The Statistical Value of Lyapunov Stability in Reinforcement Learning

Anonymous authors
Paper under double-blind review

Abstract

Reinforcement learning for long-running tasks, such as robotic and queueing control, must achieve high reward while maintaining stability. In many such tasks, state variables such as motion tracking errors and queue lengths may be unbounded, making it challenging to learn a globally stable and near-optimal policy from finite data. Counterintuitively, we show that Lyapunov stability makes learning in countably infinite MDPs effectively finite: negative Lyapunov drift concentrates long-run occupancy within a finite region, which we call a *Lyapunov core*. Building on this insight, we develop a model-based algorithm LyCore, which learns from samples collected only within an accuracy-dependent Lyapunov core and uses a known stabilizing policy elsewhere, with guarantees of global stability and near-optimality among Lyapunov-stable policies. We further establish matching upper and lower bounds on the sample complexity up to logarithmic factors, separating the fundamental costs of stability and performance learning. Our algorithm attains these bounds by combining uniform drift sampling with shell-dependent model sampling, which allocates fewer model samples to outer, rarely visited regions. The bounds reveal an interesting phase transition: performance learning dominates at high accuracy in low effective dimensions, whereas stability learning can dominate in higher effective dimensions. Experiments on a multi-shell synthetic MDP support the predicted rates, while a nonlinear IEEE 33-bus feeder voltage-control task illustrates the method under realistic model uncertainty.

1 Introduction

Reinforcement learning (RL) has achieved remarkable empirical success in sequential decision-making problems (Mnih et al., 2015; Silver et al., 2016; Schulman et al., 2017). However, applying RL to many physical systems often requires more than high reward: a learned policy must also maintain stable and reliable behavior during operation (Berkenkamp et al., 2017; Chow et al., 2018; Dean et al., 2019). For example, a robotic controller should improve task performance while maintaining stable motion under disturbances. Similarly, a data-center scheduler should improve energy efficiency while maintaining job queue stability (Tassiulas & Ephremides, 1992; Stolyar, 2004; Gandhi et al., 2009; Wierman et al., 2009). Across these domains, stability is a fundamental requirement for reliable online decision making.

Learning stable policies in such systems poses two intertwined challenges. First, state variables such as queue backlogs or robotic tracking errors may be unbounded, so finite data can characterize the dynamics only within a limited region (Shah et al., 2020; Liu et al., 2022; Pavse et al., 2024). Second, both the performance and stability depend on the policy-induced dynamics of the full state space. Under unknown transitions, an action that improves reward may also drive the state away from the desired stable region, so learning high-reward decisions does not by itself ensure stability. This raises a fundamental question:

Can finite data suffice to learn a globally stable, near-optimal policy in an unbounded state space, and how does stability shape the required sample complexity?

We answer this question by developing a model-based framework for stability-constrained reinforcement learning in MDPs with countably infinite states. Our key insight is that Lyapunov stability makes global policy learning effectively finite: negative Lyapunov drift concentrates long-run occu-

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.

pancy within a finite region, which we call a Lyapunov core. Building on this principle, LyCore learns from samples collected only within an accuracy-dependent Lyapunov core and uses a known stabilizing policy elsewhere. We prove that the resulting policy is globally stable and near-optimal among Lyapunov-stable policies. We also establish matching upper and lower bounds on the sample complexity that separate the fundamental costs of performance learning and stability learning.

Our contributions can be summarized as follows:

- **Finite learning under Lyapunov stability.** We prove that Lyapunov drift concentrates the long-run occupancy of stable policies within a finite Lyapunov core. The average reward of every Lyapunov-stable policy can be approximated to any desired accuracy by using a known stabilizing policy outside a sufficiently large Lyapunov core. This principle makes finite-sample learning possible over a countably infinite state space.
- **Stability-constrained learning algorithm.** We develop LyCore, which estimates the environment model and Lyapunov drift within a Lyapunov core and computes a policy through a stability-constrained average-reward linear program. Its sampling scheme combines uniform drift sampling with shell-dependent model sampling, which allocates fewer samples to state-action pairs in outer, rarely visited regions.
- **Matching minimax sample complexity.** We establish a high-probability sample-complexity upper bound for the proposed algorithm and prove a matching lower bound up to logarithmic factors. The bounds identify performance learning and stability learning as two intrinsic statistical costs. They reveal a phase transition governed by the growth rate of the Lyapunov core: performance learning dominates at high accuracy under slow growth, whereas stability learning can dominate under faster growth. The key technical ingredient is a shell-weighted concentration analysis using drift-controlled return times, which controls long-run model errors without requiring a contraction assumption.

