SCALABLE ROBUST MCTS WITH ANYTIME DETERMINISTIC OPTIMALITY GUARANTEES

Anonymous authors
Paper under double-blind review

ABSTRACT

MDP planners often rely on nominal transition models that may differ from the true dynamics. Robust MDP planning accounts for this uncertainty through a set of admissible models, but doing so within the limited computation available online is challenging: the search tree grows exponentially with the horizon and each robust backup requires an additional optimization. Existing online robust methods remain limited in scale and offer at most probabilistic guarantees. We introduce, to the best of our knowledge, the first deterministic anytime guarantees for robust online MDP planning. We obtain these guarantees by restricting computation to the sampled search tree while deterministically bounding the contribution of unsampled trajectories. For any fixed policy, we derive lower and upper bounds on the gap between its projected and true robust values under both rectangular and non-rectangular uncertainty sets. Under rectangular uncertainty, we propagate these bounds through the robust Bellman operator to bound the optimal robust value. These certificates are computable from the current sampled tree, hold at every node, tighten as the sampled support expands, and converge to the robust values as full support is recovered. We develop CAR-MCTS, a bound-guided MCTS planner that uses these deterministic certificates to guide exploration, prune provably suboptimal branches, and terminate once the optimal root action is certified. The fixed-policy bounds also yield an anytime policy-evaluation mechanism that certifies the robust value of any given policy, independent of how it was obtained. Experiments verify the deterministic certificates, their tightening under support expansion, substantial branch pruning, root-action certification, and convergence to the optimal robust policy. To our knowledge, this is the first scalable robust MCTS framework with anytime deterministic guarantees for both online planning and policy evaluation, without enumerating the full state space.

1 INTRODUCTION

Markov decision processes (MDPs) provide a standard framework for sequential decision making under uncertainty, with the objective of finding a policy that maximizes the expected cumulative reward. As the state space or planning horizon grows, solving the MDP globally becomes computationally intractable. Online planning instead focuses computation on the current decision through a lookahead from the current state. Common online methods, such as Monte Carlo Tree Search (MCTS), address this lookahead through sampling, incrementally constructing and selectively expanding a partial search tree (Kocsis & Szepesvári, 2006). By allocating simulations to a subset of future outcomes, MCTS reduces the cost of online planning while refining root action-value.

Model-based online planning relies on a generative transition model that may differ from the true dynamics. Consequently, nominal value estimates may be inaccurate and nominally optimal policies may become suboptimal under model perturbations (Wiesemann et al., 2013). Robust Markov decision processes (RMDPs) address this sensitivity by optimizing against the worst-case model within a set of admissible transition models (Iyengar, 2005; Nilim & El Ghaoui, 2005), but each robust backup then requires an additional optimization over transition models. Repeating these backups throughout the online search tree under a limited decision-time budget further restricts how much of the tree can be explored.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

To improve scalability, robust RL has explored approaches such as function approximation and policy learning from sampled interactions under model uncertainty (Tamar et al., 2014; Wang & Zou, 2021; Panaganti & Kalathil, 2022). These methods generally aim to learn a robust policy over the state space, rather than allocating computation to the specific decision faced online. More recently, sample-based methods have targeted robust online planning directly, including Robust Sparse Sampling (RSS) (Shazman et al., 2026) and robust MCTS (Dam et al., 2026). However, RSS scales exponentially with the planning horizon, while Dam et al. (2026) robust MCTS relies on the full state space, hindering scalability in real world problems that often exhibit large state spaces.

Beyond scalability, online planning acts on a partially explored tree, leaving the effect of unexplored outcomes on the current decision unknown. This is particularly important in the robust setting, where such outcomes may alter the worst-case value and preferred action. For MCTS under a known model, Shah et al. (2022) provide finite-time probabilistic guarantees, which Dam et al. (2026) leverage to establish finite-sample guarantees for Robust-Power-UCT. Similarly, RSS provides probabilistic finite-sample guarantees for robust sample-based planning (Shazman et al., 2026). In a complementary line of work, Barenboim & Indelman (2023; 2026) establish deterministic anytime guarantees for sample-based anytime POMDP planning in the nominal setting. However, such deterministic anytime guarantees are not available for robust online planning, where the effect of unexplored outcomes must be assessed jointly with model uncertainty.

To address these gaps, we formulate robust online planning directly over the sampled portion of the search tree while explicitly accounting for the probability mass associated with unexplored outcomes. Restricting computation to the sampled tree preserves the scalability required for online planning. The resulting bounds quantify how much unexplored outcomes may still affect the current solution. This remaining uncertainty guides further search, focusing the limited planning budget where it can most effectively tighten the bounds and ultimately certify the decision.

Contributions. To the best of our knowledge, this is the first work to provide deterministic anytime bounds for sample-based online planning in robust MDPs. Our main contributions are:

- We derive deterministic anytime two-sided bounds on the robust value of a fixed policy under general and non-rectangular uncertainty sets. The bounds tighten monotonically as the sampled support expands and, in the non-robust setting, improve upon prior deterministic sample-based bounds by providing tighter lower and upper guarantees.
- Under rectangular uncertainty, we extend the fixed-policy bounds to deterministic anytime lower and upper certificates on the **optimal** robust value through robust Bellman recursion.
- We introduce **CAR-MCTS** (*Certified Anytime Robust MCTS*), an anytime robust planner with deterministic optimality guarantees computed solely from the constructed partial search tree. By avoiding enumeration of the full state space, CAR-MCTS remains applicable to large state spaces while retaining online computation. It uses the upper certificate to guide exploration and prunes provably suboptimal actions to focus computation on unresolved branches. Once the bounds certify the optimal root action, the search can terminate immediately and return the certified optimal decision.
- We empirically evaluate CAR-MCTS against existing robust online planners, demonstrating certificate tightening, pruning, root-action certification, and convergence toward the optimal robust solution.

2 Related Work

Robust online planning has received comparatively limited attention. Sharma et al. (2019) introduced RAMCP, a Monte Carlo planner for risk-sensitive Bayes-adaptive planning, but its formulation relies on a prior over a parametric family of transition models. Kohankhaki et al. (2024) developed UA-MCTS, which incorporates transition uncertainty into MCTS exploration; its empirical evaluation considers deterministic environments, while its theoretical analysis is limited to the corrupted-bandit setting. More closely related to our setting, Shazman et al. (2026) introduced Robust Sparse Sampling (RSS), which extends Sparse Sampling to RMDPs using Sample Average Approximation and provides finite-sample performance guarantees. Its complexity is independent of the state-space cardinality but grows exponentially with the planning horizon. Its analysis is further based on non-negative rewards and assumes the existence of a zero-reward fail-state that is absorb-

ing under every model in the uncertainty set. Dam et al. (2026) developed a robust MCTS planner that combines robust power-mean backups with exploration bonuses and establishes finite-sample convergence throughout the search tree. While its tree search is selective, the robust transition optimization retains an explicit dependence on the full state space, thereby limiting scalability as the state-space cardinality grows.

3 PRELIMINARIES

A robust MDP accounts for model uncertainty by replacing the single transition model with a set of admissible models. We consider a finite-horizon non-stationary RMDP $\mathcal{M}^{\text{rob}} = (\mathcal{S}, \mathcal{A}, \mathcal{P}^T_{0:H-1}, r, H)$, where $\mathcal{S}$ and $\mathcal{A}$, are the finite state and action spaces. The set $\mathcal{P}^T_{0:H-1}$ contains admissible non-stationary transition model sequences $P^T_{0:H-1}$, over the horizon H. The information available to the agent is summarized by the history of actions and states up to time t which is denoted by $h_t \triangleq (a_{0:t-1}, s_{0:t})$. The reward function $r : \mathcal{S} \times \mathcal{A} \to [r_{min}, r_{max}]$ is assumed to be bounded. A policy specifies how actions are selected based on the history. For a finite-horizon problem, a policy is a sequence $\pi = \{\pi_t\}_{t=0}^{H-1}$, where each π_t maps a history h_t to an action: $\pi_t : \mathcal{H}_t \to \mathcal{A}$. Given a sequence of nominal models $\mathring{P}^T_{0:H-1}$, such that the kernel at time step k is $\mathring{P}^T_k : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$, Let $\mathcal{P}^T_{t:H-1}$ denote a general set of admissible non-stationary transition model sequences. One common choice is a ball uncertainty set of the form:

$$\mathcal{P}^T_{t:H-1} = \{P^T_{t:H-1} : \; D\left(P^T_{t:H-1} || \mathring{P}^T_{t:H-1}\right) \leq \rho\}, \tag{1}$$

where $D(\cdot, \cdot)$ denotes a distance between sequences of transition kernels, potentially coupling deviations across time steps and state-action pairs, and ρ quantifies the uncertainty radius. The robust value function evaluates π under the worst-case admissible model, and is defined as:

$$V^{\pi,\text{rob}}_t(h_t) = \inf_{P^T_{t:H-1} \in \mathcal{P}^T_{t:H-1}} V^\pi_t(h_t; P^T_{t:H-1}), \tag{2}$$

where $V^\pi_t(h_t; P^T_{t:H-1})$ is the value function given a sequence of models. To reduce clutter, when clear from context, we omit the explicit dependence on $P^T_{t:H-1}$, and write $V^\pi_t(h_t; P^T_{t:H-1})$ as:

$$V^\pi_t(h_t) = r\big(s_t, \pi_t(h_t)\big) + \mathbb{E}^{\pi, P^T}_{s_{t+1:H-1}} \left[\sum_{j=t+1}^{H-1} r\big(s_j, \pi_j(h_j)\big) \mid h_t \right]. \tag{3}$$

The non-rectangular formulation in (1) and (2), forces the adversary to choose the model sequence jointly, so the admissibility of the model at one time may depend on choices at others. This coupling prevents a dynamic-programming decomposition of the Bellman backup, motivating the rectangular relaxation introduced by Iyengar (2005); Nilim & El Ghaoui (2005). The rectangular set at time step t is defined as $\mathcal{P}^{T,\text{rect}}_t = \bigotimes_{\substack{s \in \mathcal{S} \\ a \in \mathcal{A}}} \mathcal{P}^T_t(s, a)$, and the state-action uncertainty set is:

$$\mathcal{P}^T_t(s, a) = \left\{ P^T_t(\cdot|s, a) : D\left(P^T_t(\cdot|s, a) || \mathring{P}^T_t(\cdot|s, a)\right) \leq \rho_{s,a} \right\}. \tag{4}$$

Rectangularity is imposed across time and transition rows. First, nature may choose a different worst-case model at each stage independently of its choices at other stages. Second, the uncertainty set is (s, a)-rectangular, such that each transition row $P^T_t(\cdot \mid s, a)$ is selected independently. Since we represent the planning problem as a history tree, we additionally adopt a history-rectangular interpretation: each history-action node (h_t, a) is associated with an independent local uncertainty set $\mathcal{P}^T_t(s_t, a)$. Thus, nature may select different admissible transition rows at distinct history nodes. This represents an adversary that may adapt its models choices locally rather than committing in advance to a fixed sequence of models. Under this rectangular formulation, the robust value in (2) admits the recursive form:

$$V^{\pi,\text{rob}}_{t,\text{rect}}(h_t) = r\big(s_t, \pi_t(h_t)\big) + \inf_{P^T_t \in \mathcal{P}^T_t} \mathbb{E}^{P^T_t}_{s_{t+1}} \left[V^{\pi,\text{rob}}_{t+1,\text{rect}}(h_{t+1}) \right]. \tag{5}$$

The rectangular relaxation comes at a cost of a more conservative robust value: since $\mathcal{P}_{t:H} \subseteq \mathcal{P}^{rect}_{t:H}$, enlarging the uncertainty set can only decrease the robust value, yielding $V^{\pi,\text{rob}}_{t,\text{rect}}(h_t) \leq V^{\pi,\text{rob}}_t(h_t)$.

Monte Carlo Tree Search. Monte Carlo Tree Search (MCTS) incrementally constructs a finite lookahead tree rooted at the current history h_t through repeated simulations (Kocsis & Szepesvári,

2006). During each simulation, at a decision node h_k, $k \geq t$, an action is selected according to an exploration-strategy that balances exploration and exploitation, and a successor state is sampled from a generative transition model. If the resulting successor has not previously been encountered, the tree is expanded with the new history $h_{k+1} = (h_k, a_k, s_{k+1})$. The simulation then continues until the planning horizon is reached, after which the resulting values are backed up along the visited path. Repeating this process progressively expands the sampled portion of the planning tree while concentrating computation on branches favored by the search strategy. At decision time, the planner selects an action at the root based on the resulting action-value estimates. In our setting, this sampled tree defines the support of the simplified RMDP introduced in Section 4.

4 Simplified Robust MDP

Computing the exact optimal robust value online is generally intractable: the planning tree grows exponentially with the horizon, while each robust backup additionally requires optimization over the transition uncertainty set. We therefore construct a projected planning problem defined only over the portion of the tree represented by the current samples. Importantly, the projected transition probabilities are not renormalized; probability mass assigned to unsampled next-state outcomes is retained implicitly and will later be used to bound the approximation error deterministically. At a history-action node $h^-_{k+1} = (h_k, a_k)$, let $\widehat{\mathcal{S}}(h^-_{k+1}) \subseteq \mathcal{S}$ denote the sampled next-state support. For compactness, we use $s_{t+1:j} \in \widehat{\mathcal{S}}(h^-_j)$ to denote that $s_{k+1} \in \widehat{\mathcal{S}}(h^-_{k+1})$ for every $k = t, \ldots, j-1$.

4.1 Projected Transition Model

For each history-action node $h^-_{t+1} = (h_t, a_t)$, we define the projected transition model by restricting the full transition model to the sampled next-state support:

$$\widehat{P}^T_t(s_{t+1} \mid s_t, a_t; h^-_{t+1}) \triangleq P^T_t(s_{t+1} \mid s_t, a_t), \qquad s_{t+1} \in \widehat{\mathcal{S}}(h^-_{t+1}). \tag{6}$$

The projected probabilities are not renormalized. Hence, $\widehat{P}^T_t$ is generally a sub-probability vector, where the missing mass corresponds exactly to transitions leaving the sampled support. Restricting the transition model in this manner reduces each robust backup from an optimization over $\mathbb{R}^{|\mathcal{S}|}$ to one over $\mathbb{R}^{|\widehat{\mathcal{S}}(h^-_{t+1})|}$. This reduction makes each robust backup depend only on the sampled support and constitutes the main computational saving of the projected planner. Using these we define the simplified value function:

$$\widehat{V}^\pi_t(h_t) = \sum_{j=t}^{H-1} \sum_{s_{t+1:j} \in \widehat{\mathcal{S}}(h^-_j)} \left[\prod_{k=t}^{j-1} \widehat{P}^T_k\big(s_{k+1} \mid s_k, \pi_k(h_k)\big) \right] r\big(s_j, \pi_j(h_j)\big). \tag{7}$$

Unlike an ordinary value function, $\widehat{V}^\pi_t$ is not an expectation under a normalized transition model; it is the contribution of trajectories represented in the sampled tree.

4.2 Projected Uncertainty Set

Simply imposing the original uncertainty constraint directly in the lower-dimensional space does not, in general, preserve the set of models induced by the original RMDP. For a full transition model P^T_k, we define the restriction operator $\mathcal{R}^T_k(P^T_k; h_k) \triangleq \widehat{P}^T_k(s_{k+1}|s_k, a_k), \;\; s_{k+1} \in \widehat{\mathcal{S}}(h^-_{k+1})$. Let $\mathcal{R}^T_{t:H-1}$ apply this restriction over the planning horizon. We define the projected uncertainty set as the image of the original uncertainty set under this restriction:

$$\widehat{\mathcal{P}}^T_{t:H-1}(h_{H-1}) \triangleq \left\{ \mathcal{R}^T_{t:H-1}(P^T_{t:H-1}) : P^T_{t:H-1} \in \mathcal{P}^T_{t:H-1} \right\}. \tag{8}$$

Thus, every projected model in $\widehat{\mathcal{P}}^T_{t:H-1}$ is the restriction of at least one admissible full model; equivalently, its omitted components can be completed to a model belonging to $\mathcal{P}^T_{t:H-1}$. We define the resulting simplified-RMDP, as the tuple $\widehat{\mathcal{M}}^{\mathrm{rob}} = (\widehat{\mathcal{S}}, \mathcal{A}, \widehat{\mathcal{P}}^T_{t:H-1}, r, H)$. Given a policy π, its simplified robust value is the worst-case projected value over all projected models consistent with the original uncertainty set:

$$\widehat{V}^{\pi,\mathrm{rob}}_t(h_t) = \inf_{\widehat{P}^T_{t:H-1} \in \widehat{\mathcal{P}}^T_{t:H-1}} \widehat{V}^\pi_t(h_t; \widehat{P}^T_{t:H-1}). \tag{9}$$

Thus, $\widehat{V}_t^{\pi,\mathrm{rob}}$ applies the same worst-case model criterion as the original robust value, but evaluates only the trajectories represented in the sampled tree.

5 Anytime Deterministic Guarantees

Having defined the simplified RMDP, we now characterize the approximation induced by restricting planning to the sampled tree. We first derive deterministic anytime certificates for the robust value of an arbitrary fixed policy under general uncertainty sets. We then exploit rectangularity to propagate these certificates recursively through the robust Bellman optimality operator and obtain deterministic bounds on the optimal robust value.

