

Solving Long-run Average Reward Robust MDPs via Stochastic Games

Krishnendu Chatterjee¹, Ehsan Kafshdar Goharshady¹, Mehrdad Karrabi¹,
Petr Novotný², Đorđe Žikelić^{3*}

¹Institute of Science and Technology Austria (ISTA), Austria

²Masaryk University, Czech Republic

³Singapore Management University, Singapore

{krishnendu.chatterjee, ehsan.goharshady, mehrdad.karrabi}@ist.ac.at,
petr.novotny@fi.muni.cz, dzikelic@smu.edu.sg

Abstract

Markov decision processes (MDPs) provide a standard framework for sequential decision making under uncertainty. However, MDPs do not take uncertainty in transition probabilities into account. Robust Markov decision processes (RMDPs) address this shortcoming of MDPs by assigning to each transition an uncertainty set rather than a single probability value. In this work, we consider polytopic RMDPs in which all uncertainty sets are polytopes and study the problem of solving long-run average reward polytopic RMDPs. We present a novel perspective on this problem and show that it can be reduced to solving long-run average reward turn-based stochastic games with *finite state and action spaces*. This reduction allows us to derive several important consequences that were hitherto not known to hold for polytopic RMDPs. First, we derive new computational complexity bounds for solving long-run average reward polytopic RMDPs, showing for the first time that the threshold decision problem for them is in $\text{NP} \cap \text{CONP}$ and that they admit a randomized algorithm with sub-exponential expected runtime. Second, we present Robust Polytopic Policy Iteration (RPPI), a novel policy iteration algorithm for solving long-run average reward polytopic RMDPs. Our experimental evaluation shows that RPPI is much more efficient in solving long-run average reward polytopic RMDPs compared to state-of-the-art methods based on value iteration.

1 Introduction

Robust Markov decision processes. Markov decision processes (MDPs) are widely adopted as a framework for sequential decision making in environments that exhibit stochastic uncertainty [Puterman, 1994]. The goal of solving an MDP is to compute a policy which maximizes expected payoff with respect to some reward objective. However, classical algorithms for solving MDPs assume that tran-

sition probabilities are known and available. While this is plausible from a theoretical perspective, in practice this assumption is not always justified – transition probabilities are typically inferred from data and as such their estimates come with a certain level of uncertainty. This has motivated the study of *robust Markov decision processes (RMDPs)* [Nilim and Ghaoui, 2003; Iyengar, 2005]. RMDPs differ from MDPs in that they only assume knowledge of some uncertainty set containing true transition probabilities. The goal of solving RMDPs is to compute a policy which maximizes the *worst-case expected payoff* with respect to all possible choices of transition probabilities belonging to the uncertainty set. The advantage of considering RMDPs rather than MDPs for planning and sequential decision making tasks is that solving RMDPs provides policies that are *robust* to potential uncertainty in the underlying environment model.

Long-run average reward RMDPs. Recent years have seen significant interest in solving RMDPs. Two most fundamental reward objectives defined over unbounded or infinite time horizon trajectories are the *discounted-sum reward* and the *long-run average reward*. While there are many works on solving discounted-sum reward RMDPs (see an overview of related work below), the problem of solving long-run average reward RMDPs remains comparatively less explored. However, long-run average reward is, along with discounted-sum, one of the two fundamental objectives used in agent planning and reinforcement learning [Puterman, 1994; Sutton *et al.*, 1999]. Unlike discounted-sum, which places more emphasis on earlier rewards and prefixes of a run, long run average reward focuses on long-term sustainability of the reward signal.

Prior work and challenges. Most prior work on solving long-run average reward RMDPs focuses on special instances of RMDPs in which uncertainty sets are either products of intervals (i.e. L^∞ -balls) or L^1 -balls around some pivot probability distribution. These works typically focus on studying *Blackwell optimality* in RMDPs. In standard MDPs, Blackwell optimality [Puterman, 1994] is a classical result which says that there exists some threshold discount factor $\gamma^* \in [0, 1)$ such that the same policy maximizes expected discounted-sum reward for any discount factor $\gamma \in [\gamma^*, 1)$ while also maximizing the long-run average reward in the MDP. This allows using the methods for discounted-sum reward MDPs to solve long-run average reward MDPs.

*Part of the work done while the author was at the Institute of Science and Technology Austria (ISTA).

The work of [Tewari and Bartlett, 2007] considers RMDPs in which uncertainty sets are products of intervals (also called interval MDPs [Givan *et al.*, 2000]), and is the first to establish Blackwell optimality for interval MDPs. It also proposes a value iteration algorithm for solving long-run average reward interval MDPs. The works [Grand-Clément and Petrik, 2023; Goyal and Grand-Clément, 2023] establish Blackwell optimality for RMDPs with general polytopic uncertainty sets. Moreover, the work [Grand-Clément and Petrik, 2023] designs value and policy iteration algorithms for solving long-run average reward RMDPs with interval or L^1 -uncertainty sets. Finally, the recent work of [Wang *et al.*, 2023] establishes Blackwell optimality and proposes a value iteration algorithm for a general class of RMDPs in which uncertainty sets are only assumed to be compact, but in which the RMDP is assumed to be unichain.

While these works present significant advances in solving long-run average reward RMDPs, our understanding of this problem still suffers from a few limitations:

- *Computational complexity.* While prior works focus on studying Blackwell optimality in RMDPs and algorithms for solving them, no upper bound on the computational complexity of solving this problem is known.
- *Structural or uncertainty restrictions.* The value iteration algorithm of [Wang *et al.*, 2023] requires the RMDP to be unichain. On the other hand, other algorithms discussed above are applicable to RMDPs with either interval or L^1 -uncertainty sets.
- *Policy iteration.* In classical MDP and stochastic games [Puterman, 1994; Hansen *et al.*, 2013] and in discounted-sum RMDPs [Sinha and Ghate, 2016; Ho *et al.*, 2021], the number of iterations of policy iteration is known to be more efficient than value iteration. However, for RMDPs with uncertainty sets that are not intervals or L^1 -balls, the only available algorithm is that of [Wang *et al.*, 2023] which is based on value iteration.

Our contributions. This work focuses on computational complexity aspects and efficient algorithms for solving long-run average reward RMDPs and our goal is to address the limitations discussed above. To that end, we consider *polytopic RMDPs* in which all uncertainty sets are assumed to be polytopes and which unify and strictly subsume RMDPs with interval or L^1 -uncertainty sets. We present a *new perspective* on the problem of solving long-run average reward polytopic RMDPs by showing that it admits a *linear-time reduction* to the problem of solving long-run average reward *turn-based stochastic games (TBSGs)* in which both state and action sets are finite. The reduction is linear in the size of the RMDP and the number of vertices of uncertainty polytopes. While the connection between RMDPs and TBSGs has been informally mentioned in several prior works, e.g. [Nilim and Ghaoui, 2003; Tessler *et al.*, 2019; Goyal and Grand-Clément, 2023], we are not aware that any prior work has actually formalized this reduction. The reduction formalization turns out to be technically non-trivial and we provide a first formal proof of the correctness of the reduction. The reduction formalization is the main technical contribution of our work.

