedings of the International Conference on Automated Planning and Scheduling*, 24(1): 314–322.

Yan, R.; Santos, G.; Norman, G.; Parker, D.; and Kwiatkowska, M. 2024. HSVI-based online minimax strategies for partially observable stochastic games with neural perception mechanisms. In Abate, A.; Cannon, M.; Margellos, K.; and Papachristodoulou, A., eds., *Proceedings of the 6th Annual Learning for Dynamics and Control Conference*, volume 242 of *Proceedings of Machine Learning Research*, 80–91. PMLR.

Zinkevich, M.; Johanson, M.; Bowling, M.; and Piccione, C. 2007. Regret minimization in games with incomplete information. In *Proceedings of the 21st International Conference on Neural Information Processing Systems*, NIPS'07, 1729–1736. Red Hook, NY, USA: Curran Associates Inc. ISBN 9781605603520.