5.1 Fixed-Policy Guarantees

We first consider an arbitrary policy π and quantify the error introduced by evaluating it on the simplified RMDP rather than the full RMDP. For a projected model sequence $\widehat{P}^T_{t:H-1}$, we define the cumulative in-support mass: $m_t^\pi\left(\widehat{P}^T_{t:H-1}\right) \triangleq \sum_{j=t}^{H-1}\sum_{s_{t+1:j}\in\widehat{\mathcal{S}}(h_j^-)}\prod_{k=t}^{j-1}\widehat{P}_k^T\left(s_{k+1}\mid s_k,\pi_k(h_k)\right)$. At each future step j, the inner sum is the probability mass that remains within the sampled tree up to j. Hence, m_t^π measures the cumulative coverage of the policy-induced future by the current sampled tree.

Theorem 1 (Fixed-policy deterministic bounds)**.** *Let h_t be the history at time t, and $V_t^{\pi,\mathrm{rob}}(h_t)$ and $\widehat{V}_t^{\pi,\mathrm{rob}}(h_t)$ denote the robust values of the same policy π in the full* (2) *and simplified* (9) *RMDPs, respectively. For any transition uncertainty set $\mathcal{P}^T_{t:H-1}$ and its projected counterpart $\widehat{\mathcal{P}}^T_{t:H-1}$:*

$$\mathcal{L}_t^\pi(h_t) \le V_t^{\pi,\mathrm{rob}}(h_t) - \widehat{V}_t^{\pi,\mathrm{rob}}(h_t) \;\le\; \mathcal{U}_t^\pi(h_t), \tag{10}$$

where

$$\mathcal{U}_t^\pi(h_t) \triangleq \sup_{\widehat{P}^T_{t:H-1}\in\widehat{\mathcal{P}}^T_{t:H-1}}\left[r_{max}\left(H-t-m_t^\pi\left(\widehat{P}^T_{t:H-1}\right)\right)\right], \tag{11}$$

$$\mathcal{L}_t^\pi(h_t) \triangleq \inf_{\widehat{P}^T_{t:H-1}\in\widehat{\mathcal{P}}^T_{t:H-1}}\left[r_{min}\left(H-t-m_t^\pi\left(\widehat{P}^T_{t:H-1}\right)\right)\right]. \tag{12}$$

The proof is provided in the Appendix A.1. The quantity $H-t-m_t^\pi$ is the cumulative probability mass not represented by the sampled tree over the remaining horizon. Since the reward is bounded as $r_{\min}\le r(s,a)\le r_{\max}$, Theorem 1 bounds the value contribution of the unsampled portion of the tree using the reward range. Importantly, the result does not rely on rectangularity or on a particular structure of the transition uncertainty set, and thus applies to general robust MDP formulations.

Corollary 1 (Asymptotic convergence)**.** *For any policy π, as the sampled state support expands to the full state space at every node $\widehat{\mathcal{S}}(h_k^-)\uparrow\mathcal{S},\ k=t,\dots,H$, the simplified robust value* (9) *convergences to the true robust value* (2)*:*

$$\widehat{V}_t^{\pi,\mathrm{rob}}(h_t)\to V_t^{\pi,\mathrm{rob}}(h_t). \tag{13}$$

The proof is provided in the Appendix A.6.

Corollary 2 (Monotonic tightening under support expansion)**.** *Let n denote the MCTS simulation index. Since the search tree is expanded incrementally, the sampled supports are nested across simulations: $\widehat{\mathcal{S}}^{(n)}(h_k^-)\subseteq\widehat{\mathcal{S}}^{(n+1)}(h_k^-),\ k=t,\dots,H$. Thus the deterministic bounds of Theorem 1 tighten monotonically:*

$$\left|\mathcal{L}_t^{\pi,(n+1)}(h_t)\right|\le\left|\mathcal{L}_t^{\pi,(n)}(h_t)\right|,\qquad\left|\mathcal{U}_t^{\pi,(n+1)}(h_t)\right|\le\left|\mathcal{U}_t^{\pi,(n)}(h_t)\right|. \tag{14}$$

Indeed, expanding the sampled support can only increase the in-support probability mass, such that $m_t^{\pi,(n)}\le m_t^{\pi,(n+1)}$. Consequently, the cumulative unrepresented mass $H-t-m_t^\pi$ decreases, and both bounds contract monotonically toward zero. Thus, adding sampled support can only improve the certificate. The convergence rate is governed by how quickly the planner captures probability mass in the sampled support. Different support-expansion strategies may therefore yield different rates of certificate tightening without affecting its validity.

Anytime Certified Policy Evaluation. The bounds in Theorem 1 depend only on the current sampled support, the projected uncertainty set, and the evaluated policy, and are therefore computable from the current partial tree provided the corresponding optimization over the projected uncertainty set can be solved. Consequently, they provide an anytime deterministic certificate on the robust value of any policy, including under non-rectangular uncertainty sets, independently of how that policy was obtained.

Recursive Form Under Rectangular Uncertainty. Rectangularity enables a recursive robust Bellman decomposition, allowing the lower and upper fixed-policy certificates to be propagated through the planning tree. Accordingly, the fixed-policy certificates of Theorem 1 admit a recursive action-conditioned form. The recursive quantity introduced below is directly related to the cumulative in-support mass m_t^π of Theorem 1. Under rectangular uncertainty, $1 + \delta_t^\pi(s_t \mid h_t) = \inf_{\widehat{P}^T_{t:H-1} \in \widehat{\mathcal{P}}^T_{t:H-1}} m_t^\pi(\widehat{P}^T_{t:H-1})$. The unit term corresponds to the current stage, while δ_t^π represents the worst-case cumulative in-support mass from the next step onward. Rectangularity allows this global minimization to be decomposed recursively as:

$$\delta_k^\pi(s_k|h_k) = \inf_{\widehat{P}^T_k \in \widehat{\mathcal{P}}^T_k} \delta_k^{\pi,\widehat{P}^T_k}(s_k, \pi_k(h_k)) \qquad k = t, \dots, H-2, \tag{15}$$

$$\delta_k^{\pi,\widehat{P}^T_k}(s_k, a_k|h_k) = \sum_{s_{k+1} \in \widehat{\mathcal{S}}(h^-_{k+1})} \widehat{P}^T_k(s_{k+1} \mid s_k, a_k) \left[1 + \delta_{k+1}^\pi(s_{k+1}|h_{k+1})\right], \tag{16}$$

with a terminal condition $\delta_{H-1}^\pi(s_{H-1}) \equiv 0$. The term $1 + \delta_{t+1}^\pi$ accounts for the probability mass retained at the next state together with all subsequent in-support mass along the continuation. The action-conditioned bounds are defined as:

$$\mathcal{L}^{\pi,\widehat{P}^T_t}_{t,rect}(h_t, a_t) \triangleq r_{min}\Big(H - 1 - t - \delta_t^{\pi,\widehat{P}^T_t}(s_t, a_t|h_t)\Big), \tag{17}$$

$$\mathcal{U}^{\pi,\widehat{P}^T_t}_{t,rect}(h_t, a_t) \triangleq r_{max}\Big(H - 1 - t - \delta_t^{\pi,\widehat{P}^T_t}(s_t, a_t|h_t)\Big). \tag{18}$$

For clarity, we present the recursive bounds for the common case $r_{\min} \le 0 \le r_{\max}$. The remaining reward-sign cases follow analogously by replacing the corresponding infimum with a supremum when the multiplying reward bound is negative; the full expressions are given in the appendix.

5.2 Optimality Guarantees

Building on the recursive fixed-policy certificates derived above for rectangular uncertainty, we now extend them to bound the **optimal robust value**. By optimizing the lower and upper recursive backups over actions, we obtain deterministic anytime bounds on $V^{*,\mathrm{rob}}_{t,\mathrm{rect}}$. Using the action-conditioned bounds (17) and (18), for an arbitrary continuation policy π, we define the lower and upper robust deterministic backups as:

$$\mathcal{LRDB}_t^\pi(h_t, a_t) \triangleq r(s_t, a_t) + \inf_{\widehat{P}^T_t \in \widehat{\mathcal{P}}^T_t} \left(\widehat{\mathbb{E}}^{\widehat{P}^T_t}_{s_{t+1}|h_t} \left[\widehat{V}^{\pi,\widehat{P}^{T,\mathcal{LRDB}}_{t+1}}_{t+1,rect}(h_{t+1}) \right] + \mathcal{L}^{\pi,\widehat{P}^T_t}_{t,rect}(h_t, a_t) \right), \tag{19}$$

$$\mathcal{URDB}_t^\pi(h_t, a_t) \triangleq r(s_t, a_t) + \inf_{\widehat{P}^T_t \in \widehat{\mathcal{P}}^T_t} \left(\widehat{\mathbb{E}}^{\widehat{P}^T_t}_{s_{t+1}|h_t} \left[\widehat{V}^{\pi,\widehat{P}^{T,\mathcal{URDB}}_{t+1}}_{t+1,rect}(h_{t+1}) \right] + \mathcal{U}^{\pi,\widehat{P}^T_t}_{t,rect}(h_t, a_t) \right). \tag{20}$$

We use $\widehat{P}^{T,\mathcal{LRDB}}_{t+1}$ and $\widehat{P}^{T,\mathcal{URDB}}_{t+1}$ to denote the projected continuation models induced by the minimizing model choices of the corresponding recursive LRDB and URDB backups from time $t+1$ onward. Importantly, the projected value and its correction term are evaluated under the same transition model inside each robust backup. Thus, the adversarial model jointly determines both the projected return and the corresponding approximation correction, rather than minimizing them independently. We now leverage these recursive deterministic backups to establish deterministic anytime optimality guarantees for the rectangular RMDP.

Theorem 2 (Optimal robust value bounds)**.** *Let h_t be an expanded history node at time step t, and let π^L and π^U be the lower- and upper-bound policies defined in* (22)*. Under rectangular transition uncertainty, the optimal robust value is bounded deterministically by:*

$$\mathcal{LRDB}_t^{\pi^L}\left(h_t, \pi_t^L(h_t)\right) \le V^{*,\mathrm{rob}}_{t,\mathrm{rect}}(h_t) \le \mathcal{URDB}_t^{\pi^U}\left(h_t, \pi_t^U(h_t)\right), \tag{21}$$

where π_t^L and π_t^U are the lower- and upper-bound policies defined greedily with respect to their corresponding deterministic backups:

$$\pi_t^L(h_t) = \arg\max_{a_t \in \mathcal{A}_t} \Big\{ \mathcal{LRDB}_t^{\pi^L}(h_t, a_t) \Big\}, \quad \pi_t^U(h_t) = \arg\max_{a_t \in \mathcal{A}_t} \Big\{ \mathcal{URDB}_t^{\pi^U}(h_t, a_t) \Big\}. \tag{22}$$

Theorem 2 assumes that all actions in $\mathcal{A}$ are initialized at every expanded decision node. The proof is provided in the Appendix A.7.

Corollary 3 (Asymptotic optimality)**.** *As the sampled state support expands to the full state space at every node, $\widehat{\mathcal{S}}(h_k^-) \uparrow \mathcal{S},\ k = t, \ldots, H$, both deterministic bounds converge to the optimal rectangular robust value:*

$$\mathcal{LRDB}_t^{\pi^L}\left(h_t, \pi_t^L(h_t)\right) \longrightarrow V_{t,\mathrm{rect}}^{*,\mathrm{rob}}(h_t), \tag{23}$$

$$\mathcal{URDB}_t^{\pi^U}\left(h_t, \pi_t^U(h_t)\right) \longrightarrow V_{t,\mathrm{rect}}^{*,\mathrm{rob}}(h_t). \tag{24}$$

The proof is provided in the Appendix A.9.

Action pruning and certification. The lower and upper action-value certificates induce a deterministic ordering test between actions. For two actions a, a', if: $\mathcal{URDB}(h, a) < \mathcal{LRDB}(h, a')$, then a is provably suboptimal and its subtree can be safely pruned. Likewise, an action a^* is certified optimal when: $\mathcal{LRDB}(h, a^*) > \max_{a \neq a^*} \mathcal{URDB}(h, a)$. This eliminates redundant search in branches that can no longer affect the optimal decision, allowing computation to be focused on unresolved regions of the tree. At the root, this provides an anytime stopping criterion: planning can terminate as soon as the optimal root action is certified, rather than exhausting the search budget.

Robust backup computation. For the TV uncertainty sets in our experiments, each inner robust optimization reduces to a robust expectation over the sampled successors and one aggregate unsampled state, solvable in $O(|\widehat{\mathcal{S}}(h,a)| \log |\widehat{\mathcal{S}}(h,a)|)$ time. See Appendix B.

Together, Theorems 1 and 2 provide runtime-accessible deterministic certificates for both policy evaluation and optimal robust planning. We next use these certificates within our MCTS planner and evaluate their effect on convergence, pruning, and root-action certification.

6 Algorithm

Algorithm 1 CAR-MCTS: Certified Anytime Robust MCTS

Input: root state s_0, horizon H, budget N
Output: root action a, certification flag c, action-value bounds

$h_0 \leftarrow$ NewNode$(s_0, 0)$
$n \leftarrow 0$
while $n < N$ **do**
  $(c, a^\star) \leftarrow$ Certify(h_0)
  if c **then**
    return $(a^\star, \text{TRUE}, [\mathcal{LRDB}(s_0, a^\star), \mathcal{URDB}(s_0, a^\star)])$
  end if
  $h \leftarrow h_0$
  path $\leftarrow \emptyset$
  while $t(h) < H - 1$ **and** $n < N$ **do**
    $a \leftarrow \arg\max_a \mathcal{URDB}_{t(h)}(h, a)$
    $s' \sim \mathring{P}^T_{t(h)}(\cdot \mid s(h), a)$
    $n \leftarrow n + 1$
    append (h, a) to *path*
    $h \leftarrow$ ExpandIfNeeded(h, a, s')
  end while
  for all (h, a) in reverse(*path*) **do**
    Backup(h, a)
    Prune(h)
  end for
end while
$a \leftarrow \arg\max_a \widehat{Q}_0^{\mathrm{rob}}(s_0, a)$
return $(a, \text{FALSE}, [\mathcal{LRDB}(s_0, a), \mathcal{URDB}(s_0, a)])$

Operations:

NewNode(s, t)
Initialize all actions $a \in \mathcal{A}$.

ExpandIfNeeded(h, a, s')
If s' is not in the sampled support, add it and create the corresponding child node. Return the child.

Backup(h, a)
Update $\widehat{Q}^{\mathrm{rob}}$, $\mathcal{LRDB}$, and $\mathcal{URDB}$.

Prune(h)
Prune action a if
$$\mathcal{URDB}(h, a) < \max_{a'} \mathcal{LRDB}(h, a').$$

Certify(h)
Let
$$a^\star = \arg\max_a \mathcal{LRDB}(h, a).$$
Return true if
$$\mathcal{LRDB}(h, a^\star) > \max_{a \neq a^\star} \mathcal{URDB}(h, a).$$

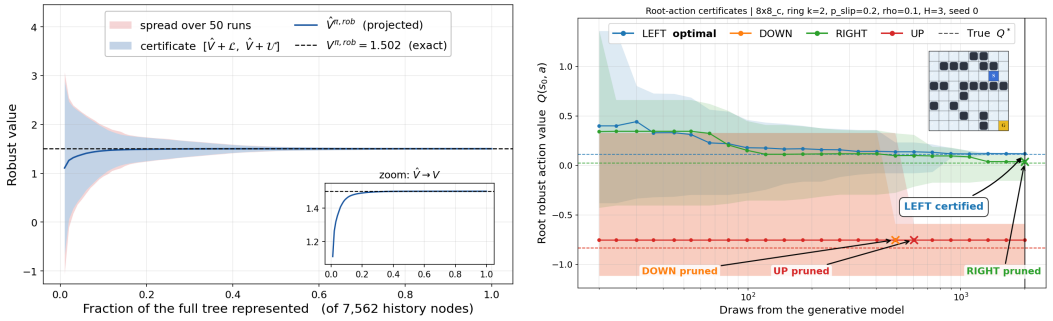
We instantiate the proposed certificates in **CAR-MCTS** (*Certified Anytime Robust MCTS*), summarized in Algorithm 1. At each simulation, the planner descends the current search tree by selecting

the action with the largest upper certificate $\mathcal{URDB}$ and samples the next state from the nominal transition model. Previously unseen successors are added to the sampled support, after which the projected robust values and certificates are backed up along the visited path, avoiding explicit enumeration of the full state space and thereby preserving scalability to large state spaces. During the backup, actions whose upper certificate falls below a competing lower certificate are pruned at every visited node. At the root, planning terminates early whenever an action is certified optimal. If certification is not reached within the sampling budget, CAR-MCTS returns the action with the largest projected robust action-value $\widehat{Q}_0^{\mathrm{rob}}$ together with deterministic optimality certificates.