While definitely compelling, the reduction is not only of

theoretical interest. TBSGs have been widely studied within computational game theory and formal methods communities [Shapley, 1953; Filar and Vrieze, 1996; Kučera, 2011] and this reduction allows us to leverage a plethora of results on TBSGs in order to establish new results on long-run average reward RMDPs that were hitherto not known to be true. In particular, our results allow us to overcome all the limitations discussed above. Our main contributions are as follows:

1. We establish novel *complexity results* on solving long-run average reward polytopic RMDPs, showing for the first time that the threshold decision problem for them is in $NP \cap \text{CONP}$ and that they admit a randomized algorithm with sub-exponential expected runtime. Moreover, for RMDPs with constant number of actions and treewidth of the underlying graph, the problem admits an algorithm with quasi-polynomial runtime.
2. By utilizing Blackwell optimality for TBSGs and the fact that policy iteration has already been shown to be sound for discounted-sum reward TBSGs, we propose *Robust Polytopic Policy Iteration (RPPI)*, our novel algorithm for solving long-run average reward polytopic RMDPs. Our algorithm *does not impose structural restrictions* on the RMDP such as unichain or aperiodic.
3. *Experimental evaluation.* We implement RPPI and experimentally compare it against the value iteration algorithm of [Wang *et al.*, 2023]. We show that RPPI achieves significant efficiency gains on unichain RMDPs while also being practically applicable to RMDPs without such structural restrictions.

Related work. The related work on solving long-run average reward RMDPs was discussed above, hence we omit repetition. In what follows, we overview a larger body of existing works on solving discounted-sum reward RMDPs. Existing approaches can be classified into model-based and model-free. Model-based approaches assume the full knowledge of the uncertainty set and compute optimal policies with exact values on the worst-case expected payoff that they achieve, see e.g. [Nilim and Ghaoui, 2003; Iyengar, 2005; Wiesemann *et al.*, 2013; Kaufman and Schaefer, 2013; Tamar *et al.*, 2014; Ghavamzadeh *et al.*, 2016; Ho *et al.*, 2021]. These works also provide deep theoretical insight into RMDPs and show that many classical results on MDPs transfer to the RMDP setting. Model-free approaches, on the other hand, do not assume the full knowledge of the uncertainty set. Rather, they learn the uncertainty sets from data and as such provide more scalable and practical algorithms, although without guarantees on optimality of learned policies [Roy *et al.*, 2017; Tessler *et al.*, 2019; Wang and Zou, 2021; Wang and Zou, 2022; Ho *et al.*, 2018; Yang *et al.*, 2022].

In the final days of preparing this submission, we found a concurrent work of [Grand-Clément *et al.*, 2023] which also explores the connection between long-run average reward RMDPs and TBSGs in order to study Blackwell optimality. Our focus in this work is on computational complexity aspects and efficient algorithms for solving this problem. Furthermore, we also establish that our reduction is efficient, along with its formal correctness, that allows transfer of algorithms from the stochastic games literature to RMDPs.

2 Preliminaries and Models

For a set X , we denote by $\Delta(X)$ the set of all probability distributions over X .

MDPs. A (finite-state, discrete-time) *Markov decision process (MDP)* [Puterman, 1994] is a tuple (S, A, δ, r, s_0) , where S is a finite set of *states*, A is a finite set of *actions*, $\delta: S \times A \rightarrow \Delta(S)$ is a probabilistic *transition function*, $r: S \times A \rightarrow \mathbb{R}$ is a *reward function*, and $s_0 \in S$ is the *initial state*. We abbreviate the transition probability $\delta(s, a)(s')$ by $\delta(s' | s, a)$.

The dynamics of an MDP are defined through the notion of a policy. A *policy* is a map $\sigma: (S \times A)^* \times S \rightarrow \Delta(A)$ which to each finite history of state-action pairs ending in a state assigns the probability distribution over the actions. Under a policy σ , the MDP evolves as follows. The MDP starts in the initial state s_0 . Then, in any time step $t \in \{0, 1, 2, \dots\}$, based on the current history $h_t = s_0, a_0, \dots, s_{t-1}, a_{t-1}, s_t$ the policy σ samples an action a_t to play according to the probability distribution $\sigma(h_t)$. The agent then receives the immediate reward $r_t = r(s_t, a_t)$ and the MDP transitions into the next state s_{t+1} sampled according to the probability distribution $\delta(s_t, a_t)$. The process continues in this manner *ad infinitum*. A policy is said to be:

- *positional (or memoryless)*, if the choice of the probability distribution over actions depends only on the last state in the history, i.e. if $\sigma(s_0, a_0, \dots, s_{t-1}, a_{t-1}, s_t) = \sigma(s'_0, a'_0, \dots, s'_{t-1}, a'_{t-1}, s'_t)$ whenever $s_t = s'_t$;
- *pure*, if for every history h the probability distribution $\sigma(h)$ assigns probability 1 to a single action in A .

Robust MDPs. Robust MDPs generalize MDPs in that they prescribe to each state-action pair an *uncertainty set*, which contains a set of probability distributions over the states in S . Thus, MDPs present a special case of robust MDPs in which the uncertainty set prescribed to each state-action pair contains only a single probability distribution over the states.

Formally, a *robust MDP (RMDP)* [Nilim and Ghaoui, 2003; Iyengar, 2005] is a tuple $(S, A, \mathcal{P}, r, s_0)$, where S is a finite set of *states*, A is a finite set of *actions*, $\mathcal{P} \subseteq \Delta(S)^{S \times A}$ is an *uncertainty set*, $r: S \times A \rightarrow \mathbb{R}$ is a *reward function*, and $s_0 \in S$ is the initial state.

The intuition behind RMDPs is that, at each time step, the true transition function is chosen adversarially from the uncertainty set \mathcal{P} . The dynamics of RMDPs are defined through a pair of policies – the *agent policy* and the *environment (or adversarial) policy*. The agent policy σ is defined analogously as in the case of MDPs above. On the other hand, the environment policy is a map $\pi: (S \times A)^* \times S \times A \rightarrow \mathcal{P}$ which to each finite history of state-action pairs assigns an element of the uncertainty set \mathcal{P} . Under the agent policy σ and the environment policy π , the RMDP evolves analogously as an MDP defined above, with the difference that in each time step $t \in \{0, 1, 2, \dots\}$, the state s_{t+1} is sampled from $\delta_t(s_t, a_t)$, where $\delta_t = \pi(s_0, a_0, \dots, a_{t-1}, s_t, a_t)$.

We note that our definition differs a bit from those found in the literature, where the environment policies have limited adaptiveness: they are either stationary [Ho *et al.*, 2021] or Markovian (i.e., time-dependent) [Wang *et al.*, 2023]. However, one of the corollaries of our results is that all three defi-

nitions lead to equivalent problems in the setting we consider.

We denote by $\mathbb{P}_{\mathcal{M}}^{\sigma, \pi}$ and $\mathbb{E}_{\mathcal{M}}^{\sigma, \pi}$ the probability measure and the expectation operator induced by the pair of policies σ, π over the space of infinite trajectories $s_0, a_0, s_1, a_1, \dots$ of \mathcal{M} .

(s,a)-rectangularity. In this work, we restrict to the study of *(s, a)-rectangular uncertainty sets* [Nilim and Ghaoui, 2003; Iyengar, 2005], which are the most commonly studied class of uncertainty sets and are of the form $\mathcal{P} = \prod_{(s,a) \in S \times A} \mathcal{P}_{(s,a)}$, where each $\mathcal{P}_{(s,a)} \subseteq \Delta(S)$. The assumption means that the environment can choose transition probabilities for each state-action pair independently from other state-action pairs.