7 RESULTS

Experimental setup. We evaluate CAR-MCTS in the Frozen Lake domain with varying grid sizes, terminal holes, and different start and goal cells. At each nonterminal cell, the agent chooses one of four actions: LEFT, DOWN, RIGHT, or UP. Under the nominal transition model, the agent moves toward the intended neighboring cell with probability $1 - p_{\mathrm{slip}}$; the remaining probability p_{slip} is distributed over cells in concentric Chebyshev-distance rings around the current cell, up to a configurable radius k. This ring model allows the number of possible successor states to be varied across experiments. Transition uncertainty is modeled by rectangular total-variation sets around the nominal transition rows. The reward function and remaining experiment-specific parameters, including the grid size, horizon, p_{slip}, ring radius k, and uncertainty radius, are provided in the Appendix.

We first evaluate the fixed-policy guarantees of Theorem 1 by progressively expanding the sampled tree of a fixed policy. Figure 1a plots the projected robust value and its deterministic certificate against the fraction of the complete depth-H tree represented, with the exact robust value obtained from the full tree shown for reference. Throughout the search, the exact value remains within the certificate, while the interval contracts as the sampled support expands and the projected value converges to the exact value. The experiment is repeated over 50 independent nested tree expansions, illustrating both the anytime validity and the convergence of the certificate.



(a) Fixed-policy certificate tightening.

(b) Pruning and root-action certification.

Figure 1: Deterministic certificates: fixed-policy evaluation and CAR-MCTS planning.

Figure 1b illustrates how CAR-MCTS uses the certificates during planning. Actions DOWN and UP, which lead toward nearby holes, are quickly identified as suboptimal and pruned; their certificates therefore stop changing once their subtrees are no longer sampled. The remaining computation is concentrated on the LEFT and RIGHT actions. Since tree traversal selects the action with the largest $\mathcal{URDB}$, the more promising LEFT branch is sampled more frequently, causing its certificate and projected value to tighten faster. Eventually, LEFT is deterministically certified as the optimal action. Even when certification is not reached within the budget, this targeted allocation improves the projected values used by CAR-MCTS's $\arg\max_a \widehat{Q}^{\mathrm{rob}}(s_0, a)$ fallback decision.

Comparison with robust planning baselines. Figure 2 compares CAR-MCTS with nominal UCT, RSS (Shazman et al., 2026), and Robust Power UCT (Dam et al., 2026) on a 16×16 Frozen Lake domain. The execution dynamics are fixed at $p_{\mathrm{true}} = 0.05$, while p_{slip} denotes the slip probability assumed by the planner and the uncertainty radius is adjusted to the resulting model mismatch. Hence, larger values of p_{slip} primarily increase model mismatch rather than envi-

ronment difficulty. CAR-MCTS maintains substantially higher return over increasing mismatch than all baselines, with the advantage becoming particularly pronounced at larger horizons. At $H = 11$, CAR-MCTS maintains near-maximal return through $p_{\text{slip}} = 0.5$, whereas nominal UCT and Robust Power UCT deteriorate sharply and show no comparable benefit from deeper lookahead. In particular, the degradation of nominal UCT illustrates the limitation of planning with an inaccurate nominal model when transition uncertainty is significant. RSS is omitted at $H = 11$ because its sparse-sampling expansion becomes computationally prohibitive. At $p_{\text{slip}} = 0.6$, CAR-MCTS also deteriorates as the large model mismatch induces a correspondingly large uncertainty set, leading to increasingly conservative behavior under the robust policy.

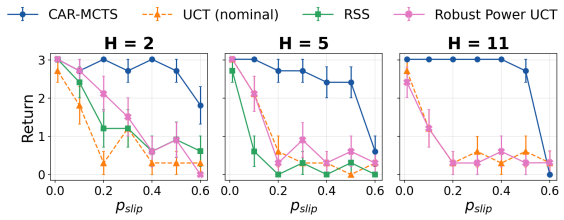


Figure 2: CAR-MCTS vs. robust planning baselines across model mismatch and planning horizons.

Scalability with state-space size. Figure 3 isolates the dependence of robust planning on state-space cardinality. For each grid size, the transition kernel is constructed so that every state has nonzero probability under every transition row, i.e., $|\operatorname{supp} P^T(\cdot \mid s, a)| = |\mathcal{S}|$, while the planning horizon and simulation budget are held fixed. CAR-MCTS remains nearly constant at 5–6 ms per decision as $|\mathcal{S}|$ grows from 10^2 to over 10^6. In contrast, Robust Power UCT grows from 2.1 ms to 11.6 s. This difference stems from the robust backups: CAR-MCTS operates only on the sampled support generated by its fixed simulation budget, whereas Robust Power UCT incurs computation over the full state space, making online planning increasingly impractical as $|\mathcal{S}|$ grows. Thus, CAR-MCTS scales essentially independently of full state-space cardinality, maintaining practical decision times even in very large state spaces. The same distinction is reflected in memory usage: CAR-MCTS requires only the sampled-support representation and remains at approximately 0.2 MB across the sweep, whereas Robust Power UCT's full-state representation grows with $|\mathcal{S}|$, reaching 152 MB at $|\mathcal{S}| \approx 10^6$.

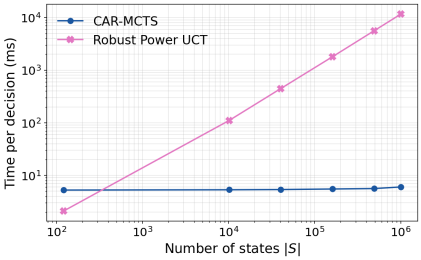


Figure 3: Decision time of CAR-MCTS and Robust Power UCT as $|\mathcal{S}|$ grows under full-support transition kernels.

Additional ablations on robustness to increasing environment slipperiness and model mismatch, sensitivity to the number of feasible successor states, scalability with planning depth, true optimal-action selection rate, and practical certification and pruning rates are provided in Appendix C.

8 Conclusion

In this work, we presented a methodology for scalable robust online MDP planning with deterministic anytime guarantees. The core principle we propose is to restrict computation to the sampled search tree while explicitly accounting for the contribution of unexplored outcomes, thereby avoiding enumeration of the full state space. We first derived deterministic lower and upper bounds on the error between the simplified and full robust values of an arbitrary policy under general transition uncertainty, including non-rectangular sets, and showed that these certificates tighten as the sampled support expands and converge as the full support is recovered. Under rectangular uncertainty, we extended this construction to recursive action-value certificates that bound the optimal robust value, enabling provably suboptimal actions to be pruned, reducing unnecessary computation, and optimal root actions to be certified for early termination. Building on these results, we introduced CAR-MCTS, which integrates the certificates directly into MCTS to guide exploration, prune resolved branches, and terminate once the root decision is certified, while retaining deterministic action-value certificates when the sampling budget is exhausted. Experiments validate the derived guarantees, demonstrate robustness to model mismatch, and show that CAR-MCTS remains practical as the full state space grows while competing robust MCTS methods become increasingly costly. To our knowledge, this is the first robust online MCTS framework combining scalable sampled-support planning with deterministic anytime guarantees.

AI USE STATEMENT

In this work, we used AI tools to assist with text polishing, related-work discovery, manuscript review for clarity and potential issues, and verification of code implementations. We take responsibility for the final content of this work.

REFERENCES

Moran Barenboim and Vadim Indelman. Online pomdp planning with anytime deterministic guarantees. *Advances in Neural Information Processing Systems*, 36:79886–79902, 2023.

Moran Barenboim and Vadim Indelman. Online pomdp planning with anytime deterministic optimality guarantees. *Artificial Intelligence*, 350:104442, 2026. ISSN 0004-3702. doi: https://doi.org/10.1016/j.artint.2025.104442. URL https://www.sciencedirect.com/science/article/pii/S0004370225001614.

Tuan Dam, Kishan Panaganti, Brahim Driss, and Adam Wierman. Online robust reinforcement learning through monte-carlo planning. *arXiv preprint arXiv:2609.18599*, 2026.

Garud N Iyengar. Robust dynamic programming. *Mathematics of Operations Research*, 30(2): 257–280, 2005.

Levente Kocsis and Csaba Szepesvári. Bandit based monte-carlo planning. In *European conference on machine learning*, pp. 282–293. Springer, 2006.

Farnaz Kohankhaki, Kiarash Aghakasiri, Hongming Zhang, Ting-Han Wei, Chao Gao, and Martin Müller. Monte carlo tree search in the presence of transition uncertainty. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 20151–20158, 2024.

Arnab Nilim and Laurent El Ghaoui. Robust control of markov decision processes with uncertain transition matrices. *Operations Research*, 53(5):780–798, 2005.

Kishan Panaganti and Dileep Kalathil. Sample complexity of robust reinforcement learning with a generative model. In *International Conference on Artificial Intelligence and Statistics*, pp. 9582–9602. PMLR, 2022.

Devavrat Shah, Qiaomin Xie, and Zhi Xu. Nonasymptotic analysis of monte carlo tree search. *Operations research*, 70(6):3234–3260, 2022.

Apoorva Sharma, James Harrison, Matthew Tsao, and Marco Pavone. Robust and adaptive planning under model uncertainty. In *Proceedings of the international conference on automated planning and scheduling*, volume 29, pp. 410–418, 2019.

Tamir Shazman, Idan Lev-Yehudi, Ron Benchetrit, and Vadim Indelman. Online robust planning under model uncertainty: A sample-based approach. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 36386–36393, 2026.

Aviv Tamar, Shie Mannor, and Huan Xu. Scaling up robust mdps using function approximation. In *International conference on machine learning*, pp. 181–189. PMLR, 2014.

Yue Wang and Shaofeng Zou. Online robust reinforcement learning with model uncertainty. *Advances in Neural Information Processing Systems*, 34:7193–7206, 2021.

Wolfram Wiesemann, Daniel Kuhn, and Berç Rustem. Robust markov decision processes. *Mathematics of Operations Research*, 38(1):153–183, 2013.

A MATHEMATICAL ANALYSIS AND PROOFS

A.1 PROOF OF THEOREM 1 (FIXED-POLICY DETERMINISTIC BOUNDS):

Essentials lemmas and definitions for the proof:

1. The restriction mapping $\mathcal{R}$:

$$\mathcal{R}^T_{t:k-1}(P^T_{t:k-1}; h_{k-1}) = \begin{cases} P^T_{t:k-1}, & s_{t:k-1} \in \widehat{\mathcal{S}}(h_{k-1}), \\ 0, & \text{otherwise}. \end{cases} \tag{25}$$

Lemma 1 (restated). *Let H be the horizon of the MDP, $\pi = \{\pi_k\}_{k=t}^{H-1}$ be the policy,$r : \mathcal{S} \times \mathcal{A} \to [r_{\min}, r_{\max}]$ be the bounded state-action reward function , and $\widehat{P}^T_k$ be the projected transition model defined in* (6)*. The theoretical and simplified value functions V^π_t and $\widehat{V}^\pi_t$ at time step t are:*

$$V^\pi_t(h_t) = \sum_{j=t}^{H-1} \sum_{s_{t+1:j}} \prod_{k=t}^{j-1} P^T_k\big(s_{k+1} \mid s_k, \pi_k(h_k)\big) r\big(s_j, \pi_j(h_j)\big), \tag{26}$$

$$\widehat{V}^\pi_t(h_t) = \sum_{j=t}^{H-1} \sum_{s_{t+1:j} \in \widehat{\mathcal{S}}(h_j)} \prod_{k=t}^{j-1} \widehat{P}^T_k\big(s_{k+1} \mid s_k, \pi_k(h_k)\big) r\big(s_j, \pi_j(h_j)\big). \tag{27}$$

Lemma 2 (restated). *For any 2 functions $f(\theta)$ and $\widehat{f}(\theta)$, the following holds:*

$$\inf_\theta f(\theta) - \inf_\theta \hat{f}(\theta) \le \sup_\theta \Big(f(\theta) - \hat{f}(\theta) \Big). \tag{28}$$

Lemma 3 (restated). *Let $\mathcal{P}$ denote some uncertainty set, and $\widehat{\mathcal{P}}$ denote its projected counterpart defined in* (8)*. Let $\mathcal{R}$ be the restriction map that takes a full probability distribution $P \in \mathcal{P}$ and retains only the entries associated with the sampled support:*

$$\mathcal{R}(P; \widehat{\mathcal{S}}) = \begin{cases} P(s), & s \in \widehat{\mathcal{S}}, \\ 0, & \textit{otherwise}. \end{cases} \tag{29}$$

Then minimizing some function $f : \widehat{P} \to \mathbb{R}$ over $\widehat{\mathcal{P}}$ is the same as minimizing $f(\mathcal{R}(P))$ over the full set $\mathcal{P}$, namely:

$$\inf_{P \in \mathcal{P}} f(\mathcal{R}(P)) = \inf_{\widehat{P} \in \widehat{\mathcal{P}}} f(\widehat{P}). \tag{30}$$

Lemma 4 (restated). *For any 2 functions $f(\theta)$ and $\widehat{f}(\theta)$, the following holds:*

$$\inf_\theta \Big(f(\theta) - \hat{f}(\theta) \Big) . \le \inf_\theta f(\theta) - \inf_\theta \hat{f}(\theta) \tag{31}$$

Proof of Theorem 1.

Theorem 1 (restated) (Fixed-policy deterministic bounds)**.** *Let h_t be the history at time t, and $V^{\pi,\mathrm{rob}}_t(h_t)$ and $\widehat{V}^{\pi,\mathrm{rob}}_t(h_t)$ denote the robust values of the same policy π in the full* (2) *and simplified* (9) *RMDPs, respectively. For any transition uncertainty set $\mathcal{P}^T_{t:H-1}$ and its projected counterpart $\widehat{\mathcal{P}}^T_{t:H-1}$:*

$$\mathcal{L}^\pi_t(h_t) \le V^{\pi,\mathrm{rob}}_t(h_t) - \widehat{V}^{\pi,\mathrm{rob}}_t(h_t) \ \le \ \mathcal{U}^\pi_t(h_t), \tag{32}$$

such that:

$$\mathcal{U}_t^\pi(h_t) \triangleq \begin{cases} r_{max}\left(H-t-\inf\limits_{\widehat{P}^T_{t:H-1}\in\widehat{\mathcal{P}}^T_{t:H-1}} m_t^\pi\left(\widehat{P}^T_{t:H-1}\right)\right), r_{max}\geq 0, \\ r_{max}\left(H-t-\sup\limits_{\widehat{P}^T_{t:H-1}\in\widehat{\mathcal{P}}^T_{t:H-1}} m_t^\pi\left(\widehat{P}^T_{t:H-1}\right)\right), r_{max}< 0, \end{cases} \tag{33}$$

$$\mathcal{L}_t^\pi(h_t) \triangleq \begin{cases} r_{min}\left(H-t-\sup\limits_{\widehat{P}^T_{t:H-1}\in\widehat{\mathcal{P}}^T_{t:H-1}} m_t^\pi\left(\widehat{P}^T_{t:H-1}\right)\right), r_{min}\geq 0, \\ r_{min}\left(H-t-\inf\limits_{\widehat{P}^T_{t:H-1}\in\widehat{\mathcal{P}}^T_{t:H-1}} m_t^\pi\left(\widehat{P}^T_{t:H-1}\right)\right), r_{min}< 0, \end{cases} \tag{34}$$

and

$$m_t^\pi\left(\widehat{P}^T_{t:H-1}\right) \triangleq \sum_{j=t}^{H-1}\sum_{s_{t+1:j}\in\widehat{\mathcal{S}}(h_j)}\prod_{k=t}^{j-1}\widehat{P}_k^T\big(s_{k+1}\mid s_k,\pi_k(h_k)\big). \tag{35}$$

Proof.
The robust non-rectangular value function is defined as:

$$V_t^{\pi,\mathrm{rob}}(h_t) \triangleq \inf_{P^T_{t:H-1}\in\mathcal{P}^T_{t:H-1}} V_t^\pi\left(h_t; P^T_{t:H-1}\right), \tag{36}$$

and the simplified robust value function:

$$\widehat{V}_t^{\pi,\mathrm{rob}}(h_t) \triangleq \inf_{\widehat{P}^T_{t:H-1}\in\widehat{\mathcal{P}}^T_{t:H-1}} \widehat{V}_t^\pi\left(h_t; \widehat{P}^T_{t:H-1}\right), \tag{37}$$

where the theoretical and simplified value functions are defined in (26) and (27).

Using Lemma (3) together with the restriction map $\mathcal{R}^T_{t:H-1}$ from (25), which project the full uncertainty set onto the projected uncertainty set, we obtain:

$$\widehat{V}_t^{\pi,\mathrm{rob}}(h_t) = \inf_{P^T_{t:H-1}\in\mathcal{P}^T_{t:H-1}} \widehat{V}_t^\pi\big(h_t; \mathcal{R}^T_{t:H-1}(P^T_{t:H-1})\big). \tag{38}$$

We first begin with the **Upper Bound**:

$$\begin{aligned} &V_t^{\pi,\mathrm{rob}}(h_t) - \widehat{V}_t^{\pi,\mathrm{rob}}(h_t) \\ =& \inf_{P^T_{t:H-1}\in\mathcal{P}^T_{t:H-1}} V_t^\pi\big(h_t;\, P^T_{t:H-1}\big) - \inf_{P^T_{t:H-1}\in\mathcal{P}^T_{t:H-1}} \widehat{V}_t^\pi\big(h_t;\, \mathcal{R}^T_{t:H-1}(P^T_{t:H-1})\big) &(39)\\ \leq& \sup_{P^T_{t:H-1}\in\mathcal{P}^T_{t:H-1}} \Big(V_t^\pi\big(h_t;\, P^T_{t:H-1}\big) - \widehat{V}_t^\pi\big(h_t;\, \mathcal{R}^T_{t:H-1}(P^T_{t:H-1})\big)\Big) &(40)\end{aligned}$$

For the last transition we used Lemma 2.