Objectives We consider RMDPs with respect to *long-run average* (also known as *mean payoff*) and *discounted-sum* objectives. For each trajectory $\tau = s_0, a_0, s_1, a_1, \dots$, we define its long-run average payoff as

$$\text{LIMAVG}(\tau) = \liminf_{N \rightarrow \infty} \frac{1}{N+1} \cdot \sum_{i=0}^N r(s_i, a_i).$$

For every $\gamma \in (0, 1)$, we define its γ -discounted payoff as

$$\text{DISC}_{\gamma}(\tau) = \lim_{N \rightarrow \infty} \sum_{i=0}^N \gamma^i \cdot r(s_i, a_i).$$

Problem statement. The *value* of an RMDP \mathcal{M} given policies σ, π is the expected payoff incurred under the policies $\text{Val}(\mathcal{M}, \sigma, \pi) = \mathbb{E}_{\mathcal{M}}^{\sigma, \pi}[\text{PAYOFF}]$, where PAYOFF is either LIMAVG or DISC_{γ} . The *optimal value* of an RMDP \mathcal{M} is the maximal expected payoff achievable by the agent against any environmental choices of transition functions:

$$\text{Val}^*(\mathcal{M}) = \sup_{\sigma} \inf_{\pi} \text{Val}(\mathcal{M}, \sigma, \pi).$$

An agent strategy σ^* is *optimal* if it achieves the value against any environment policy, i.e., if $\inf_{\pi} \text{Val}(\mathcal{M}, \sigma^*, \pi) = \text{Val}^*(\mathcal{M})$. Given an RMDP with either objective, our goal is to compute its value and a corresponding optimal policy.

Assumption: polytopic RMDPs. Towards making our problem computationally tractable, we need to assume a certain form of uncertainty sets in RMDPs. In this work, we consider a very general setting of *polytopic uncertainty sets*. An RMDP $\mathcal{M} = (S, A, \mathcal{P}, r, s_0)$ is said to be *polytopic* if, for each state-action pair $(s, a) \in S \times A$, the uncertainty set $\mathcal{P}(s, a) \subseteq \mathbb{R}^{|S|}$ is a polytope (i.e. a convex hull of finitely many points in $\mathbb{R}^{|S|}$). These strictly subsume RMDPs with L^1 -uncertainty (or total variation) sets [Ho *et al.*, 2021], interval MDPs [Givan *et al.*, 2000] and contamination models [Wang *et al.*, 2023]. Furthermore, all our results are applicable to uncertainty sets which are *subsets of polytopes* so long as all polytope vertices are within the uncertainty set.

3 Reduction to Turn-based Stochastic Games

This section presents the main result of this work. We show that the problem of solving long-run average reward polytopic RMDPs admits a linear-time reduction to the problem of solving long-run average reward turn-based stochastic games. The reduction is linear in the size of the RMDP and the number of vertices of uncertainty polytopes. In what follows, we

first recall turn-based stochastic games. We then formally define and prove our reduction, which is the main theoretical result of this work. Finally, we use the reduction to obtain a sequel of novel results on RMDPs by leveraging results on turn-based stochastic games.

3.1 Background on Turn-based Stochastic Games

Turn-based stochastic games are a standard model of decision making in the presence of both adversarial agent and stochastic uncertainty [Shapley, 1953; Filar and Vrieze, 1996; Kučera, 2011]. Formally, a (finite-state, two-player, zero-sum) *turn-based stochastic game (TBSG)* is a tuple $\mathcal{G} = (S, A, \delta, r, s_0)$ where all the elements are defined as in MDPs, with the additional requirement that the state set S is partitioned into sets S_{\max} and S_{\min} and the action set A is partitioned into sets A_{\max} and A_{\min} . We say that these vertices and actions belong to players *Max* and *Min*, respectively. Similarly to MDPs and RMDPs, the dynamics of TBSGs are defined through the notion of policies for each player. We distinguish between Max- and Min-histories depending on whether the history ends in a state belonging to Max or Min.

The value of a policy pair, optimal value and optimal policy for long-run average and discounted-sum objectives are defined analogously as for RMDPs. We denote by Σ and Π the sets of all policies of Max and Min, respectively. Similarly, we denote by Σ_p and Π_p the sets of all pure positional policies of Max and Min, respectively. The following is a classical result on TBSGs that will be an important ingredient in constructing our reduction from polytopic RMDPs. Intuitively, it says that both players can achieve the optimal expected payoff by using pure positional policies. For long-run average objective, the result was proved by [Gillette, 1957; Liggett and Lippman, 1969]. For discounted-sum, positional determinacy was proved by [Shapley, 1953], pure positional determinacy is a simple consequence [Bertrand *et al.*, 2023].

Theorem 1 (Pure positional determinacy). *Given a TBSG \mathcal{G} , the following equality holds for both long-run average and discounted-sum objectives:*

$$\begin{aligned} \text{Val}^*(\mathcal{G}) &= \sup_{\sigma \in \Sigma} \inf_{\pi \in \Pi} \text{Val}(\mathcal{G}, \sigma, \pi) = \inf_{\pi \in \Pi} \sup_{\sigma \in \Sigma} \text{Val}(\mathcal{G}, \sigma, \pi) \\ &= \max_{\sigma \in \Sigma_p} \min_{\pi \in \Pi_p} \text{Val}(\mathcal{G}, \sigma, \pi) = \min_{\pi \in \Pi_p} \max_{\sigma \in \Sigma_p} \text{Val}(\mathcal{G}, \sigma, \pi). \end{aligned}$$

Comparison to RMDPs. While the adversarial structure of TBSGs mimics the one of RMDPs, there is a crucial distinction: in TBSGs, both players may only select from *finitely many actions* at each step, while in RMDPs, the adversary can choose from a possibly *continuous set of distributions*. Hence, RMDPs do not immediately reduce to TBSGs.

3.2 Reduction

Intuition. A polytopic RMDP \mathcal{M} induces a TBSG $\mathcal{G}_{\mathcal{M}}$ as follows. The TBSG $\mathcal{G}_{\mathcal{M}}$ is played between the agent and the environment players, which alternate in turns. The goal of the agent is to maximize long-run average reward, whereas the goal of the environment is to minimize it. Hence, the agent is the Max player and the environment is the Min player.

The agent and the environment players alternate in turns. First, the TBSG is in a state belonging to the agent player and

the agent player selects an action to play. The state and the action sets belonging to the agent player in the TBSG coincide with the state and the action sets of the agent in the RMDP. Thus, the TBSG dynamics in states belonging to the agent player simulate the RMDP dynamics. On the other hand, the state and the action sets belonging to the environment player in the TBSG coincide with the set of all state-action pairs and the set of all vertices of uncertainty polytopes in the RMDP, respectively. Thus, the TBSG dynamics in states belonging to the environment player simulate the process in which a probabilistic transition function from which the next state will be sampled is selected from the uncertainty polytope in the RMDP. Note that pure strategies of the environment player can only select probabilistic transition functions that are vertices of the uncertainty polytopes, however any other element of the uncertainty polytope can be selected by using randomized strategies. Finally, the reward function of the TBSG also simulates that of the RMDP – the rewards of state-action pairs belonging to the adversary player in the TBSG are twice the rewards in the RMDP, whereas the rewards induced by the environment player’s moves in the TBSG are 0. Thus, under every policy pair, the TBSG intuitively induces an infinite play whose long-run average is the same as that in the RMDP. In what follows, we formalize this construction.

Formal definition. Let $\mathcal{M} = (S^{\mathcal{M}}, A^{\mathcal{M}}, \mathcal{P}^{\mathcal{M}}, r^{\mathcal{M}}, s_0^{\mathcal{M}})$ be a polytopic RMDP. We define a TBSG $\mathcal{G}_{\mathcal{M}} = (S^{\mathcal{G}_{\mathcal{M}}}, A^{\mathcal{G}_{\mathcal{M}}}, \delta^{\mathcal{G}_{\mathcal{M}}}, r^{\mathcal{G}_{\mathcal{M}}}, s_0^{\mathcal{G}_{\mathcal{M}}})$ as below, and say that $\mathcal{G}_{\mathcal{M}}$ is *induced by \mathcal{M}* . In what follows, for each state-action pair $(s, a) \in S^{\mathcal{M}} \times A^{\mathcal{M}}$ in the RMDP \mathcal{M} , we use $V_{s,a}^{\mathcal{M}}$ to denote the set of all vertices of the uncertainty polytope $\mathcal{P}^{\mathcal{M}}(s, a)$:

- **States.** $S^{\mathcal{G}_{\mathcal{M}}}$ is defined as a union of two disjoint sets $S_{\max}^{\mathcal{G}_{\mathcal{M}}}$ and $S_{\min}^{\mathcal{G}_{\mathcal{M}}}$, where $S_{\max}^{\mathcal{G}_{\mathcal{M}}} = S^{\mathcal{M}}$ and $S_{\min}^{\mathcal{G}_{\mathcal{M}}} = S^{\mathcal{M}} \times A^{\mathcal{M}}$ are copies of the sets of all states and all state-action pairs in \mathcal{M} .
- **Actions.** $A^{\mathcal{G}_{\mathcal{M}}}$ is defined as a union of two disjoint sets $A_{\max}^{\mathcal{G}_{\mathcal{M}}}$ and $A_{\min}^{\mathcal{G}_{\mathcal{M}}}$, where $A_{\max}^{\mathcal{G}_{\mathcal{M}}} = A^{\mathcal{M}}$ and $A_{\min}^{\mathcal{G}_{\mathcal{M}}} = \cup_{(s,a) \in S^{\mathcal{M}} \times A^{\mathcal{M}}} V_{s,a}^{\mathcal{M}}$ are copies of the sets of all actions and all vertices of uncertainty polytopes in \mathcal{M} .
- **Transition function.** $\delta^{\mathcal{G}_{\mathcal{M}}} : S_{\max}^{\mathcal{G}_{\mathcal{M}}} \times A_{\max}^{\mathcal{G}_{\mathcal{M}}} \cup S_{\min}^{\mathcal{G}_{\mathcal{M}}} \times A_{\min}^{\mathcal{G}_{\mathcal{M}}} \rightarrow \Delta(S^{\mathcal{G}_{\mathcal{M}}})$ is defined as follows:

- For state-action pairs $(s, a) \in S_{\max}^{\mathcal{G}_{\mathcal{M}}} \times A_{\max}^{\mathcal{G}_{\mathcal{M}}}$ belonging to Max, we let $\delta^{\mathcal{G}_{\mathcal{M}}}(s, a)$ be the Dirac probability distribution over $\Delta(S^{\mathcal{G}_{\mathcal{M}}})$ with all probability mass in the Min state $(s, a) \in S_{\min}^{\mathcal{G}_{\mathcal{M}}}$, i.e.

$$\delta^{\mathcal{G}_{\mathcal{M}}}(s, a)(s') = \begin{cases} 1, & \text{if } s' = (s, a) \in S_{\min}^{\mathcal{G}_{\mathcal{M}}}, \\ 0, & \text{otherwise.} \end{cases}$$

- For state-action pairs $((s, a), v) \in S_{\min}^{\mathcal{G}_{\mathcal{M}}} \times A_{\min}^{\mathcal{G}_{\mathcal{M}}}$ belonging to Min, we let $\delta^{\mathcal{G}_{\mathcal{M}}}(s, a)$ be the probability distribution over $S_{\max}^{\mathcal{G}_{\mathcal{M}}} = S^{\mathcal{M}}$ induced by vertex v of uncertainty polytope $\mathcal{P}^{\mathcal{M}}(s, a)$, i.e.

$$\delta^{\mathcal{G}_{\mathcal{M}}}((s, a), v)(s') = \begin{cases} v(s'), & \text{if } s' \in S_{\max}^{\mathcal{G}_{\mathcal{M}}}, \\ 0, & \text{otherwise.} \end{cases}$$

- **Reward function.** $r^{\mathcal{G}_{\mathcal{M}}} : S_{\max}^{\mathcal{G}_{\mathcal{M}}} \times A_{\max}^{\mathcal{G}_{\mathcal{M}}} \cup S_{\min}^{\mathcal{G}_{\mathcal{M}}} \times A_{\min}^{\mathcal{G}_{\mathcal{M}}} \rightarrow \mathbb{R}$ is defined as follows:

- For state-action pairs $(s, a) \in S_{\max}^{\mathcal{G}_M} \times A_{\max}^{\mathcal{G}_M}$, the reward is twice the reward incurred in the RMDP \mathcal{M} , i.e. $r^{\mathcal{G}_M}(s, a) = 2 \cdot r^{\mathcal{M}}(s, a)$.
- For state-action pairs $((s, a), v) \in S_{\min}^{\mathcal{G}_M} \times A_{\min}^{\mathcal{G}_M}$, the reward is 0, i.e. $r^{\mathcal{G}_M}((s, a), v) = 0$.
- *Initial state.* $s_0^{\mathcal{G}_M} = s_0^{\mathcal{M}} \in S_{\max}^{\mathcal{G}_M}$, i.e. the initial state in \mathcal{G}_M is the copy of the initial state in \mathcal{M} .

The following theorem establishes reduction correctness. In the proof, considering polytopic RMDPs allows us to soundly prune the action space of the environment up to corners of the polytope. This is then used to reduce the RMDP to a TBSG with finite state and action space.