Writing the value functions explicitly:

$$\begin{aligned} \sup_{P^T_{t:H-1}\in\mathcal{P}^T_{t:H-1}} \Bigg(&\sum_{j=t}^{H-1}\sum_{s_{t+1:j}}\prod_{k=t}^{j-1} P_k^T\big(s_{k+1}\mid s_k,\pi_k(h_k)\big) r\big(s_j,\pi_j(h_j)\big) - \\ &\sum_{j=t}^{H-1}\sum_{s_{t+1:j}\in\widehat{\mathcal{S}}(h_j)}\prod_{k=t}^{j-1} \mathcal{R}_k^T\left(P_k^T\big(s_{k+1}\mid s_k,\pi_k(h_k)\big)\right) r\big(s_j,\pi_j(h_j)\big)\Bigg) \end{aligned} \tag{41}$$

splitting the sums into in/out support:

$$\sup_{P^T_{t:H-1}\in\mathcal{P}^T_{t:H-1}}\left(\sum_{j=t}^{H-1}\Big(\sum_{s_{t+1:j}\in\widehat{\mathcal{S}}(h_j)}\prod_{k=t}^{j-1}\underbrace{\left(P_k^T\big(s_{k+1}\mid s_k,\pi_k(h_k)\big)-\mathcal{R}_k^T\left(P_k^T\big(s_{k+1}\mid s_k,\pi_k(h_k)\big)\right)\right)}_{\equiv 0} r\big(s_j,\pi_j(h_j)\big)\right.$$
$$\left.+\sum_{s_{t+1:j}\notin\widehat{\mathcal{S}}(h_j)}\prod_{k=t}^{j-1}P_k^T\big(s_{k+1}\mid s_k,\pi_k(h_k)\big)r\big(s_j,\pi_j(h_j)\big)\Big)\right)= \tag{42}$$

The under braced term equals zero by the definition of the Restricted mapping $\mathcal{R}$. Therefore:

$$\sup_{P^T_{t:H-1}\in\mathcal{P}^T_{t:H-1}}\left(\sum_{j=t}^{H-1}\sum_{s_{t+1:j}\notin\widehat{\mathcal{S}}(h_j)}\prod_{k=t}^{j-1}P_k^T\big(s_{k+1}\mid s_k,\pi_k(h_k)\big)r\big(s_j,\pi_j(h_j)\big)\right) \tag{43}$$

$$\leq r_{max}\sup_{P^T_{t:H-1}\in\mathcal{P}^T_{t:H-1}}\left(\sum_{j=t}^{H-1}\sum_{s_{t+1:j}\notin\widehat{\mathcal{S}}(h_j)}\prod_{k=t}^{j-1}P_k^T\big(s_{k+1}\mid s_k,\pi_k(h_k)\big)\right) \tag{44}$$

Since $r(s,a)\leq r_{\max}$, the out-of-support contribution is upper bounded by $r_{\max}$ times its probability mass. For $r_{\max}\geq 0$, maximizing this product is equivalent to maximizing the out-of-support mass, yielding the supremum in (44). If $r_{\max}<0$, the supremum is replaced by an infimum.

Since $P(s_{t+1:j}|h_t)$ defines a full probability distribution over sequences of states trajectories, we have:

$$\sum_{s_{t+1:j}}P(s_{t+1:j}|h_t)=\sum_{s_{t+1:j}}\prod_{k=t}^{j-1}P_k^T\big(s_{k+1}\mid s_k,\pi_k(h_k)\big)\equiv 1, \tag{45}$$

therefore:

$$\sum_{s_{t+1:j}\notin\widehat{\mathcal{S}}(h_j)}P(s_{t+1:j}|h_t)=1-\sum_{s_{t+1:j}\in\widehat{\mathcal{S}}(h_j)}P(s_{t+1:j}|h_t). \tag{46}$$

Thus (44) equals:

$$r_{max}\sup_{P^T_{t:H-1}\in\mathcal{P}^T_{t:H-1}}\left(\sum_{j=t}^{H-1}\left(1-\sum_{s_{t+1:j}\in\widehat{\mathcal{S}}(h_j)}\prod_{k=t}^{j-1}P_k^T\big(s_{k+1}\mid s_k,\pi_k(h_k)\big)\right)\right)= \tag{47}$$

$$r_{max}\sup_{P^T_{t:H-1}\in\mathcal{P}^T_{t:H-1}}\left(\sum_{j=t}^{H-1}\left(1-\sum_{s_{t+1:j}\in\widehat{\mathcal{S}}(h_j)}\prod_{k=t}^{j-1}\mathcal{R}_k^T\left(P_k^T\big(s_{k+1}\mid s_k,\pi_k(h_k)\big)\right)\right)\right)= \tag{48}$$

Using Lemma 3 again:

$$r_{max}\sup_{\widehat{P}^T_{t:H-1}\in\widehat{\mathcal{P}}^T_{t:H-1}}\left(\sum_{j=t}^{H-1}\left(1-\sum_{s_{t+1:j}\in\widehat{\mathcal{S}}(h_j)}\prod_{k=t}^{j-1}\widehat{P}_k^T\big(s_{k+1}\mid s_k,\pi_k(h_k)\big)\right)\right)= \tag{49}$$

$$r_{max}\left(H-t-\inf_{\widehat{P}^T_{t:H-1}\in\widehat{\mathcal{P}}^T_{t:H-1}}\sum_{j=t}^{H-1}\sum_{s_{t+1:j}\in\widehat{\mathcal{S}}(h_j)}\prod_{k=t}^{j-1}\widehat{P}_k^T\big(s_{k+1}\mid s_k,\pi_k(h_k)\big)\right)\triangleq \tag{50}$$

$$\mathcal{U}_t^\pi(h_t) \tag{51}$$

Lower Bound:

$$V_t^{\pi,\mathrm{rob}}(h_t) - \widehat{V}_t^{\pi,\mathrm{rob}}(h_t) =$$

$$\inf_{P_{t:H-1}^T \in \mathcal{P}_{t:H-1}^T} V_t^\pi\big(h_t;\, P_{t:H-1}^T\big) - \inf_{P_{t:H-1}^T \in \mathcal{P}_{t:H-1}^T} \widehat{V}_t^\pi\big(h_t;\, \mathcal{R}_{t:H-1}^T(P_{t:H-1}^T)\big) \geq \tag{52}$$

$$\inf_{P_{t:H-1}^T \in \mathcal{P}_{t:H-1}^T} \Big(V_t^\pi\big(h_t;\, P_{t:H-1}^T\big) - \widehat{V}_t^\pi\big(h_t;\, \mathcal{R}_{t:H-1}^T(P_{t:H-1}^T)\big)\Big) = \tag{53}$$

For the last transition we used Lemma 4. As shown above in (42), the difference on the support is canceled, and we get:

$$\inf_{P_{t:H-1}^T \in \mathcal{P}_{t:H-1}^T} \sum_{j=t}^{H-1} \left(\sum_{s_{t+1:j} \notin \widehat{\mathcal{S}}(h_j)} \prod_{k=t}^{j-1} P_k^T\big(s_{k+1} \mid s_k, \pi_k(h_k)\big) r\big(s_j, \pi_j(h_j)\big) \right) \geq \tag{54}$$

Since $r(s,a) \geq r_{\min}$, the out-of-support contribution is lower bounded by $r_{\min}$ times its probability mass. For $r_{\min} \leq 0$, minimizing this product is equivalent to maximizing the out-of-support mass, yielding the supremum below. If $r_{\min} > 0$, the supremum is replaced by an infimum.

$$r_{min} \sup_{P_{t:H-1}^T \in \mathcal{P}_{t:H-1}^T} \sum_{j=t}^{H-1} \left(\sum_{s_{t+1:j} \notin \widehat{\mathcal{S}}(h_j)} \prod_{k=t}^{j-1} P_k^T\big(s_{k+1} \mid s_k, \pi_k(h_k)\big) \right) = \tag{55}$$

$$r_{min} \sup_{\widehat{P}_{t:H-1}^T \in \widehat{\mathcal{P}}_{t:H-1}^T} \sum_{j=t}^{H-1} \left(1 - \sum_{s_{t+1:j} \in \widehat{\mathcal{S}}(h_j)} \prod_{k=t}^{j-1} \widehat{P}_k^T\big(s_{k+1} \mid s_k, \pi_k(h_k)\big) \right) = \tag{56}$$

$$r_{min} \left(H - t - \inf_{\widehat{P}_{t:H-1}^T \in \widehat{\mathcal{P}}_{t:H-1}^T} \sum_{j=t}^{H-1} \sum_{s_{t+1:j} \in \widehat{\mathcal{S}}(h_j)} \prod_{k=t}^{j-1} \widehat{P}_k^T\big(s_{k+1} \mid s_k, \pi_k(h_k)\big) \right) \tag{57}$$

$$\triangleq \mathcal{L}_t^\pi(h_t) \tag{58}$$

which concludes the proof. □

Transition to a rectangular setting:
If we assume rectangular uncertainty sets 4, the above bounds 51, 58 can be written as nested optimizations, that allows dynamic programming:

$$\begin{aligned} \mathcal{U}_{t,rect}^{\pi^U}(h_t) \triangleq r_{max} \Bigg(H - 1 - t - \Bigg(& \inf_{\widehat{P}_t^T \in \widehat{\mathcal{P}}_t^T} \sum_{s_{t+1} \in \widehat{\mathcal{S}}(h_{t+1}^-)} \widehat{P}_t^T(s_{t+1} \mid \pi_t(h_t), s_t; h_t) \Bigg(\\ & 1 + \inf_{\widehat{P}_{t+1}^T \in \widehat{\mathcal{P}}_{t+1}^T} \sum_{s_{t+2} \in \widehat{\mathcal{S}}(h_{t+2}^-)} \widehat{P}_{t+1}^T(s_{t+2} \mid \pi_{t+1}, s_{t+1}) \Bigg(\\ & \cdots \\ & 1 + \inf_{\widehat{P}_{H-2}^T \in \widehat{\mathcal{P}}_{H-2}^T} \sum_{s_{H-1} \in \widehat{\mathcal{S}}(h_{H-1}^-)} \widehat{P}_{H-2}^T(s_{H-1} \mid \pi_{H-2}, s_{H-2}) \Bigg) \cdots \Bigg) \end{aligned} \tag{59}$$

$$\mathcal{L}^{\pi^L}_{t,rect}(h_t) \triangleq r_{min}\Bigg(H-1-t-\Bigg(\inf_{\widehat{P}^T_t\in\widehat{\mathcal{P}}^T_t}\sum_{s_{t+1}\in\widehat{\mathcal{S}}(h^-_{t+1})}\widehat{P}^T_t(s_{t+1}\mid\pi_t(h_t),s_t;h_t)\Bigg($$

$$1+\inf_{\widehat{P}^T_{t+1}\in\widehat{\mathcal{P}}^T_{t+1}}\sum_{s_{t+2}\in\widehat{\mathcal{S}}(h^-_{t+2})}\widehat{P}^T_{t+1}(s_{t+2}\mid\pi_{t+1},s_{t+1})\Bigg($$

$$\cdots$$

$$1+\inf_{\widehat{P}^T_{H-2}\in\widehat{\mathcal{P}}^T_{H-2}}\sum_{s_{H-1}\in\widehat{\mathcal{S}}(h^-_{H-1})}\widehat{P}^T_{H-2}(s_{H-1}\mid\pi_{H-2},s_{H-2})\Bigg)\cdots\Bigg)\Bigg) \tag{60}$$

We can write 59, 60 in a compact form, recursively:

$$\mathcal{U}^{\pi^U}_{t,rect}(h_t) \triangleq r_{max}\Big(H-1-t-\delta^{\pi^U}_t(s_t|h_t)\Big) \tag{61}$$

$$\mathcal{L}^{\pi^L}_{t,rect}(h_t) \triangleq r_{min}\Big(H-1-t-\delta^{\pi^L}_t(s_t|h_t)\Big) \tag{62}$$

Such that for $k\in[t,H-2]$:

$$\delta^{\pi}_k(s_k|h_k)=\inf_{\widehat{P}^T_k\in\widehat{\mathcal{P}}^T_k}\delta^{\pi,\widehat{P}^T_k}_k(s_k,\pi_k(h_k)) \tag{63}$$

$$\delta^{\pi,\widehat{P}^T_k}_k(s_k,a_k|h_k)=\sum_{s_{k+1}\in\widehat{\mathcal{S}}(h^-_{k+1})}\widehat{P}^T_k(s_{k+1}\mid s_k,a_k)\ \left[1+\delta^{\pi}_{k+1}(s_{k+1}|h_{k+1})\right] \tag{64}$$

and for $k=H-1$ there is no continuation therefore:

$$\delta^{\pi}_{H-1}(s_{H-1})\equiv 0 \tag{65}$$

Note: we switch the inf to sup:

1. In equations 60, 63: if $r_{\min}>0$.
2. In equations 59, 63: if $r_{\max}<0$.
3. δ is shared only when $r_{min}\le 0\le r_{max}$

A.2 Proof of Lemma 1

Lemma 1. *Let H be the horizon of the MDP, $\pi=\{\pi_k\}_{k=t}^{H-1}$ be the policy,$r:\mathcal{S}\times\mathcal{A}\to[r_{\min},r_{\max}]$ be the bounded state-action reward function , and $\widehat{P}^T_k$ be the projected transition model defined in* (6)*. The theoretical and simplified value functions V^π_t and $\widehat{V}^\pi_t$ at time step t are:*

$$V^\pi_t(h_t)=\sum_{j=t}^{H-1}\sum_{s_{t+1:j}}\prod_{k=t}^{j-1}P^T_k\big(s_{k+1}\mid s_k,\pi_k(h_k)\big)r\big(s_j,\pi_j(h_j)\big), \tag{66}$$

$$\widehat{V}^\pi_t(h_t)=\sum_{j=t}^{H-1}\sum_{s_{t+1:j}\in\widehat{\mathcal{S}}(h_j)}\prod_{k=t}^{j-1}\widehat{P}^T_k\big(s_{k+1}\mid s_k,\pi_k(h_k)\big)r\big(s_j,\pi_j(h_j)\big). \tag{67}$$

Proof. The value function is the expected future cumulative reward:

$$V_t^{\pi}(h_t) = \mathbb{E}^{\pi,P^T}_{s_{t+1:H-1}} \left[\sum_{j=t}^{H-1} r(s_j, \pi_j) \mid h_t \right]. \tag{68}$$

By the linearity of the expectation $\mathbb{E}[\sum f(\cdot)] = \sum \mathbb{E}[f(\cdot)]$, and writing the expectation explicitly:

$$\sum_{j=t}^{H-1} \mathbb{E}^{\pi,P^T}_{s_{t+1:H-1}} [r(s_j, \pi_j) \mid h_t] = \sum_{j=t}^{H-1} \sum_{s_{t+1:j}} P\big(s_{t+1:j}|h_t\big) r(s_j, \pi_j). \tag{69}$$

Applying chain rule on the states sequence probability:

$$P(s_{t+1:j}|h_t) = \prod_{k=t}^{j-1} P(s_{k+1}|h_t, s_{t+1:k}). \tag{70}$$

We assume markovian models, therefore the probability of transition to next state depends only on current state and action. since $h_k = (s_{t:k}, a_{t:k-1})$, the states sequence probability is:

$$P(s_{t+1:j}|h_t) = \prod_{k=t}^{j-1} P_k^T\big(s_{k+1}|s_k, \pi(h_k)\big). \tag{71}$$

Where in (71) we assume non-stationary transition model. Using (69) and (71), we derived the final expression for the value function:

$$V_t^{\pi}(h_t) = \sum_{j=t}^{H-1} \sum_{s_{t+1:j}} \prod_{k=t}^{j-1} P_k^T\big(s_{k+1}|s_k, \pi(h_k)\big) r\big(s_j, \pi_j(h_j)\big). \tag{72}$$

Similarly, using the projected transition model defined in (6), the simplified value function is:

$$\widehat{V}_t^{\pi}(h_t) = \sum_{j=t}^{H-1} \sum_{s_{t+1:j} \in \widehat{\mathcal{S}}(h_j)} \prod_{k=t}^{j-1} \widehat{P}_k^T\big(s_{k+1} \mid s_k, \pi_k(h_k)\big) r\big(s_j, \pi_j(h_j)\big). \tag{73}$$

□

For **stochastic policies**:

$$V_t^{\pi}(h_t) = \sum_{j=t}^{H-1} \sum_{a_{t:j}} \sum_{s_{t+1:j}} \left(\prod_{k=t}^{j-1} \pi(a_k|h_k) P_k^T(s_{k+1}|s_k, a_k) \right) \pi(a_j|h_j) r(s_j, a_j). \tag{74}$$

A.3 Proof of Lemma 2

Lemma 2. *for any 2 function $f(\theta)$, $\widehat{f}(\theta)$, the following holds:*

$$\inf_{\theta} f(\theta) - \inf_{\theta} \hat{f}(\theta) \leq \sup_{\theta} \Big(f(\theta) - \hat{f}(\theta) \Big) \tag{75}$$

Proof.
Let

$$\varepsilon \triangleq \sup_{\theta \in \Theta} \Big(f(\theta) - g(\theta) \Big)$$

Then for every θ,

$$f(\theta) - g(\theta) \leq \varepsilon \quad \rightarrow \quad f(\theta) \leq \varepsilon + g(\theta)$$

Taking minima gives

$$\inf_{\theta} f(\theta) \leq \inf_{\theta} g(\theta) + \varepsilon$$

Therefore

$$\inf_{\theta} f(\theta) - \inf_{\theta} g(\theta) \leq \varepsilon \triangleq \sup_{\theta \in \Theta} \Big(f(\theta) - g(\theta) \Big)$$

□

A.4 Proof of Lemma 3

Lemma 3. *Let $\mathcal{P}$ denote some uncertainty set, and $\widehat{\mathcal{P}}$ denote its projected version. Let $\mathcal{R}$ be the restriction map that takes a full probability distribution $P \in \mathcal{P}$ and retains only the entries associated with the sampled support:*

$$\mathcal{R}(P; \widehat{\mathcal{S}}) = \begin{cases} P(s), & s \in \widehat{\mathcal{S}}, \\ 0, & \textit{otherwise}. \end{cases} \tag{76}$$

Then minimizing some function $f : \widehat{P} \to \mathbb{R}$ over $\widehat{\mathcal{P}}$ is the same as minimizing $f(\mathcal{R}(P))$ over the full set $\mathcal{P}$, namely:

$$\inf_{P \in \mathcal{P}} f(\mathcal{R}(P)) = \inf_{\widehat{P} \in \widehat{\mathcal{P}}} f(\widehat{P}), \tag{77}$$

Proof.
By the definition of the projected uncertainty sets as the images of the full uncertainty sets under the restriction maps, we have

$$\widehat{\mathcal{P}} = \mathcal{R}\left(\mathcal{P}\right), \tag{78}$$

We prove the desired equality by showing both inequalities. First, let

$$P \in \mathcal{P}, \tag{79}$$

be arbitrary full distribution. By definition:

$$\mathcal{R}(P) \in \widehat{\mathcal{P}}, \tag{80}$$

Therefore,

$$\inf_{\widehat{P} \in \widehat{\mathcal{P}}} f(\widehat{P}) \leq f(\mathcal{R}(P)). \tag{81}$$

Since this holds for every

$$P \in \mathcal{P}, \tag{82}$$

we can take the infimum over the full uncertainty set and obtain

$$\inf_{\widehat{P} \in \widehat{\mathcal{P}}} f(\widehat{P}) \leq \inf_{P \in \mathcal{P}} f(\mathcal{R}(P)). \tag{83}$$

For the opposite direction, let

$$\widehat{P} \in \widehat{\mathcal{P}}, \tag{84}$$

be arbitrary projected distribution. Since the projected uncertainty set is defined as the image of the full uncertainty set, there exist a full model:

$$P \in \mathcal{P}, \tag{85}$$

such that

$$\widehat{P} = \mathcal{R}(P), \tag{86}$$

Therefore,

$$f(\widehat{P}) = f(\mathcal{R}(P)) \tag{87}$$

Hence,

$$\inf_{P \in \mathcal{P}} f(\mathcal{R}(P)) \leq f(\widehat{P}). \tag{88}$$

Since this holds for every

$$\widehat{P} \in \widehat{\mathcal{P}}, \tag{89}$$

we can take the infimum over the projected uncertainty set and obtain

$$\inf_{P \in \mathcal{P}} f(\mathcal{R}(P)) \leq \inf_{\widehat{P} \in \widehat{\mathcal{P}}} f(\widehat{P}). \tag{90}$$

Combining the two inequalities gives equality:

$$\inf_{P \in \mathcal{P}} f(\mathcal{R}(P)) = \inf_{\widehat{P} \in \widehat{\mathcal{P}}} f(\widehat{P}). \tag{91}$$

Which proves the Lemma.

□

A.5 Proof of Lemma 4

Lemma 4. *for any 2 function $f(\theta), \widehat{f}(\theta)$, the following holds:*

$$\inf_{\theta} \Big(f(\theta) - \hat{f}(\theta) \Big) \leq \inf_{\theta} f(\theta) - \inf_{\theta} \hat{f}(\theta) \tag{92}$$

Proof.
Let

$$\varepsilon \triangleq \inf_{\theta \in \Theta} \Big(f(\theta) - g(\theta) \Big)$$

Then for every θ,

$$\varepsilon \leq f(\theta) - g(\theta) \quad \rightarrow \quad \varepsilon + g(\theta) \leq f(\theta)$$

Taking minima gives

$$\varepsilon + \inf_{\theta} g(\theta) \leq \inf_{\theta} f(\theta)$$

Therefore

$$\varepsilon \triangleq \inf_{\theta \in \Theta} \Big(f(\theta) - g(\theta) \Big) \leq \inf_{\theta} f(\theta) - \inf_{\theta} g(\theta)$$

□

A.6 Proof of Corollary 1 (Asymptotic convergence)

Corollary 1 (restated) (Asymptotic convergence)**.** *For any policy π, as the sampled state support expands to the full state space at every node $\widehat{\mathcal{S}}(h_k^-) \uparrow \mathcal{S},\ k = t, \ldots, H$, the simplified robust value* (9) *convergences to the true robust value* (2)*:*

$$\widehat{V}_t^{\pi,\mathrm{rob}}(h_t) \xrightarrow{\widehat{\mathcal{S}} \to \mathcal{S}} V_t^{\pi,\mathrm{rob}}(h_t). \tag{93}$$

Proof. Restating the upper and lower bounds of theorem 1:

$$\mathcal{U}_t^\pi(h_t) \triangleq \begin{cases} r_{max}\left(H - t - \inf\limits_{\widehat{P}_{t:H-1}^T \in \widehat{\mathcal{P}}_{t:H-1}^T} m_t^\pi\left(\widehat{P}_{t:H-1}^T\right)\right), r_{max} \geq 0, \\ r_{max}\left(H - t - \sup\limits_{\widehat{P}_{t:H-1}^T \in \widehat{\mathcal{P}}_{t:H-1}^T} m_t^\pi\left(\widehat{P}_{t:H-1}^T\right)\right), r_{max} < 0 \end{cases} \tag{94}$$

$$\mathcal{L}_t^\pi(h_t) \triangleq \begin{cases} r_{min}\left(H - t - \sup\limits_{\widehat{P}_{t:H-1}^T \in \widehat{\mathcal{P}}_{t:H-1}^T} m_t^\pi\left(\widehat{P}_{t:H-1}^T\right)\right), r_{min} \geq 0, \\ r_{min}\left(H - t - \inf\limits_{\widehat{P}_{t:H-1}^T \in \widehat{\mathcal{P}}_{t:H-1}^T} m_t^\pi\left(\widehat{P}_{t:H-1}^T\right)\right), r_{min} < 0 \end{cases} \tag{95}$$

$$m_t^\pi\left(\widehat{P}_{t:H-1}^T\right) \triangleq \sum_{j=t}^{H-1} \sum_{s_{t+1:j} \in \widehat{\mathcal{S}}(h_j)} \prod_{k=t}^{j-1} \widehat{P}_k^T\left(s_{k+1} \mid s_k, \pi_k(h_k)\right) \tag{96}$$

The state-trajectory likelihood is bounded by 1, and is monotonic non-decreasing for every new state sample that is added, and equals 1 over the full state space, namely:

$$\sum_{s_{t+1:j} \in \widehat{\mathcal{S}}(h_j)} P(s_{t+1:j}|h_t) = \sum_{s_{t+1:j} \in \widehat{\mathcal{S}}(h_j)} \prod_{k=t}^{j-1} P_k^T\left(s_{k+1} \mid s_k, \pi_k(h_k)\right) \xrightarrow{\widehat{\mathcal{S}} \to \mathcal{S}} 1, \tag{97}$$

Therefore

$$m_t^\pi\left(\widehat{P}_{t:H-1}^T\right) \xrightarrow{\widehat{\mathcal{S}} \to \mathcal{S}} H - t \tag{98}$$

Which induces:

$$\mathcal{U}_t^\pi(h_t) \xrightarrow{\widehat{\mathcal{S}} \to \mathcal{S}} 0 \tag{99}$$

$$\mathcal{L}_t^\pi(h_t) \xrightarrow{\widehat{\mathcal{S}} \to \mathcal{S}} 0 \tag{100}$$

□

A.7 Proof of Theorem 2 (Optimal robust value bounds):

Essentials lemmas and definitions for the proof:

1. In-support expectation operator:

$$\widehat{\mathbb{E}}_s[f(h)] = \sum_{s \in \widehat{\mathcal{S}}} P_t^T(s) \cdot f(h) \tag{101}$$

2. Upper robust deterministic bound:

$$\mathcal{URDB}_t^{\pi^U}(h_t, a_t) \triangleq r(s_t, a_t) + \inf_{\widehat{P}_t^T \in \widehat{\mathcal{P}}_t^T} \left(\widehat{\mathbb{E}}^{\widehat{P}_t^T}_{s_{t+1}|h_t} \left[\widehat{V}^{\pi^U, \widehat{P}_{t+1}^T, \mathcal{URDB}}_{t+1,rect}(h_{t+1}) \right] + \mathcal{U}^{\pi^U, \widehat{P}_t^T}_{t,rect}(h_t, a_t) \right) \tag{102}$$

$$= \widehat{Q}^{\pi^U, \widehat{P}_t^T, \mathcal{URDB}}_{t,rect}(h_t, a_t) + \mathcal{U}^{\pi^U, \widehat{P}_t^T, \mathcal{URDB}}_{t,rect}(h_t, a_t) \tag{103}$$

3. Lower robust deterministic bound:

$$\mathcal{LRDB}_t^{\pi^L}(h_t, a_t) \triangleq r(s_t, a_t) + \inf_{\widehat{P}_t^T \in \widehat{\mathcal{P}}_t^T} \left(\widehat{\mathbb{E}}^{\widehat{P}_t^T}_{s_{t+1}|h_t} \left[\widehat{V}^{\pi^L, \widehat{P}_{t+1}^T, \mathcal{LRDB}}_{t+1,rect}(h_{t+1}) \right] + \mathcal{L}^{\pi^L, \widehat{P}_t^T}_{t,rect}(h_t, a_t) \right) \tag{104}$$

$$= \widehat{Q}^{\pi^L, \widehat{P}_t^T, \mathcal{LRDB}}_{t,rect}(h_t, a_t) + \mathcal{L}^{\pi^L, \widehat{P}_t^T, \mathcal{LRDB}}_{t,rect}(h_t, a_t) \tag{105}$$

Lemma 5 (restated). *Let h_t be the history at time step t. For any action $a_t \in \mathcal{A}_t$, and future action a_{t+1} selected by the greedy policies π^L, π^U, the following inequalities hold:*

$$\mathcal{LRDB}_t^{\pi^L}(h_t, a_t) \leq r(s_t, a_t) + \inf_{P_t^T \in \mathcal{P}_t^T} \left(\widehat{\mathbb{E}}^{P_t^T}_{s_{t+1}|h_t} \left[\mathcal{LRDB}_{t+1}^{\pi^L}(h_{t+1}) \right] + P^{out} \cdot r_{min} \cdot (H - 1 - t) \right), \tag{106}$$

And:

$$\mathcal{URDB}_t^{\pi^U}(h_t, a_t) \geq r(s_t, a_t) + \inf_{P_t^T \in \mathcal{P}_t^T} \left(\widehat{\mathbb{E}}^{P_t^T}_{s_{t+1}|h_t} \left[\mathcal{URDB}_{t+1}^{\pi^U}(h_{t+1}) \right] + P^{out} \cdot r_{max} \cdot (H - 1 - t) \right). \tag{107}$$

Proof of Theorem 2.

Theorem 2 (restated) (Optimal robust value bounds)**.** *Let h_t be an expanded history node at time step t, and let π^L and π^U be the lower- and upper-bound policies defined in* (109),(110)*. Under rectangular transition uncertainty, the optimal robust value is bounded deterministically by:*

$$\mathcal{LRDB}_t^{\pi^L}(h_t) \leq V^{*,\text{rob}}_{t,rect}(h_t) \leq \mathcal{URDB}_t^{\pi^U}(h_t). \tag{108}$$

Subject to the policies: $\pi^U = \{\pi_k^U\}_{k=H-1}^t$ and $\pi^L = \{\pi_k^L\}_{k=H-1}^t$:

$$\pi_t^L(h_t) = \arg\max_{a_t \in \mathcal{A}_t} \left\{ \mathcal{LRDB}_t^{\pi^L}(h_t, a_t) \right\} = \arg\max_{a_t \in \mathcal{A}_t} \left\{ \widehat{Q}^{\pi^L, \widehat{P}_t^T, \mathcal{LRDB}}_{t,rect}(h_t, a_t) + \mathcal{L}^{\pi^L, \widehat{P}_t^T, \mathcal{LRDB}}_{t,rect}(h_t, a_t) \right\}, \tag{109}$$

$$\pi_t^U(h_t) = \arg\max_{a_t \in \mathcal{A}_t} \left\{ \mathcal{URDB}_t^{\pi^U}(h_t, a_t) \right\} = \arg\max_{a_t \in \mathcal{A}_t} \left\{ \widehat{Q}^{\pi^U, \widehat{P}_t^T, \mathcal{URDB}}_{t,rect}(h_t, a_t) + \mathcal{U}^{\pi^U, \widehat{P}_t^T, \mathcal{URDB}}_{t,rect}(h_t, a_t) \right\}, \tag{110}$$

Proof.
We prove by induction, that by applying the Bellman optimality operator on upper/lower bounds to

the robust value function, we get an upper/lower bounds on the **optimal** value function.

Base case ($t = H-1$):

the optimal action-value at the terminal node is:

$$Q^{*,\text{rob}}_{H-1,rect}(h_{H-1}, a_{H-1}) = r(s_{H-1}, a_{H-1}). \tag{111}$$

At the terminal time step $H-1$ of the MDP, there is no future transition term, therefore:

$$\mathcal{U}^{\pi^U}_{H-1,rect} = \mathcal{L}^{\pi^L}_{H-1,rect} = 0 \tag{112}$$

We set the terminal action-value to be the immediate reward:

$$\widehat{Q}^{\pi^{\text{L}},\text{rob}}_{H-1,rect}(h_{H-1}, a_{H-1}) = \widehat{Q}^{\pi^{\text{U}},\text{rob}}_{H-1,rect}(h_{H-1}, a_{H-1}) = r(s_{H-1}, a_{H-1}). \tag{113}$$

$$\mathcal{LRDB}^{\pi^L}_{H-1}(h_{H-1}, a_{H-1}) = \mathcal{URDB}^{\pi^U}_{H-1}(h_{H-1}, a_{H-1}) = r(s_{H-1}, a_{H-1}). \tag{114}$$

and the base-case for the induction is the best immediate reward:

$$\max_{a_{H-1}\in\mathcal{A}_{H-1}} \mathcal{LRDB}^{\pi^L}_{H-1}(h_{H-1}, a_{H-1}) \leq V^{*,\text{rob}}_{H-1,rect}(h_{H-1}) \leq \max_{a_{H-1}\in\mathcal{A}_{H-1}} \mathcal{URDB}^{\pi^U}_{H-1}(h_{H-1}, a_{H-1}) \tag{115}$$

$$V^{*,\text{rob}}_{H-1,rect}(h_{H-1}) \equiv \max_{a_{H-1}\in\mathcal{A}_{H-1}} r(s_{H-1}, a_{H-1}). \tag{116}$$

Induction Hypothesis: at time step $t+1$, for every $h_{t+1} = (h_t, a_t, s_{t+1})$ with $s_{t+1} \in \widehat{\mathcal{S}}(h^-_{t+1})$ we assume the following holds:

$$\mathcal{LRDB}^{\pi^L}_{t+1}(h_{t+1}) \leq V^{*,\text{rob}}_{t+1,rect}(h_{t+1}) \;\leq\; \mathcal{URDB}^{\pi^U}_{t+1}(h_{t+1}). \tag{117}$$

This means that at the next depth of the tree $\mathcal{LRDB}^{\pi^L}_{t+1}$, $\mathcal{URDB}^{\pi^U}_{t+1}$ are already valid lower/upper bounds on the optimal rectangular robust value.