Theorem 2 (Correctness). *Consider a polytopic RMDP $\mathcal{M} = (S^{\mathcal{M}}, A^{\mathcal{M}}, \mathcal{P}^{\mathcal{M}}, r^{\mathcal{M}}, s_0^{\mathcal{M}})$ and define a TBSG $\mathcal{G}_M = (S^{\mathcal{G}_M}, A^{\mathcal{G}_M}, \delta^{\mathcal{G}_M}, r^{\mathcal{G}_M}, s_0^{\mathcal{G}_M})$ as above. Then, for long-run average objective we have $Val^*(\mathcal{M}) = Val^*(\mathcal{G}_M)$.*

Proof sketch, full proof in [Chatterjee et al., 2023a]. Denote by $\Sigma^{\mathcal{M}}$ and $\Pi^{\mathcal{M}}$ the sets of all policies of the agent and the environment in the polytopic RMDP, and similarly define $\Sigma^{\mathcal{G}_M}$ and $\Pi^{\mathcal{G}_M}$ in the TBSG. We use $\Sigma_p^{\mathcal{M}}, \Pi_p^{\mathcal{M}}, \Sigma_p^{\mathcal{G}_M}$ and $\Pi_p^{\mathcal{G}_M}$ to denote the subsets of pure positional policies. In the full proof, we first define mappings from the sets of agent and environment policies in the polytopic RMDP to the sets of agent player and environment player policies in the TBSG:

$$\Phi : \Sigma^{\mathcal{M}} \rightarrow \Sigma^{\mathcal{G}_M}, \quad \Psi : \Pi^{\mathcal{M}} \rightarrow \Pi^{\mathcal{G}_M},$$

We then show that these mappings preserve the values of policy pairs under long-run average objective, i.e. that

$$Val(\mathcal{M}, \sigma, \pi) = Val(\mathcal{G}_M, \Phi(\sigma), \Psi(\pi))$$

hold for each $\sigma \in \Sigma^{\mathcal{M}}$ and $\pi \in \Pi^{\mathcal{M}}$. Finally, we show that when restricted to pure positional policies, these mappings give rise to *one-to-one correspondences* (i.e. *bijections*)

$$\Phi_p : \Sigma_p^{\mathcal{M}} \rightarrow \Sigma_p^{\mathcal{G}_M}, \quad \Psi_p : \Pi_p^{\mathcal{M}} \rightarrow \Pi_p^{\mathcal{G}_M}.$$

Since by Theorem 1 there exist pure positional policies that achieve the optimal value for each player in the TBSG, we can conclude that there exist pure positional policies that achieve the optimal value in the polytopic RMDP and that the optimal values in the polytopic RMDP and the TBSG coincide. \square

In order to study the computational complexity of the reduction, we first need to fix a representation of polytopic RMDPs. In particular, we assume that each polytopic RMDP is represented as an ordered tuple $\mathcal{M} = (S^{\mathcal{M}}, A^{\mathcal{M}}, \mathcal{P}^{\mathcal{M}}, r^{\mathcal{M}}, s_0^{\mathcal{M}})$, where $S^{\mathcal{M}}$ is a list of states, $A^{\mathcal{M}}$ is a list of actions, $\mathcal{P}^{\mathcal{M}}$ is a list of polytope vertices for each state-action pair, $r^{\mathcal{M}}$ is a list of rewards for each state-action pair and $s_0^{\mathcal{M}}$ is an element of the state list. The following theorem shows that the above is a linear-time reduction in the size of the RMDP and the number of vertices of uncertainty polytopes.

Theorem 3 (Complexity, proof in [Chatterjee et al., 2023a]). *Consider a polytopic RMDP $\mathcal{M} = (S^{\mathcal{M}}, A^{\mathcal{M}}, \mathcal{P}^{\mathcal{M}}, r^{\mathcal{M}}, s_0^{\mathcal{M}})$ and define a TBSG $\mathcal{G}_M = (S^{\mathcal{G}_M}, A^{\mathcal{G}_M}, \delta^{\mathcal{G}_M}, r^{\mathcal{G}_M}, s_0^{\mathcal{G}_M})$ as above. Then, the size of \mathcal{G}_M is linear in the size of \mathcal{M} and the number of vertices of uncertainty polytopes.*

3.3 Discussion and Implications

Our reduction enables us to leverage the rich repository of results about TBSGs achieved within the computational game theory and formal methods communities. Since our reduction yields a well-defined and effective correspondence between policies in the polytopic RMDP and the TBSG, the results for TBSGs naturally carry over to the polytopic RMDP setting. In this subsection we highlight several such results, focusing on the issues of complexity and efficient algorithms.

Pure positional determinacy. First, as an immediate corollary of our reduction and of Theorem 1, we get the following result. While this result was already established for polytopic RMDPs by a direct analysis and without the reduction to TBSGs [Goyal and Grand-Clément, 2023], we state it below as it will be used for proving subsequent results.

Corollary 1. *In a polytopic RMDP under long-run average objective, both the agent and the environment have pure positional optimal policies.*

Problem complexity. The decision problems associated with solving long-run average reward TBSGs (such as deciding whether the optimal value of the game is at least a given threshold v) are known to be in $\text{NP} \cap \text{coNP}$. This is due to Theorem 1: for NP membership, one can guess an optimal pure positional policy of, say player Max, and then verify that it enforces value $\geq v$: the verification can be done in polynomial time since fixing a pure positional policy in an RMDP yields a standard MDP that can be solved in polynomial time e.g. by linear programming [Puterman, 1994]. The argument for coNP membership is dual. By Corollary 1, the same characterization holds for RMDPs.

Corollary 2. *Given an RMDP \mathcal{M} under long-run average objective and a threshold $v \in \mathbb{Q}$, the problem whether $Val^*(\mathcal{M}) \geq v$ belongs to $\text{NP} \cap \text{coNP}$.*

Efficient algorithms. *Simple stochastic games (SSGs)* [Condon, 1992] are a special class of TBSGs with reachability objectives that captures the complexity of the whole TBSG class. It was shown by [Andersson and Miltersen, 2009] that solving TBSGs with both discounted and long-run average payoffs is polynomial-time reducible to solving SSGs (where by solving we mean computing optimal values and policies).

While SSGs are not known to be solvable in polynomial time, several sub-exponential algorithms for them were developed. The randomized algorithm due to [Ludwig, 1995] solves a given SSG in time $\mathcal{O}(2^{\sqrt{\min\{|S_{\max}|, |S_{\min}|\}}}} \cdot \text{poly}(|\mathcal{G}|))$, where $|\mathcal{G}|$ is the encoding size of the game. Neither our reduction from RMDPs to TBSGs nor the reduction from TBSGs to SSGs [Andersson and Miltersen, 2009] add any player Max vertices. Hence, we get the following result.

Corollary 3. *There is a randomized algorithm for solving long-run average reward polytopic RMDPs whose expected runtime is $\mathcal{O}(2^{\sqrt{|S|}} \cdot \text{poly}(|\mathcal{M}|))$, i.e., sub-exponential.*

Remark 1 (Discounted polytopic RMDPs). *In the extended version of the paper [Chatterjee et al., 2023a], we show the above reduction can be modified to also work for discounted-sum objectives. Hence Corollaries 1–3 hold for discounted-sum objectives as well.*

In [Chatterjee *et al.*, 2023b] authors provide an algorithm for solving TBSGs parameterized by the treewidth of the underlying graph. The algorithm has runtime $\mathcal{O}((tn^2 \log nW)^{t \log n})$ where n is the number of nodes, t is the treewidth of the graph and W is the maximum absolute value of rewards in the game. Let \mathcal{M} be an RMDP with m actions and whose underlying graph has treewidth t . In the extended version of the paper [Chatterjee *et al.*, 2023a], we show that $\mathcal{G}_{\mathcal{M}}$ has treewidth at most $(t+1)(m+1)-1$, which implies the following result.

Corollary 4. *Given an RMDP \mathcal{M} with constant number of actions and treewidth of the underlying graph, the long-run average objective can be solved in quasi-polynomial time.*

4 Algorithm for Long-run Average RMDPs

In this section, we present our algorithm *Robust Polytopic Policy Iteration (RPPI)* for computing long-run average values and optimal pure positional policies in polytopic RMDPs. The pseudocode of our algorithm is shown in Algorithm 1. Our algorithm can be viewed as a policy iteration algorithm for solving long-run average reward polytopic RMDPs. Policy iteration was hitherto *not known to be sound* for solving long-run average reward RMDPs, thus the design of a policy iteration algorithm is another contribution of our work.