Induction Step: We will now show that the hypothesis holds for time step t, specifically:

$$\mathcal{LRDB}^{\pi^L}_{t}(h_t, a_t) \leq Q^{*,\text{rob}}_{t,rect}(h_t, a_t) \leq \mathcal{URDB}^{\pi^U}_{t}(h_t, a_t) \tag{118}$$

Writing the the action value in the recursive Bellman form:

$$Q^{*,\text{rob}}_{t,\text{rect}}(h_t, a_t) = r(s_t, a_t) + \inf_{P^T_t \in \mathcal{P}^T_t} \mathbb{E}^{P^T_t}_{s_{t+1}|h_t}\left[V^{*,\text{rob}}_{t+1,\text{rect}}(h_{t+1})\right] = \tag{119}$$

$$r(s_t, a_t) + \inf_{P^T_t \in \mathcal{P}^T_t} \left(\sum_{s_{t+1}\in\widehat{\mathcal{S}}} P^T_t(s_{t+1}|s_t, a_t) V^{*,\text{rob}}_{t+1,\text{rect}}(h_{t+1}) + \sum_{s_{t+1}\notin\widehat{\mathcal{S}}} P^T_t(s_{t+1}|s_t, a_t) V^{*,\text{rob}}_{t+1,\text{rect}}(h_{t+1}) \right) \tag{120}$$

For simplicity we use the following abbreviation:

$$P^{out} = \sum_{s_{t+1}\notin\widehat{\mathcal{S}}} P^T_t(s_{t+1}|s_t, a_t)$$

by the induction hypothesis at time step $t+1$, and bound the out of support term by $r_{max}/r_{min} \cdot (H-1-t)$:

$$r(s_t, a_t) + \inf_{P^T_t \in \mathcal{P}^T_t} \left(\widehat{\mathbb{E}}^{P^T_t}_{s_{t+1}|h_t}\left[\mathcal{LRDB}^{\pi^L}_{t+1}(h_{t+1})\right] + P^{out} \cdot r_{min} \cdot (H-1-t) \right) \leq \tag{121}$$

$$Q^{*,\text{rob}}_{t,\text{rect}}(h_t, a_t) \leq$$

$$r(s_t, a_t) + \inf_{P^T_t \in \mathcal{P}^T_t} \left(\widehat{\mathbb{E}}^{P^T_t}_{s_{t+1}|h_t}\left[\mathcal{URDB}^{\pi^U}_{t+1}(h_{t+1})\right] + P^{out} \cdot r_{max} \cdot (H-1-t) \right),$$

following Lemma 5:

$$\mathcal{URDB}_t^{\pi^U}(h_t, a_t) \geq r(s_t, a_t) + \inf_{P_t^T \in \mathcal{P}_t^T} \left(\widehat{\mathbb{E}}^{P_t^T}_{s_{t+1}|h_t} \left[\mathcal{URDB}_{t+1}^{\pi^U}(h_{t+1}) \right] + P^{out} \cdot r_{max} \cdot (H - 1 - t) \right), \tag{122}$$

$$\mathcal{LRDB}_t^{\pi^L}(h_t, a_t) \leq r(s_t, a_t) + \inf_{P_t^T \in \mathcal{P}_t^T} \left(\widehat{\mathbb{E}}^{P_t^T}_{s_{t+1}|h_t} \left[\mathcal{LRDB}_{t+1}^{\pi^L}(h_{t+1}) \right] + P^{out} \cdot r_{min} \cdot (H - 1 - t) \right), \tag{123}$$

taking maximum over the actions gives:

$$V^{*,rob}_{t,rect}(h_t) = \max_{a_t \in \mathcal{A}_t} Q^{*,rob}_{t,rect}(h_t, a_t) \leq \max_{a_t \in \mathcal{A}_t} \mathcal{URDB}_t^{\pi^U}(h_t, a_t) \triangleq \mathcal{URDB}_t^{\pi^U}(h_t), \tag{124}$$

$$V^{*,rob}_{t,rect}(h_t) = \max_{a_t \in \mathcal{A}_t} Q^{*,rob}_{t,rect}(h_t, a_t) \geq \max_{a_t \in \mathcal{A}_t} \mathcal{LRDB}_t^{\pi^L}(h_t, a_t) \triangleq \mathcal{LRDB}_t^{\pi^L}(h_t), \tag{125}$$

which proves the induction step, and the theorem.

□

A.8 Proof of Lemma 5

Restating essential definition for the lemma:

1) $\mathcal{URDB}/\mathcal{LRDB}$ transition model:

$$\widehat{P}_t^{T,\mathcal{URDB}}(\cdot|s_t, a_t) = \arg\min_{\widehat{P}_t^T \in \widehat{\mathcal{P}}_t^T} \left\{ \widehat{\mathbb{E}}^{\widehat{P}_t^T}_{s_{t+1}|h_t} \left[\widehat{V}^{\pi^U, \widehat{P}_{t+1}^{T,\mathcal{URDB}}}_{t+1,rect}(h_{t+1}) \right] + \mathcal{U}^{\pi^U, \widehat{P}_t^T}_{t,rect}(h_t, a_t) \right\}, \tag{126}$$

$$\widehat{P}_t^{T,\mathcal{LRDB}}(\cdot|s_t, a_t) = \arg\min_{\widehat{P}_t^T \in \widehat{\mathcal{P}}_t^T} \left\{ \widehat{\mathbb{E}}^{\widehat{P}_t^T}_{s_{t+1}|h_t} \left[\widehat{V}^{\pi^L, \widehat{P}_{t+1}^{T,\mathcal{LRDB}}}_{t+1,rect}(h_{t+1}) \right] + \mathcal{L}^{\pi^L, \widehat{P}_t^T}_{t,rect}(h_t, a_t) \right\}, \tag{127}$$

2) For simplicity, we denote the probabily mass for the off-support by:

$$P^{out} = \sum_{s_{t+1} \notin \widehat{\mathcal{S}}} P_t^T(s_{t+1}|s_t, a_t)$$

Proof of Theorem 5.

Lemma 5. *Let h_t be the history at time step t. For any action $a_t \in \mathcal{A}_t$, and future action a_{t+1} selected by the greedy policies π^L, π^U, the following inequalities hold:*

$$\mathcal{LRDB}_t^{\pi^L}(h_t, a_t) \leq r(s_t, a_t) + \inf_{P_t^T \in \mathcal{P}_t^T} \left(\widehat{\mathbb{E}}^{P_t^T}_{s_{t+1}|h_t} \left[\mathcal{LRDB}_{t+1}^{\pi^L}(h_{t+1}) \right] + P^{out} \cdot r_{min} \cdot (H - 1 - t) \right), \tag{128}$$

And:

$$\mathcal{URDB}_t^{\pi^U}(h_t, a_t) \geq r(s_t, a_t) + \inf_{P_t^T \in \mathcal{P}_t^T} \left(\widehat{\mathbb{E}}^{P_t^T}_{s_{t+1}|h_t} \left[\mathcal{URDB}_{t+1}^{\pi^U}(h_{t+1}) \right] + P^{out} \cdot r_{max} \cdot (H - 1 - t) \right). \tag{129}$$

Proof.
We first focus on the **upper** bound:
By the definition of the $\mathcal{URDB}_t^{\pi^U}(h_t, a_t)$ (102), at time step $t+1$:

$$r(s_t,a_t)+\inf_{P_t^T\in\mathcal{P}_t^T}\Big(\widehat{\mathbb{E}}^{P_t^T}_{s_{t+1}|h_t}\Big[\mathcal{URDB}^{\pi^U}_{t+1}(h_{t+1})\Big]+P^{out}\cdot r_{max}\cdot(H-1-t)\Big)= \tag{130}$$

$$r(s_t,a_t)+\inf_{P_t^T\in\mathcal{P}_t^T}\Big(\widehat{\mathbb{E}}^{P_t^T}_{s_{t+1}|h_t}\Big[\max_{a_{t+1}\in\mathcal{A}_{t+1}}\Big\{\mathcal{URDB}^{\pi^U}_{t+1}(h_{t+1},a_{t+1})\Big\}\Big]+P^{out}\cdot r_{max}\cdot(H-1-t)\Big)= \tag{131}$$

$$r(s_t,a_t)+\inf_{P_t^T\in\mathcal{P}_t^T}\Big(\widehat{\mathbb{E}}^{P_t^T}_{s_{t+1}|h_t}\Big[\max_{a_{t+1}\in\mathcal{A}_{t+1}}\Big\{\widehat{Q}^{\pi^U,\widehat{P}^{T,\mathcal{URDB}}_{t+1}}_{t+1,rect}(h_{t+1},a_{t+1})+\mathcal{U}^{\pi^U,\widehat{P}^{T,\mathcal{URDB}}_{t+1}}_{t+1,rect}(h_{t+1},a_{t+1})\Big\}\Big]$$
$$+P^{out}\cdot r_{max}\cdot(H-1-t)\Big) \tag{132}$$

The action a_{t+1} that maximizes the expression is the same action that comes from the policy π^U, namely:

$$a^U_{t+1}=\pi^U_{t+1}(h_{t+1})=\arg\max_{a_{t+1}\in\mathcal{A}_t}\Big\{\widehat{Q}^{\pi^U,\widehat{P}^{T,\mathcal{URDB}}_{t+1}}_{t+1,rect}(h_{t+1},a_{t+1})+\mathcal{U}^{\pi^U,\widehat{P}^{T,\mathcal{URDB}}_{t+1}}_{t+1,rect}(h_{t+1},a_{t+1})\Big\} \tag{133}$$

Therefore we replace a_{t+1} with π^U_{t+1} $\Big(\widehat{Q}(h_{t+1},\pi^U_{t+1})\to\widehat{V}(h_{t+1})\Big)$:

$$r(s_t,a_t)+\inf_{P_t^T\in\mathcal{P}_t^T}\Big(\widehat{\mathbb{E}}^{P_t^T}_{s_{t+1}|h_t}\Big[\widehat{V}^{\pi^U,\widehat{P}^{T,\mathcal{URDB}}_{t+1}}_{t+1,rect}(h_{t+1})+\mathcal{U}^{\pi^U,\widehat{P}^{T,\mathcal{URDB}}_{t+1}}_{t+1,rect}(h_{t+1},\pi^U_{t+1})\Big]\Big)$$
$$+P^{out}\cdot r_{max}\cdot(H-1-t)\Big)= \tag{134}$$

we split the sum to in/out sampled support:

$$r(s_t,a_t)+\inf_{P_t^T\in\mathcal{P}_t^T}\Bigg(\sum_{s_{t+1}\in\widehat{\mathcal{S}}}P_t^T(s_{t+1}|s_t,a_t)\Big[\widehat{V}^{\pi^U,\widehat{P}^{T,\mathcal{URDB}}_{t+1}}_{t+1,rect}(h_{t+1})+\mathcal{U}^{\pi^U,\widehat{P}^{T,\mathcal{URDB}}_{t+1}}_{t+1,rect}(h_{t+1},\pi^U_{t+1})\Big]$$
$$+\sum_{s_{t+1}\notin\widehat{\mathcal{S}}}P_t^T(s_{t+1}|s_t,a_t)\Big[r_{max}\cdot(H-1-t)\Big]\Bigg)= \tag{135}$$

$$r(s_t,a_t)+\inf_{P_t^T\in\mathcal{P}_t^T}\Bigg(\sum_{s_{t+1}\in\widehat{\mathcal{S}}}P_t^T(s_{t+1}|s_t,a_t)\Big[\widehat{V}^{\pi^U,\widehat{P}^{T,\mathcal{URDB}}_{t+1}}_{t+1,rect}(h_{t+1})+r_{max}\cdot\Big(H-2-t-\delta^{\pi^U,\widehat{P}^{T,\mathcal{URDB}}_{t+1}}_{t+1}(s_{t+1}|h_{t+1})\Big)\Big]$$
$$+\Big(1-\sum_{s_{t+1}\in\widehat{\mathcal{S}}}P_t^T(s_{t+1}|s_t,a_t)\Big)\cdot\Big[r_{max}\cdot(H-1-t)\Big]\Bigg)= \tag{136}$$

$$r(s_t,a_t)+\inf_{P_t^T\in\mathcal{P}_t^T}\Bigg(\sum_{s_{t+1}\in\widehat{\mathcal{S}}}P_t^T(s_{t+1}|s_t,a_t)\cdot\widehat{V}^{\pi^U,\widehat{P}^{T,\mathcal{URDB}}_{t+1}}_{t+1,rect}(h_{t+1})$$
$$+r_{max}\Big(H-1-t-\sum_{s_{t+1}\in\widehat{\mathcal{S}}}P_t^T(s_{t+1}|s_t,a_t)\cdot\Big[1+\delta^{\pi^U,\widehat{P}^{T,\mathcal{URDB}}_{t+1}}_{t+1}(s_{t+1}|h_{t+1})\Big]\Big)\Bigg)\le \tag{137}$$

as defined in (63), δ uses infimum/suprimum based on the sign of r_{max} :

$$\begin{cases}\delta^{\pi^U}_{t+1}\le\delta^{\pi^U,\widehat{P}^{T,\mathcal{URDB}}_{t+1}}_{t+1}, & r_{max}\ge 0,\\ \delta^{\pi^U}_{t+1}\ge\delta^{\pi^U,\widehat{P}^{T,\mathcal{URDB}}_{t+1}}_{t+1}, & r_{max}<0.\end{cases} \tag{138}$$

Therefore:

$$r(s_t,a_t)+\inf_{P_t^T\in\mathcal{P}_t^T}\Bigg(\sum_{s_{t+1}\in\widehat{\mathcal{S}}}P_t^T(s_{t+1}|s_t,a_t)\cdot\widehat{V}_{t+1,rect}^{\pi^U,\widehat{P}_{t+1}^{T,\mathcal{URDB}}}(h_{t+1}) + r_{max}\Big(H-1-t-\sum_{s_{t+1}\in\widehat{\mathcal{S}}}P_t^T(s_{t+1}|s_t,a_t)\cdot\Big[1+\delta_{t+1}^{\pi^U}(s_{t+1}|h_{t+1})\Big]\Big)\Bigg)= \tag{139}$$

$$r(s_t,a_t)+\inf_{P_t^T\in\mathcal{P}_t^T}\Bigg(\sum_{s_{t+1}\in\widehat{\mathcal{S}}}P_t^T(s_{t+1}|s_t,a_t)\cdot\widehat{V}_{t+1,rect}^{\pi^U,\widehat{P}_{t+1}^{T,\mathcal{URDB}}}(h_{t+1})+\mathcal{U}_{t,rect}^{\pi^U,P_t^T}(h_t,a_t)\Bigg)= \tag{140}$$

Using Lemma 3 together with the restriction map $\mathcal{R}_t^T$ from 25, that project the full uncertainty set onto the projected uncertainty set, we get the following equalities:

$$r(s_t,a_t)+\inf_{P_t^T\in\mathcal{P}_t^T}\left(\widehat{\mathbb{E}}_{s_{t+1}|h_t}^{\mathcal{R}^T(P_t^T)}\left[\widehat{V}_{t+1,rect}^{\pi^U,\widehat{P}_{t+1}^{T,\mathcal{URDB}}}(h_{t+1})\right]+\mathcal{U}_{t,rect}^{\pi^U,\mathcal{R}^T(P_t^T)}(h_t,a_t)\right)= \tag{141}$$

$$r(s_t,a_t)+\inf_{\widehat{P}_t^T\in\widehat{\mathcal{P}}_t^T}\left(\widehat{\mathbb{E}}_{s_{t+1}|h_t}^{\widehat{P}_t^T}\left[\widehat{V}_{t+1,rect}^{\pi^U,\widehat{P}_{t+1}^{T,\mathcal{URDB}}}(h_{t+1})\right]+\mathcal{U}_{t,rect}^{\pi^U,\widehat{P}_t^T}(h_t,a_t)\right)\triangleq \tag{142}$$

$$\mathcal{URDB}_t^{\pi^U}(h_t,a_t), \tag{143}$$

which proves the upper bound.

The **lower** bound follows the same proof structure, with the following modifications:

1. Replace $\mathcal{URDB}$ with $\mathcal{LRDB}$.
2. Replace $\mathcal{U}_{t+1}$ with $\mathcal{L}_{t+1}$.
3. Replace π^U with π^L.
4. Replace r_{max} with r_{min}.
5. Change the inequality directions.
6. Analogue justification to the transition after (137):

$$\begin{cases}\delta_{t+1}^{\pi^L}\geq\delta_{t+1}^{\pi^L,\widehat{P}_{t+1}^{T,\mathcal{LRDB}}}, & r_{min}\geq 0,\\ \delta_{t+1}^{\pi^L}\leq\delta_{t+1}^{\pi^L,\widehat{P}_{t+1}^{T,\mathcal{LRDB}}}, & r_{min}<0.\end{cases} \tag{144}$$

□

A.9 Proof of Corollary 3 (Asymptotic optimality)

Corollary 3 (restated) (Asymptotic optimality)**.** *As the sampled state support expands to the full state space at every node, $\widehat{\mathcal{S}}(h_k^-) \uparrow \mathcal{S},\ k = t, \dots, H$, both deterministic bounds converge to the optimal rectangular robust value:*

$$\mathcal{LRDB}_t^{\pi^L}\left(h_t, \pi_t^L(h_t)\right) \longrightarrow V_{t,\text{rect}}^{*,\text{rob}}(h_t), \tag{145}$$

$$\mathcal{URDB}_t^{\pi^U}\left(h_t, \pi_t^U(h_t)\right) \longrightarrow V_{t,\text{rect}}^{*,\text{rob}}(h_t). \tag{146}$$

Proof.
We define the following expressions, essential for the proof:

$$\epsilon \triangleq \max_{k\in[t,H-1]} \sup_{\substack{\pi_k, a_k, h_k,\\ \widehat{P}_k^T \in \widehat{\mathcal{P}}_k^T}} \max\left\{\left|\mathcal{U}_{k,rect}^{\pi,\widehat{P}_k^T}(h_k,a_k)\right|, \left|\mathcal{L}_{k,rect}^{\pi,\widehat{P}_k^T}(h_k,a_k)\right|\right\} \tag{147}$$

$$G_k \triangleq \sup_{h_k}\left|\mathcal{URDB}_k^{\pi^U}(h_k) - \mathcal{LRDB}_k^{\pi^L}(h_k)\right|, \quad k \in [t, H-1] \tag{148}$$

We will show the following for the proof:

1. $G_t \le 4 \cdot (H-1-t)\cdot\epsilon$
2. $\epsilon \xrightarrow{\widehat{\mathcal{S}}\to\mathcal{S}} 0$
3. $V_{t,rect}^{*,\text{rob}}(h_t) \in [\mathcal{LRDB}_t^{\pi^L}(h_t), \mathcal{URDB}_t^{\pi^U}(h_t)]$, and by claim 1 and 2 $G_t \xrightarrow{\widehat{\mathcal{S}}\to\mathcal{S}} 0$, which proves the corollary.