Outline. RPPI uses the reduction in Section 3 to reduce our problem to solving long-run average TBSGs, for which policy iteration is known to be sound and highly efficient [Hansen *et al.*, 2013]. Our RPPI is thus motivated by efficient implementations of policy iteration for TBSGs, which utilize Blackwell optimality and use policy iteration to solve a number of discounted-sum TBSGs while increasing the discount factor until a discount factor for which discounted-sum and long-run average policies are equal is achieved.

Algorithm details. Algorithm 1 takes as input a polytopic RMDP \mathcal{M} (line 1). It then constructs a TBSG $\mathcal{G}_{\mathcal{M}}$ as in Section 3 to reduce our problem to solving a long-run average TBSG (line 2), initializes a discount factor $\gamma = \frac{1}{2}$ (line 3) and performs policy iteration for long-run average TBSGs (lines 4-10). For each value of the discount factor γ , Algorithm 1 first solves the discounted-sum TBSG and computes a pair of optimal policies by using the discounted-sum TBSG policy iteration algorithm of [Hansen *et al.*, 2013], which runs in polynomial time, as an off-the-shelf subprocedure (line 5). It then uses Policy Profile Evaluation (PPE, see details below) to check if this policy profile is optimal for the LIMAVG objective as well (line 6). If the policy profile is found to be optimal, then the algorithm breaks the loop (lines 7-8) and returns the agent policy together with its LIMAVG value (line 12-13). Otherwise, the discount factor γ is increased (line 10) and Algorithm 1 proceeds to solving the next discounted-sum TBSG (line 4-10). By Blackwell optimality for TBSGs, Algorithm 1 is guaranteed to eventually terminate and find a pair of optimal pure positional policies for each player in the TBSG $\mathcal{G}_{\mathcal{M}}$. By the one-to-one correspondence between pure positional policies in the RMDP \mathcal{M} and the TBSG $\mathcal{G}_{\mathcal{M}}$ established in the proof of Theorem 2 and in Corollary 1, the agent player policy then gives rise to an optimal pure positional agent policy in the RMDP \mathcal{M} , as desired.

Algorithm 1 Robust Polytopic Policy Iteration (RPPI)

```

1: Input polytopic RMDP  $\mathcal{M}$ 
2:  $\mathcal{G}_{\mathcal{M}} \leftarrow$  TBSG by reduction from  $\mathcal{M}$  as in Section 3
3:  $\gamma \leftarrow \frac{1}{2}$  {initial discount factor}
4: while True do
5:    $\sigma, \pi \leftarrow$  optimal discounted-sum policies in  $\mathcal{G}_{\mathcal{M}}$  with
     discount factor  $\gamma$ , by method of [Hansen et al., 2013]
6:    $is\_optimal, rewards \leftarrow$  PPE( $\mathcal{G}_{\mathcal{M}}, \sigma, \pi$ )
7:   if  $is\_optimal$  then
8:     break
9:   end if
10:   $\gamma \leftarrow \frac{1+\gamma}{2}$ 
11: end while
12:  $Val^*(\mathcal{M}) = rewards[s_0]$ 
13: Return  $Val^*(\mathcal{M}), \sigma$ 

```

Policy profile evaluation. We now describe the PPE subprocedure used for policy profile evaluation in line 6 of Algorithm 1. PPE takes as input the TBSG $\mathcal{G}_{\mathcal{M}}$ and policies σ and π of Max and Min players. In order to check whether σ and π are optimal for the long-run average objective, by pure positional determinacy of TBSGs (Theorem 1) it suffices to checks if they provide optimal responses to each other.

To check if σ and π provide optimal responses to each other, PPE first considers an MDP obtained by fixing policy π of the Min player and solves the long-run average MDP to compute the optimal long-run average payoff of Max player against π . Similarly, PPE then considers the MDP obtained by fixing policy σ of the Max player and solves the long-run average MDP to compute the optimal long-run average payoff of Min player against σ . In both cases, the long-run average MDP can be solved via linear programming or policy iteration [Puterman, 1994]. If two MDPs have equal long-run average payoffs, then we conclude that two policies are optimal responses to each other, otherwise they are not.

The following theorem establishes the correctness of our RPPI algorithm (and the PPE subprocedure). Its proof follows from the correctness of our reduction (Theorem 2), Blackwell optimality for TBSGs [Andersson and Miltersen, 2009], and the correctness of all involved subprocedures for which we use off-the-shelf approaches.

Theorem 4. *Given a polytopic RMDP \mathcal{M} , Algorithm 1 returns the optimal long-run average value $Val^*(\mathcal{M})$ and an optimal policy for the agent.*

5 Experimental Results

We implement RPPI (Algorithm 1) and compare it against two state of the art value iteration-based methods for solving long-run average reward RMDPs with uncertainty sets not being intervals or L^1 -balls, towards demonstrating the significant computational runtime gains provided by a policy iteration-based algorithm. Furthermore, we demonstrate the applicability of our method to non-unichain polytopic RMDPs to which existing algorithms are not applicable. Our implementation is publicly available at <https://github.com/mehrdad76/RMDP-LRA>.

Implementation details. We implement the policy iteration for discounted-sum TBSPs of [Hansen *et al.*, 2013] to compute optimal strategies in line 5 in Algorithm 1. Furthermore, we use the probabilistic model checker Storm [Hensel *et al.*, 2022] as an off-the-shelf tool to compute values in long-run average reward MDPs in the Policy Profile Evaluation sub-procedure. Due to numerical precision issues in Python and Storm, in PPE we replace the equality by $|max_rewards - min_rewards| < 10^{-5}$. All experiments were run in Python 3.9 on a Ubuntu 22.04 machine with an octa-core 2.40 GHz Intel Core i5 CPU, 16 GB RAM.

Baselines. We compare the runtime of our RPPI against two value iteration-based algorithms: Robust Value Iteration (RVI) and Robust Relative Value Iteration (RRVI), both proposed in [Wang *et al.*, 2023]. Similarly to our RPPI, RVI uses Blackwell optimality to compute the long-run average value and optimal policy by solving discounted-sum reward MDPs and gradually increasing the discount factor. On the other hand, RRVI is based on a value iteration procedure that directly solves the long-run average reward RMDP.

Both RVI and RRVI impose *structural restrictions* on the RMDP. RVI requires the RMDP to be a unichain. RRVI requires the RMDP to be unichain and aperiodic. In contrast, our RPPI does not impose any of these assumptions.

Moreover, while RVI and RRVI are guaranteed to converge in the limit, they do not provide a stopping criteria for practical implementation. Hence, in our comparison, we run them both until the absolute difference between their computed values and our computed value is at most 10^{-3} .