Step 1:
base case $G_{H-1} = 0$ by equations 114.

For some $k \in [t, H-2]$:

$$\left|\mathcal{URDB}_k^{\pi^U}(h_k) - \mathcal{LRDB}_k^{\pi^L}(h_k)\right| \le \max_{a_k}\left|\mathcal{URDB}_k^{\pi^U}(h_k,a_k) - \mathcal{LRDB}_k^{\pi^L}(h_k,a_k)\right| \tag{149}$$

we continue with arbitrary a_k for now:

$$\begin{aligned}\max_{a_k}\Bigg|&\inf_{\widehat{P}_k^T\in\widehat{\mathcal{P}}_k^T}\left(\widehat{\mathbb{E}}_{s_k|h_k}^{\widehat{P}_k^T}\left[\widehat{V}_{k+1,rect}^{\pi^U,\widehat{P}_{k+1}^T,\mathcal{URDB}}(h_{k+1})\right] + \mathcal{U}_{k,rect}^{\pi^U,\widehat{P}_k^T}(h_k,a_k)\right)\\ &- \inf_{\widehat{P}_k^T\in\widehat{\mathcal{P}}_k^T}\left(\widehat{\mathbb{E}}_{s_k|h_k}^{\widehat{P}_k^T}\left[\widehat{V}_{k+1,rect}^{\pi^L,\widehat{P}_{k+1}^T,\mathcal{LRDB}}(h_{k+1})\right] + \mathcal{L}_{k,rect}^{\pi^L,\widehat{P}_k^T}(h_k,a_k)\right)\Bigg| \le\end{aligned} \tag{150}$$

$$\begin{aligned}\max_{a_k}\sup_{\widehat{P}_k^T\in\widehat{\mathcal{P}}_k^T}\Bigg|\Bigg(&\widehat{\mathbb{E}}_{s_k|h_k}^{\widehat{P}_k^T}\left[\widehat{V}_{k+1,rect}^{\pi^U,\widehat{P}_{k+1}^T,\mathcal{URDB}}(h_{k+1})\right] + \mathcal{U}_{k,rect}^{\pi^U,\widehat{P}_k^T}(h_k,a_k)\\ &-\widehat{\mathbb{E}}_{s_k|h_k}^{\widehat{P}_k^T}\left[\widehat{V}_{k+1,rect}^{\pi^L,\widehat{P}_{k+1}^T,\mathcal{LRDB}}(h_{k+1})\right] + \mathcal{L}_{k,rect}^{\pi^L,\widehat{P}_k^T}(h_k,a_k)\Bigg)\Bigg| \le\end{aligned} \tag{151}$$

using linearity of expectation, and definition of (147):

$$\sup_{\widehat{P}_k^T\in\widehat{\mathcal{P}}_k^T}\left|\left(\widehat{\mathbb{E}}_{s_k|h_k}^{\widehat{P}_k^T}\left[\widehat{V}_{k+1,rect}^{\pi^U,\widehat{P}_{k+1}^T,\mathcal{URDB}}(h_{k+1}) - \widehat{V}_{k+1,rect}^{\pi^L,\widehat{P}_{k+1}^T,\mathcal{LRDB}}(h_{k+1})\right]\right)\right| + 2\epsilon \le \tag{152}$$

Using equations 102, 104, and we also get rid of the sup:

$$\Big|\Big(\mathcal{URDB}^{\pi^U}_{k+1}(h_{k+1}) - \mathcal{U}^{\pi^U,\widehat{P}^{T,\mathcal{URDB}}_{k+1}}_{k+1,rect}(h_{k+1},\pi_{k+1})\Big) - \Big(\mathcal{LRDB}^{\pi^L}_{k+1}(h_{k+1}) - \mathcal{L}^{\pi^L,\widehat{P}^{T,\mathcal{LRDB}}_{k+1}}_{k+1,rect}(h_{k+1},\pi_{k+1})\Big)\Big| + 2\epsilon \leq \tag{153}$$

$$\Big|\mathcal{URDB}^{\pi^U}_{k+1}(h_{k+1}) - \mathcal{LRDB}^{\pi^L}_{k+1}(h_{k+1})\Big| + 4\epsilon \leq \tag{154}$$

$$\sup_{h_{k+1}}\Big|\mathcal{URDB}^{\pi^U}_{k+1}(h_{k+1}) - \mathcal{LRDB}^{\pi^L}_{k+1}(h_{k+1})\Big| + 4\epsilon = \tag{155}$$

$$G_{k+1} + 4\epsilon. \tag{156}$$

Unrolling $H-1$ times from G_{H-1} gives:

$$G_t \leq 4\cdot(H-1-t)\cdot\epsilon \tag{157}$$

Step 2:

Restating the rectangular upper and lower bounds (61) in theorem 1, with the fixed model version:

$$\mathcal{U}^{\pi,\widehat{P}^T_k}_{k,rect}(h_k) \triangleq r_{max}\Big(H-1-k-\delta^{\pi,\widehat{P}^T_k}_k(s_k|h_k)\Big) \tag{158}$$

$$\mathcal{L}^{\pi,\widehat{P}^T_k}_{k,rect}(h_k) \triangleq r_{min}\Big(H-1-k-\delta^{\pi,\widehat{P}^T_k}_k(s_k|h_k)\Big) \tag{159}$$

$$\delta^{\pi,\widehat{P}^T_k}_k(s_k|h_k) = \sum_{s_{k+1}\in\widehat{\mathcal{S}}(h_{\bar{k+1}})} \widehat{P}^T_k\big(s_{k+1}\mid s_k,\pi_k(h_k)\big)\,[1+\delta^{\pi}_{k+1}(s_{k+1})] \tag{160}$$

$$\delta^{\pi,\widehat{P}^T_k}_{H-1}(s_{H-1}) \equiv 0 \tag{161}$$

As the sampled state support increases:

$$\sum_{s_{k:H-1}\in\widehat{\mathcal{S}}(h_{\bar{H-1}})}\prod_{i=k}^{H-1} P^T_i\big(s_{i+1}\mid s_i,\pi_i(h_i)\big) \xrightarrow{\widehat{\mathcal{S}}\to\mathcal{S}} 1, \tag{162}$$

therefore:

$$\delta^{\pi,\widehat{P}^T_k}_k(s_k|h_k) \xrightarrow{\widehat{\mathcal{S}}\to\mathcal{S}} H-1-k, \tag{163}$$

which induces:

$$\mathcal{U}^{\pi,\widehat{P}^T_k}_{k,rect}(h_k) \xrightarrow{\widehat{\mathcal{S}}\to\mathcal{S}} 0, \tag{164}$$

$$\mathcal{L}^{\pi,\widehat{P}^T_k}_{k,rect}(h_k) \xrightarrow{\widehat{\mathcal{S}}\to\mathcal{S}} 0. \tag{165}$$

Therefore:

$$\epsilon \xrightarrow{\widehat{\mathcal{S}}\to\mathcal{S}} 0$$

Step 3:

$$0 \leq \mathcal{URDB}^{\pi^U}_t(h_t) - V^{*,\mathrm{rob}}_{t,rect}(h_t) \leq G_t, \tag{166}$$

$$0 \leq V^{*,\mathrm{rob}}_{t,rect}(h_t) - \mathcal{LRDB}^{\pi^L}_t(h_t) \leq G_t, \tag{167}$$

we have shown that $G_t \xrightarrow{\widehat{\mathcal{S}}\to\mathcal{S}} 0$, which then concludes the proof. □

B Robust Backup under TV Uncertainty

For the TV uncertainty sets used in our experiments, the inner optimization in (19), (20), can be solved directly on the sampled successor support. Let

$$\mathring{p}_{\text{out}} = 1 - \sum_{s' \in \widehat{\mathcal{S}}(h,a)} \mathring{P}^T(s' \mid s, a)$$

denote the nominal probability mass outside the currently sampled support. Since projection restricts the transition row without renormalization, all unsampled successors can be represented by a single aggregate outside state. Consequently, the projected uncertainty set is equivalently represented by a TV ball over the $|\widehat{\mathcal{S}}(h,a)|+1$ dimensional vector consisting of the sampled successor probabilities and the aggregate outside mass. The aggregate outside state represents the omitted probability mass, with its contribution determined by the corresponding projected backup.

The resulting inner problem is a linear optimization over a TV ball. We solve it by the standard mass-transfer procedure: probability mass of at most ρ is shifted from successors with the largest continuation values toward those with the smallest values, subject to the simplex constraints. Sorting the continuation values gives a per-backup complexity of $O(|\widehat{\mathcal{S}}(h,a)| \log |\widehat{\mathcal{S}}(h,a)|)$ and $O(|\widehat{\mathcal{S}}(h,a)|)$ memory. Thus, the robust backup depends on the currently sampled successor support rather than on the cardinality of the full state space.

C Experiments

C.1 Frozen Lake Domain

We evaluate the planners on a family of Frozen Lake domains represented by two-dimensional grids containing traversable frozen cells, terminal holes, a start cell, and a goal cell. Each grid cell corresponds to an MDP state, and we additionally introduce an absorbing terminal state s_{done}. The action space is $\mathcal{A} = \{\text{LEFT}, \text{DOWN}, \text{RIGHT}, \text{UP}\}$, corresponding to the four cardinal movement directions. Reaching either a hole or the goal terminates the episode, after which the process transitions to s_{done}. The objective is to reach the goal while avoiding holes. Different experiments vary the grid size, obstacle layout, start and goal locations, and planning horizon; their specific configurations are summarized in Section C.6.

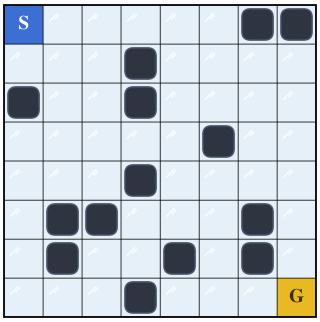


Figure 4: Frozen Lake Map Example

C.2 Ring Transition Model

The standard Frozen Lake dynamics have a small successor support. To evaluate planning with substantially larger transition supports, we use a stochastic *ring transition model*. For a state s, let

$$\mathcal{R}_d(s) = \{s' : \|s' - s\|_\infty = d\}, \qquad d = 1, \ldots, k,$$

denote the Chebyshev ring of radius d around s. For an interior state, $\mathcal{R}_d(s)$ contains $8d$ cells.

Under action a, the intended neighboring cell receives probability $1 - p_{\text{slip}}$. The remaining probability p_{slip} is distributed over the k rings. The fraction assigned to ring d is

$$q_d = \frac{\beta^{d-1}}{\sum_{\ell=1}^{k} \beta^{\ell-1}},$$

and this mass is distributed uniformly among the $8d$ cells in that ring. Thus, for an interior state, each cell in ring d receives slip probability

$$p_{\text{slip}} \frac{q_d}{8d}.$$

The intended successor is itself contained in the first ring and therefore receives both its ring contribution and the additional $1 - p_{\text{slip}}$ intended-transition probability.

Near the grid boundary, ring locations falling outside the grid are clipped to the nearest valid cell, and probability masses mapping to the same cell are accumulated. Holes block transitions: if a transition crosses an adjacent hole, the transition is redirected to that hole, which terminates the episode.

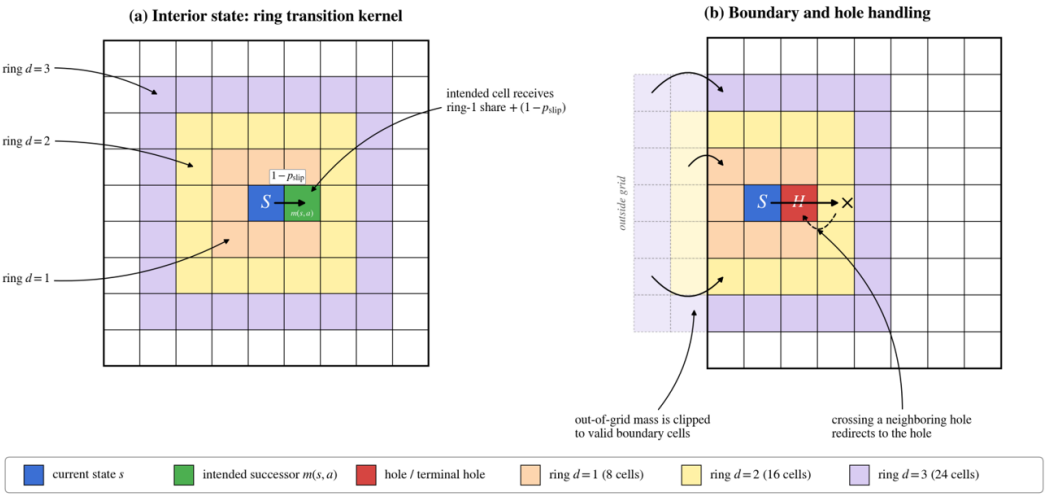


Figure 5: Illustration of the ring transition model used in the Frozen Lake experiments.

C.3 Transition Uncertainty Model

To account for model uncertainty, we construct a rectangular uncertainty set independently around each nominal transition row. Specifically, for every state-action pair (s, a),

$$\mathcal{P}^T(s,a) = \left\{ P^T(\cdot \mid s,a) : \mathrm{TV}\Big(P^T(\cdot \mid s,a), \mathring{P}^T(\cdot \mid s,a)\Big) \le \rho \right\}, \tag{168}$$

where ρ is the uncertainty radius and

$$\mathrm{TV}(p,q) = \frac{1}{2}\|p - q\|_1.$$

The uncertainty set preserves the support of the nominal transition model: states assigned zero probability by $\mathring{P}^T(\cdot \mid s,a)$ cannot receive probability mass under an admissible model. Thus, robustness accounts for uncertainty in the transition probabilities over the ring-kernel support without introducing additional successor states.

In experiments involving model mismatch, the environment is executed using a slip probability p_{true}, whereas the planner constructs its nominal model using p_{slip}. The uncertainty radius ρ is chosen to contain the true model within the corresponding TV uncertainty set. The exact values used in each experiment are reported in Section C.6.

C.4 Reward Model

We use two distance-based reward models across the experiments.

Inverse-Manhattan reward. For the main planner comparison and Ablations 1–2, we use the distance-shaped reward of Shazman et al. Shazman et al. (2026). For a nonterminal free cell s, let $m(s)$ denote its Manhattan distance to the goal, computed directly on the grid without treating holes as obstacles:

$$r_{\text{IM}}(s) = w\left(\frac{1}{1+m(s)}\right)^p. \tag{169}$$

We use $w = 0.1$ and $p = 3$, with goal reward 3 and rewards 0 for holes and s_{done}. The cubic inverse-distance shaping decays rapidly away from the goal, so most free cells receive only a small intermediate reward and the task remains predominantly driven by reaching the terminal goal.

Positive-distance reward. For several experiments we instead use the shortest-path distance $d(s)$ from s to the goal, computed while routing around holes. Let $d_{\max}$ be the largest such distance on the map and define

$$c(s) = \frac{d_{\max} - d(s)}{d_{\max}} \in [0, 1].$$

The positive-distance reward is

$$r_+(s) = w d_{\max} c(s)^p.$$

Thus, free-cell rewards increase as the agent approaches the goal, with the distance computed according to the traversable structure of the map rather than directly through holes. Goal and hole rewards are specified separately and override the shaping value, while $r(s_{\text{done}}) = 0$.

When a fixed reward span is used, we set $w = \text{span}/d_{\max}$, making the free-cell reward range independent of the map diameter. The specific reward model and parameters used by each experiment are summarized in Section C.6.

C.5 Planning Protocol

All online planning experiments use a receding-horizon protocol. At each environment step, the planner constructs a finite search tree of depth H rooted at the current state and performs planning using its available computation budget. The selected root action is then executed in the environment, the resulting next state is observed, and a new depth-H planning problem is initialized from that state. Thus, H denotes the planner's lookahead horizon rather than the episode length.

For model-mismatch experiments, planning is performed using the nominal transition model $\mathring{P}^T$ parameterized by p_{slip}, whereas executed transitions are sampled from the environment model parameterized by p_{true}. Episodes terminate upon reaching either the goal or a hole.

C.6 Experiment Configurations

Table 1 summarizes the parameters used in the main experiments. Unless stated otherwise, all experiments use the domain, transition kernel, uncertainty model, and planning protocol described above.

Table 1: Configuration of the experiments reported in the main paper.

Parameter	Fig. 1a	Fig. 1b	Fig. 2	Fig. 3
Grid	8×8	8×8	16×16	(10×10)–$(10^3 \times 10^3)$
H	6	3	$\{2, 5, 11\}$	5
k	1	2	8	$\frac{n-1}{2}$
β	1	1	1	0.39–0.99
p_{slip}	0.3	0.2	0.01–0.6	0.3
p_{true}	–	–	0.05	–
ρ	0.1	0.1	0.04–0.54	0.1
Budget	–	varied	20–200	80
Runs	50	20	20	5
Reward	r_+	r_+	r_{IM}	r_+

Figure 1a uses r_+ with goal reward 1, hole reward -1, $\text{span} = 1$, and $p = 0.5$; Figure 1b uses the same specification with $p = 1$. Figure 2 uses r_{IM} with goal reward 3, hole reward 0, $w = 0.1$, and $p = 3$. Figure 3 uses r_+ with goal reward 1, hole reward 0, and $p = 1$.

C.6.1 Ablation 1: Robustness to Environment Slipperiness and Model Mismatch

We first examine how planner performance changes as the true environment becomes increasingly slippery, and how this interacts with model mismatch. We fix the planning depth to 3 and use the 16×16 Frozen Lake domain. For the main experiment, the ring radius is $k = 8$ with $\beta = 1$. We vary the true environment slip probability as $p_{\text{true}} \in \{0.01, 0.1, 0.2, 0.3\}$, corresponding to the four columns of Figure 6. Within each column, p_{true} is fixed while the planner-assumed slip probability p_{slip} is varied from 0.01 to 0.6. Hence, moving across columns makes the true environment increasingly difficult, whereas moving along the x-axis changes the model assumed by the planner while leaving the true environment unchanged. For each pair, the

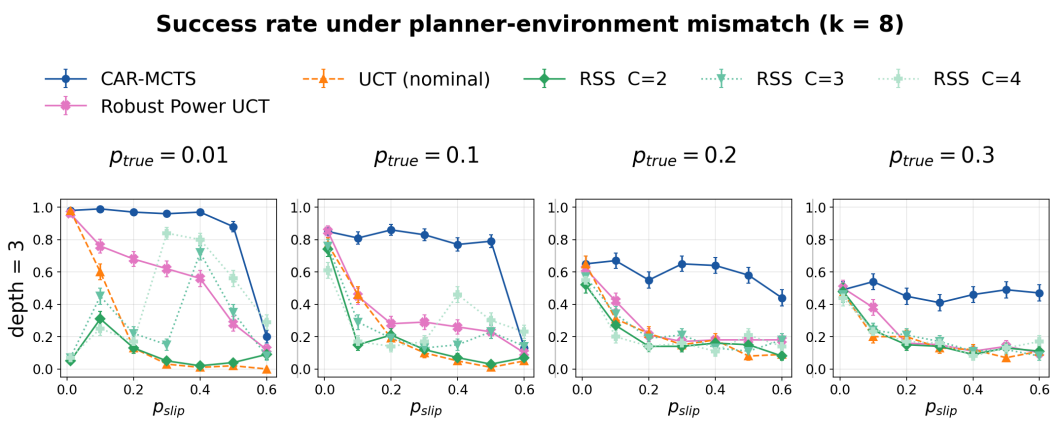


Figure 6: Success rate at planning depth 3 as the true environment becomes more slippery and the planner-assumed transition model varies. Columns correspond to $p_{\text{true}} \in \{0.01, 0.1, 0.2, 0.3\}$, while the x-axis shows the planner-assumed p_{slip}. The ring radius is $k = 8$. 500 seeds.

TV uncertainty radius is set to the smallest value that contains the true transition model. Each configuration is evaluated over 500 independent seeds.

As p_{true} increases, success rates decrease for all methods, reflecting the increasing difficulty of controlling the agent when the true environment becomes more slippery. CAR-MCTS degrades substantially more gracefully than the baselines and achieves the highest or near-highest success rate across the tested environment difficulties. Importantly, this behavior persists as p_{slip} moves away from p_{true}: within each column the true environment is unchanged, yet CAR-MCTS remains comparatively stable over increasing mismatch between the planning and true environment models.

The RSS variants are not restricted to the same sampling budget as the MCTS methods. CAR-MCTS, nominal UCT, and Robust Power UCT use 20 simulations per decision, corresponding to 60 transition samples at depth 3. In contrast, the recursive sparse-sampling expansion requires substantially more transition samples as C increases:

Method	Samples per decision	Relative to MCTS
MCTS methods	60	1×
RSS $C = 2$	584	9.7×
RSS $C = 3$	1884	31.4×
RSS $C = 4$	4368	72.8×

Table 2: Transition samples per decision at planning depth 3.

Despite this substantially larger sampling budget, RSS remains consistently below CAR-MCTS across the tested configurations. With $C = 2$, RSS is often comparable to nominal UCT and in several configurations performs worse. Thus, although RSS explicitly performs robust planning, this does not translate into a consistent empirical robustness advantage over the nominal planner under the tested model mismatch. Increasing the branching factor to $C = 3$ or $C = 4$ generally improves RSS, but the gains are inconsistent; even with tens of times more transition samples, its success rate frequently remains close to that of nominal UCT.

Effect of transition-support width. To test whether the behavior of RSS is primarily caused by the broad successor support of the $k = 8$ kernel, we repeat the same experiment with $k = 1$, while keeping the remaining settings unchanged, shown in Figure 7.

Reducing the ring radius substantially improves several competing methods, showing that their performance is sensitive to the number of feasible next states. In contrast, CAR-MCTS changes remarkably little between $k = 8$ and $k = 1$. This is an important strength of the proposed approach: CAR-MCTS maintains similar performance even when the transition support grows dramatically, whereas the competing sampling-based planners benefit substantially from the easier, narrower-support setting. RSS also improves in several configurations, but $C = 2$ remains frequently comparable to or worse than nominal UCT, while the larger-C variants still do not consistently outperform it. Hence, the behavior observed for RSS at $k = 8$ cannot be explained solely by insufficient sampling of a broad successor distribution.

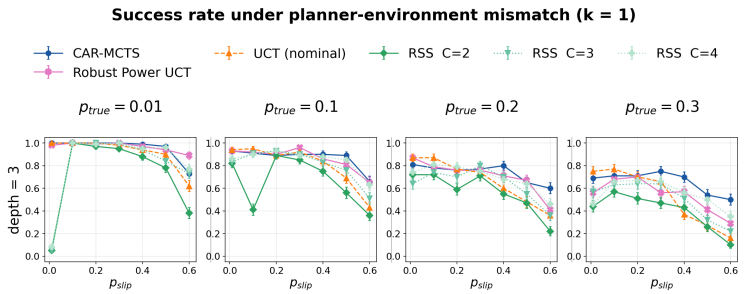


Figure 7: The same experiment as Figure 6, with the ring radius reduced from $k = 8$ to $k = 1$.

Overall, the ablation highlights three complementary strengths of CAR-MCTS. First, its performance degrades more gracefully as the true environment becomes increasingly slippery. Second, for a fixed true environment, it remains robust as the planner's assumed transition model moves away from the true dynamics. Third, its performance is far less sensitive to the width of the transition support: increasing the number of feasible next states has only a limited effect on CAR-MCTS, while substantially degrading the competing methods. Together, these results show that CAR-MCTS remains effective under both model uncertainty and broad transition dynamics, without requiring the large increase in successor samples used by RSS.

C.6.2 Ablation 2: Scalability with Planning Horizon

We next isolate the effect of the planning horizon on online computation. All environment and planner parameters are held fixed while varying $H \in \{1, \ldots, 10\}$. Experiments use a 100×100 Frozen Lake domain with hole density 0.15, ring radius $k = 8$, and $\beta = 1$. The planner uses $p_{\text{slip}} = 0.3$, while the true execution dynamics use $p_{\text{true}} = 0.1$, giving $\rho = 0.197$. CAR-MCTS and Robust Power UCT use 20 MCTS simulations per decision, and results are averaged over 20 independent seeds.

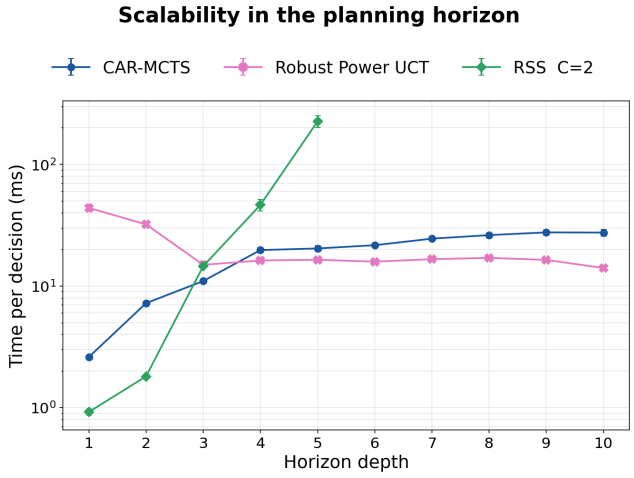


Figure 8: Decision time as a function of planning depth. CAR-MCTS and Robust Power UCT use 20 simulations per decision, while RSS uses branching factor $C = 2$. A 20-second computation cap is imposed per decision; RSS reaches this limit at depth 6, and deeper configurations are therefore not evaluated.

Figure 8 shows a clear difference in scalability with planning depth. CAR-MCTS remains within the tens-of-milliseconds regime throughout the entire sweep, with decision time increasing only gradually as the depth grows. Robust Power UCT similarly remains approximately constant over the tested range. In contrast, RSS is initially the fastest method at small depths, but its computational cost grows rapidly with each additional planning level, increasing from below 1 ms at depth 1 to more than 200 ms at depth 5. We impose a 20-second computation cap per decision; at depth 6, RSS reaches this limit before completing the planning step. Consequently, deeper RSS configurations are not evaluated.

The difference stems from the underlying search structure. RSS recursively samples C successor states for every action at each level of the tree, so its computational cost grows rapidly with depth. In contrast, CAR-MCTS uses a fixed simulation budget: increasing the planning depth lengthens the sampled trajectories and their backups, but does not require recursively expanding all sampled successor combinations. Consequently, CAR-MCTS supports substantially deeper lookahead while retaining practical online decision times.

C.6.3 Ablation 3: True Optimal-Action Selection

Finally, we evaluate the quality of the action selected by each planner directly, rather than through the stochastic outcome of an episode. For each planning problem, we compute the optimal action offline using the true transition model and measure the fraction of decisions in which each planner selects this action. Importantly, the true model is used only as an evaluation oracle and is not available to any of the planners.

We use a 4×4 Frozen Lake domain with ring radius $k = 1$, search depth 3, and 20 simulations for the MCTS-based methods. RSS uses $C = 2$, and each configuration is evaluated over 100 independent seeds. As in Ablation 1 C.6.1, the columns vary the true environment slip probability $p_{\text{true}} \in \{0.01, 0.1, 0.2, 0.3\}$, while the x-axis varies the planner-assumed p_{slip}.

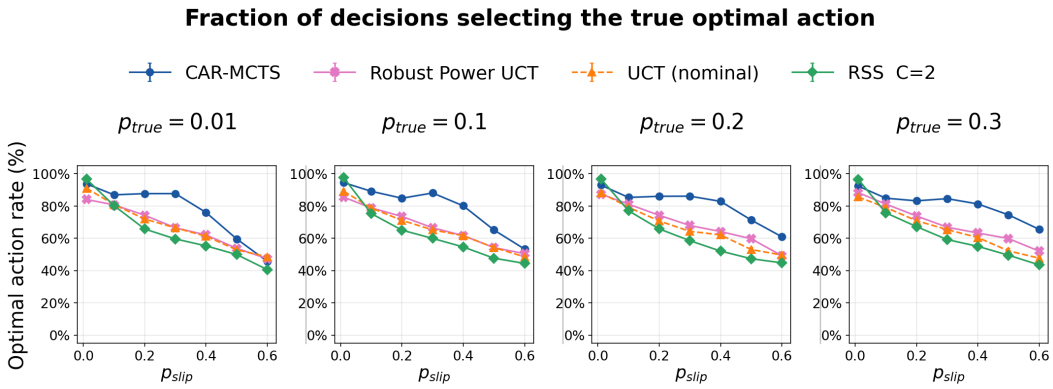


Figure 9: Fraction of decisions selecting the action that is optimal under the true transition model. Columns correspond to different values of p_{true}, while the x-axis varies the planner-assumed p_{slip}. The true model is used only for evaluation.

Figure 9 shows that CAR-MCTS selects the true optimal action more reliably than the competing planners over nearly the entire sweep. When the true environment is only mildly slippery, all methods perform well when the planning model is accurate, but their behavior diverges as the assumed model moves away from the true dynamics. CAR-MCTS degrades more gradually and maintains a substantially higher optimal-action selection rate under increasing model mismatch.

The difference becomes particularly pronounced as the true environment itself becomes more slippery. For $p_{\text{true}} = 0.2$ and 0.3, CAR-MCTS maintains a clear advantage over a broad range of p_{slip} values, showing that it continues to identify the correct decision even when both the environment is more difficult and the planner model is inaccurate. Robust Power UCT generally improves over nominal UCT in parts of the sweep, but does not approach the consistency of CAR-MCTS. RSS with $C = 2$ is typically comparable to or below nominal UCT, indicating that its robust formulation does not translate into a consistent advantage in identifying the true optimal action under the tested planning budget.

This experiment complements the episode-level results of Ablation 1 C.6.1. Higher success rates can in principle result partly from favorable stochastic transitions after an action is selected; here, the evaluation isolates the planning decision itself. The results therefore show that CAR-MCTS's advantage is not limited to better realized outcomes: it more frequently selects the action that is optimal under the true environment dynamics. Together with the previous ablations, this demonstrates stronger decision quality under model mismatch, reduced sensitivity to broad transition support, and practical scalability with planning depth.

C.6.4 Ablation 4: Certification and Pruning in Practice

Finally, we examine how frequently the deterministic bounds used by CAR-MCTS lead to root-action certification and branch pruning in practice. We evaluate three grid sizes, 4×4, 8×8, and 16×16, while varying p_{slip} and the planning budget over $\{100, 200, 500, 1000\}$ simulations. We report the fraction of decisions for which the root action is certified and the fraction of branches pruned during search.

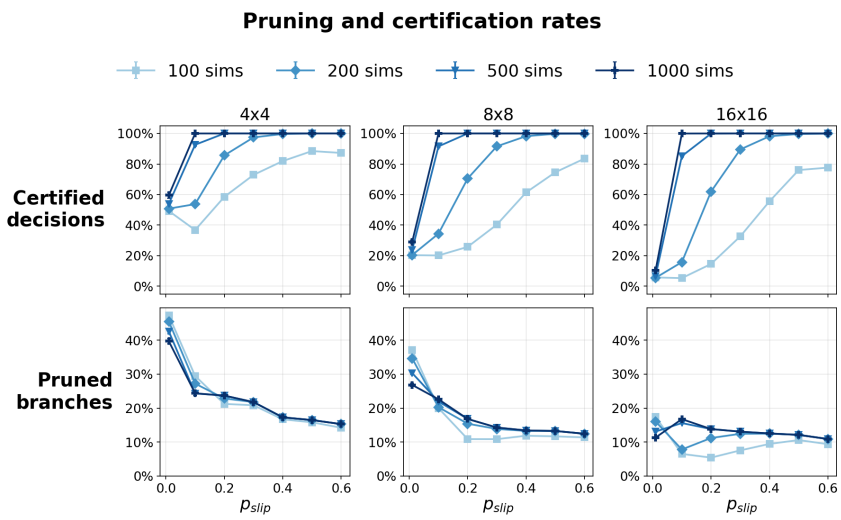


Figure 10: Root-action certification rate (top) and branch-pruning rate (bottom) for different grid sizes, planning budgets, and values of p_{slip}.

Figure 10 shows that root certification occurs frequently in practice and increases strongly with the available planning budget. Even with relatively small budgets of 100–200 simulations, CAR-MCTS certifies a substantial fraction of root decisions, with certification rates often around 50% or higher depending on the configuration. As the budget increases, the deterministic intervals tighten further and the certification rate rises sharply. At 500 and 1000 simulations, CAR-MCTS certifies the root decision in nearly all runs over a large portion of the tested configurations. This demonstrates that the certificates are not only theoretical guarantees, but can become sufficiently tight during online planning to provide an effective early-stopping criterion.

The lower row shows that the same bounds also eliminate a meaningful fraction of suboptimal branches during search. Interestingly, the measured pruning rate tends to decrease when certification becomes frequent. This is consistent with the two mechanisms playing complementary roles: while the root decision remains unresolved, continued sampling contracts the action-value bounds and enables dominated branches to be pruned; once the root action is certified, planning terminates and no further pruning is needed.

Overall, the ablation shows that CAR-MCTS uses its deterministic certificates in two complementary ways to reduce online computation: locally, by pruning branches that can no longer contain the optimal action, and globally, by terminating planning once the root action is certified. The high certification rates achieved with larger simulation budgets show that both mechanisms are actively useful in practice rather than serving only as theoretical guarantees.