Benchmarks. We consider the following benchmarks:

1. Contamination Model. Contamination Model is a conservative model of an RMDP defined over a given MDP where, at each step, the probability of taking the prescribed MDP transition function is $R \in [0, 1]$ whereas with probability $1 - R$ the environment gets to choose any probability distribution over the states. In our evaluation, we consider a Contamination Model taken from [Wang *et al.*, 2023]. For a given natural number n , the base MDP contains n states and $n + 10$ actions. The transition function and rewards for each state-action pair are then randomly generated (see [Wang *et al.*, 2023] for details). The uncertainty set $\mathcal{P}_{(s,a)}$ for state-action pair (s, a) is $\mathcal{P}_{(s,a)} = \{(1 - R)\delta_{(s,a)} + R \cdot p \mid p \in \Delta(S)\}$. The resulting RMDP is polytopic, unichain and aperiodic, hence both baselines are applicable to it.

2. Frozen Lake. This benchmark modifies the Frozen Lake environment in the OpenAI Gym [Towers *et al.*, 2023] to turn it into an RMDP. Consider an $n \times n$ grid with some cells being holes. An agent starts at the top left corner and chooses an action from $\{left, right, up, down\}$ to move to the next cell in that direction. However, the grid is slippery and the agent might move in a direction perpendicular to the chosen direction. If the agent falls into a hole, they are stuck and remain there with probability 1. The reward increases as the agent gets closer to the bottom right corner of the grid, so the agent should get to and stay around the bottom right corner. We turn this model into a polytopic RMDP by allowing the adversarial environment to perturb the transition probabilities by increasing the probability of moving to one adjacent cell

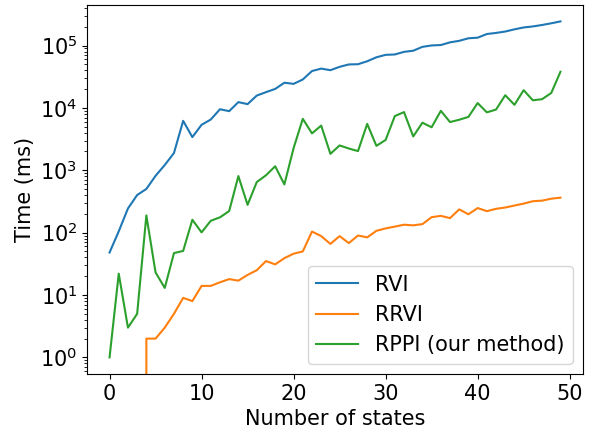


Figure 1: Runtime comparison on the Contamination Model.

by at most $d = 0.2$ while decreasing the remaining probabilities uniformly. Note that this RMDP is aperiodic and multichain, hence neither RVI nor RRVI baselines are applicable. We use this environment to evaluate our RPPI on multichain polytopic RMDPs to which existing methods are not applicable. In order to allow for a comparison, we also consider a unichain variant of the Frozen Lake environment where the holes are treated as "walls" and cannot be visited. To distinguish between the two variants, we refer to them as *Unichain* and *Multichain* Frozen Lake models, respectively.

Results on Contamination Model. We compared our method against both baselines and our results are shown in Figure 1. As indicated by the results, our RPPI is considerably faster than RVI. Given the similar structure of RPPI and RVI with the main difference being the use policy instead of value iteration, our results confirm the significant gains in computational runtime provided by our reduction to TBSPs which allowed the design of a policy iteration algorithm. On the other hand, we observe that the RRVI baseline is faster than both our method and the RVI baseline. The explanation for this is that RRVI is an algorithm specifically tailored to unichain aperiodic RMDPs, whereas our RPPI imposes no structural restrictions on the RMDP.

Results on Frozen Lake. Table ?? shows our results on the Frozen Lake model. As mentioned before, due to RMDP structural restrictions imposed by the baselines, RVI is only applicable to the Unichain model whereas RRVI is applicable to neither of the models. Our RPPI imposes no structural restrictions on the RMDP and is thus applicable to both models. As we can see from Table ?? our policy iteration-based RPPI is significantly faster than RVI, with an average runtime of 7.8 seconds. In contrast, the RVI method takes more than 2000 seconds on average, and times out on the 10×10 model. Moreover, our method successfully solves all the Multichain model instances in less than 4 minutes each.

Concluding remarks. We considered the problem of solving long-run average reward RMDPs with polytopic uncertainty sets and proposed a new perspective on this problem by establishing a linear-time reduction to long-run average

n	Unichain		Multichain
	RPPI Time (s)	VI Time (s)	RPPI Time (s)
2	0.0	0.15	0.0
3	0.0	8.3	2.6
4	0.0	156.8	0.6
5	0.1	106.6	4.0
6	6.1	1232.6	53.7
7	17.4	3347.8	119.0
8	3.7	7383.6	104.8
9	19.2	6496.6	93.0
10	23.7	Timeout	231.4
Average	7.8	2341.5	67.6
STDEV	9.5	3061.8	78.2

Table 1: Runtime comparison on the Frozen Lake models.

reward turn-based stochastic games. This reduction allowed us to leverage results from the stochastic games literature and obtain a number of results on computational complexity and efficient algorithms that were hitherto not known to hold for long-run average polytopical RMDPs. Our work leaves several interesting venues for future work. A natural question to consider is whether and how RMDPs with non-rectangular uncertainty sets could be related to stochastic games. Another direction would be to consider the connection between RMDPs with non-polytopical uncertainty sets and stochastic games, e.g. when uncertainty sets are specified in terms of KL-divergence from some nominal distribution [Wang *et al.*, 2023]. The extension to non-polytopical uncertainty sets via either (a) approximating non-polytopical with polytopical uncertainty sets and applying our method, or (b) designing new approaches, are interesting directions of future work.

Acknowledgements

This work was supported in part by the ERC-2020-CoG 863818 (FoRM-SMArt) and the Czech Science Foundation grant no. GA23-06963S.

References

- [Andersson and Miltersen, 2009] Daniel Andersson and Peter Bro Miltersen. The complexity of solving stochastic games on graphs. In Yingfei Dong, Ding-Zhu Du, and Oscar Ibarra, editors, *Algorithms and Computation*, pages 112–121, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.
- [Bertrand *et al.*, 2023] Nathalie Bertrand, Patricia Bouyer-Decitre, Nathanaël Fijalkow, and Mateusz Skomra. Stochastic games. In Nathanaël Fijalkow, editor, *Games on Graphs*. 2023.
- [Chatterjee *et al.*, 2023a] Krishnendu Chatterjee, Ehsan Kafshdar Goharshady, Mehrdad Karrabi, Petr Novotný, and Dorde Zikelic. Solving long-run average reward robust mdps via stochastic games. *CoRR*, abs/2312.13912, 2023.
- [Chatterjee *et al.*, 2023b] Krishnendu Chatterjee, Tobias Meggendorfer, Raimundo Saona, and Jakub Svoboda. Faster algorithm for turn-based stochastic games with bounded treewidth. In *SODA*, pages 4590–4605. SIAM, 2023.
- [Condon, 1992] A. Condon. The Complexity of Stochastic Games. *Information and Computation*, 96(2):203–224, 1992.
- [Filar and Vrieze, 1996] J. Filar and K. Vrieze. *Competitive Markov Decision Processes*. 1996.
- [Ghavamzadeh *et al.*, 2016] Mohammad Ghavamzadeh, Marek Petrik, and Yinlam Chow. Safe policy improvement by minimizing robust baseline regret. In Daniel D. Lee, Masashi Sugiyama, Ulrike von Luxburg, Isabelle Guyon, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pages 2298–2306, 2016.
- [Gillette, 1957] G. Gillette. Stochastic Games with Zero Stop Probabilities. *Contributions to the Theory of Games, vol III*, pages 179–187, 1957.
- [Givan *et al.*, 2000] Robert Givan, Sonia M. Leach, and Thomas L. Dean. Bounded-parameter markov decision processes. *Artif. Intell.*, 122(1-2):71–109, 2000.
- [Goyal and Grand-Clément, 2023] Vineet Goyal and Julien Grand-Clément. Robust markov decision processes: Beyond rectangularity. *Math. Oper. Res.*, 48(1):203–226, 2023.
- [Grand-Clément and Petrik, 2023] Julien Grand-Clément and Marek Petrik. Reducing blackwell and average optimality to discounted mdps via the blackwell discount factor. *CoRR*, abs/2302.00036, 2023.
- [Grand-Clement *et al.*, 2023] Julien Grand-Clement, Marek Petrik, and Nicolas Vieille. Beyond discounted returns: Robust markov decision processes with average and blackwell optimality. *arXiv preprint arXiv:2312.03618*, 2023.
- [Hansen *et al.*, 2013] Thomas Dueholm Hansen, Peter Bro Miltersen, and Uri Zwick. Strategy iteration is strongly polynomial for 2-player turn-based stochastic games with a constant discount factor. *J. ACM*, 60(1):1:1–1:16, 2013.
- [Hensel *et al.*, 2022] Christian Hensel, Sebastian Junges, Joost-Pieter Katoen, Tim Quatmann, and Matthias Volk. The probabilistic model checker storm. *Int. J. Softw. Tools Technol. Transf.*, 24(4):589–610, 2022.
- [Ho *et al.*, 2018] Chin Pang Ho, Marek Petrik, and Wolfram Wiesemann. Fast bellman updates for robust mdps. In Jennifer G. Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 1984–1993. PMLR, 2018.
- [Ho *et al.*, 2021] Chin Pang Ho, Marek Petrik, and Wolfram Wiesemann. Partial policy iteration for Π -robust markov decision processes. *J. Mach. Learn. Res.*, 22:275:1–275:46, 2021.
- [Iyengar, 2005] Garud N. Iyengar. Robust dynamic programming. *Math. Oper. Res.*, 30(2):257–280, 2005.

- [Kaufman and Schaefer, 2013] David L. Kaufman and Andrew J. Schaefer. Robust modified policy iteration. *INFORMS J. Comput.*, 25(3):396–410, 2013.
- [Kučera, 2011] A. Kučera. Turn-Based Stochastic Games. In *K.R. Apt, E. Grädel (Eds.): Lectures in Game Theory for Computer Scientists*, pages 146–184. Cambridge University Press, 2011.
- [Liggett and Lippman, 1969] T.M. Liggett and S.A. Lippman. Stochastic Games with Perfect Information and Time Average Payoff. *SIAM Review*, 11(4):604–607, 1969.
- [Ludwig, 1995] W. Ludwig. A Subexponential Randomized Algorithm for the Simple Stochastic Game Problem. 117(1):151–155, 1995.
- [Nilim and Ghaoui, 2003] Arnab Nilim and Laurent El Ghaoui. Robustness in markov decision problems with uncertain transition matrices. In Sebastian Thrun, Lawrence K. Saul, and Bernhard Schölkopf, editors, *Advances in Neural Information Processing Systems 16 [Neural Information Processing Systems, NIPS 2003, December 8-13, 2003, Vancouver and Whistler, British Columbia, Canada]*, pages 839–846. MIT Press, 2003.
- [Puterman, 1994] Martin L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Wiley Series in Probability and Statistics. Wiley, 1994.
- [Roy et al., 2017] Aurko Roy, Huan Xu, and Sebastian Pokutta. Reinforcement learning under model mismatch. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 3043–3052, 2017.
- [Shapley, 1953] L.S. Shapley. Stochastic Games. *Proceedings of the National Academy of Sciences*, 39:1095–1100, 1953.
- [Sinha and Ghate, 2016] Saumya Sinha and Archis Ghate. Policy iteration for robust nonstationary markov decision processes. *Optim. Lett.*, 10(8):1613–1628, 2016.
- [Sutton et al., 1999] Richard S. Sutton, David A. McAllester, Satinder Singh, and Yishay Mansour. Policy gradient methods for reinforcement learning with function approximation. In Sara A. Solla, Todd K. Leen, and Klaus-Robert Müller, editors, *Advances in Neural Information Processing Systems 12, [NIPS Conference, Denver, Colorado, USA, November 29 - December 4, 1999]*, pages 1057–1063. The MIT Press, 1999.
- [Tamar et al., 2014] Aviv Tamar, Shie Mannor, and Huan Xu. Scaling up robust mdps using function approximation. In *Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014*, volume 32 of *JMLR Workshop and Conference Proceedings*, pages 181–189. JMLR.org, 2014.
- [Tessler et al., 2019] Chen Tessler, Yonathan Efroni, and Shie Mannor. Action robust reinforcement learning and applications in continuous control. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 6215–6224. PMLR, 2019.
- [Tewari and Bartlett, 2007] Ambuj Tewari and Peter L. Bartlett. Bounded parameter markov decision processes with average reward criterion. In Nader H. Bshouty and Claudio Gentile, editors, *Learning Theory, 20th Annual Conference on Learning Theory, COLT 2007, San Diego, CA, USA, June 13-15, 2007, Proceedings*, volume 4539 of *Lecture Notes in Computer Science*, pages 263–277. Springer, 2007.
- [Towers et al., 2023] Mark Towers, Jordan K. Terry, Ariel Kwiatkowski, John U. Balis, Gianluca de Cola, Tristan Deleu, Manuel Goulão, Andreas Kallinteris, Arjun KG, Markus Krimmel, Rodrigo Perez-Vicente, Andrea Pierré, Sander Schulhoff, Jun Jet Tai, Andrew Tan Jin Shen, and Omar G. Younis. Gymnasium, March 2023.
- [Wang and Zou, 2021] Yue Wang and Shaofeng Zou. Online robust reinforcement learning with model uncertainty. In Marc’Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan, editors, *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 7193–7206, 2021.
- [Wang and Zou, 2022] Yue Wang and Shaofeng Zou. Policy gradient method for robust reinforcement learning. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvári, Gang Niu, and Sivan Sabato, editors, *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pages 23484–23526. PMLR, 2022.
- [Wang et al., 2023] Yue Wang, Alvaro Velasquez, George K. Atia, Ashley Prater-Bennette, and Shaofeng Zou. Robust average-reward markov decision processes. In Brian Williams, Yiling Chen, and Jennifer Neville, editors, *Thirty-Seventh AAAI Conference on Artificial Intelligence, AAAI 2023, Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence, IAAI 2023, Thirteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2023, Washington, DC, USA, February 7-14, 2023*, pages 15215–15223. AAAI Press, 2023.
- [Wiesemann et al., 2013] Wolfram Wiesemann, Daniel Kuhn, and Berç Rustem. Robust markov decision processes. *Math. Oper. Res.*, 38(1):153–183, 2013.
- [Yang et al., 2022] Wenhao Yang, Liangyu Zhang, and Zhihua Zhang. Toward theoretical understandings of robust markov decision processes: Sample complexity and asymptotics. *The Annals of Statistics*, 50(6):3223–3248, 2022.