Experiments on a controlled multi-shell MDP support the predicted sample complexity rates, while a nonlinear IEEE 33-bus feeder voltage-control study illustrates the method under realistic model uncertainty.

1.1 Related Work

Lyapunov stability and safe reinforcement learning. Safe and constrained RL studies policy learning under cumulative-cost and resource constraints, as well as safety requirements during learning or deployment (Altman, 1999; García & Fernández, 2015; Achiam et al., 2017; Chow et al., 2018; Brantley et al., 2020; Ding et al., 2021; Agarwal et al., 2022; Vaswani et al., 2022). Unlike cumulative-cost constraints, Lyapunov stability imposes a statewise constraint on the closed-loop dynamics. Lyapunov conditions have also been used to design policy updates, stabilizing controllers, cost shaping, and exploration methods under unknown dynamics (Perkins & Barto, 2002; Berkenkamp et al., 2017; Jin & Lavaei, 2020; Han et al., 2020; 2021; Lale et al., 2022; 2023; Westenbroek et al., 2023; Pavse et al., 2024; Zhang & Fan, 2024). Finite-sample stability guarantees have been developed for imitation learning and structured nonlinear control systems (Tu et al., 2022; Lale et al., 2023; Han et al., 2026). In MDPs, prior work studies stable online learning in unbounded state spaces and Lyapunov-based PAC analysis for finite weakly-coupled systems (Shah et al., 2020; Wu et al., 2026). These results do not characterize the minimax sample complexity of learning a near-optimal average-reward policy over a prescribed Lyapunov-stable policy class. We provide matching upper and lower bounds for this problem in countable-state MDPs.

Reinforcement learning in unbounded and countable-state MDPs. Learning in unbounded MDPs has been studied by exploiting queueing dynamics, birth-and-death structure, parametric models, mixed dynamics, or structured policy optimization (Dai & Gluzman, 2022; Anselmi et al., 2022; Adler & Subramanian, 2023; Wei et al., 2023; Freund et al., 2023; Jali et al., 2024; Grosof et al., 2025; Murthy et al., 2024). The closest algorithmic precedent is Liu et al. (2022), who learn within a finite queue-length region and use a known stabilizing policy outside it, proving that the resulting policy approaches optimality as the region expands. Other results establish regret or optimization guarantees under birth-and-death, parametric, or policy-gradient structure (Anselmi et al., 2022; Adler & Subramanian, 2023; Grosof et al., 2025; Murthy et al., 2024). These results provide approximation, regret, or optimization rates, but do not characterize the finite-sample complexity of learning a near-

optimal policy satisfying a Lyapunov-drift constraint. We provide an explicit upper bound for this problem together with a matching lower bound.

Sample complexity of average-reward reinforcement learning. Finite-sample theory for finite-state average-reward MDPs gives guarantees in terms of the number of state-action pairs and quantities such as mixing time, bias span, or transient time (Jin & Sidford, 2020; 2021; Wang et al., 2022; Zhang & Xie, 2023; Jin et al., 2024; Wang et al., 2024; Zurek & Chen, 2024; 2025; Lee et al., 2025). Our countable-state analysis replaces the fixed state count by the growth of an accuracy-dependent Lyapunov core. The Lyapunov drift condition further controls return times to the reference state, so the analysis does not require separate mixing-time, span, or Bellman-contraction assumptions.

2 Problem Formulation

In this section, we first introduce the countable-state average-reward MDP model. Then, we define Lyapunov-drift stability and the stability-constrained learning problem.

2.1 Countable-State Average-Reward MDPs

We consider a countable-state average-reward Markov decision process $M = (\mathcal{S}, \mathcal{A}, P, r)$, where $\mathcal{S}$ is the countable state space, $\mathcal{A}$ is a finite action space, $P(\cdot \mid s, a)$ is the transition kernel, and $r(s, a) \in [0, 1]$ is the mean one-step reward. The transition kernel and reward distributions are unknown. We assume the agent has access to a generative model: for any state-action pair (s, a), querying the model returns a reward sample R and a next-state sample S', where $R \in [0, 1]$, $\mathbb{E}[R \mid s, a] = r(s, a)$, and $S' \sim P(\cdot \mid s, a)$.

For an initial-state distribution μ and a policy $\pi : \mathcal{S} \to \Delta(\mathcal{A})$, let $(S_t, A_t, R_t)_{t\geq 0}$ denote the trajectory generated in M with $S_0 \sim \mu$ and $A_t \sim \pi(\cdot \mid S_t)$. The average reward of policy π from μ is defined as

$$\rho_M^\mu(\pi) := \liminf_{T\to\infty} \frac{1}{T}\mathbb{E}_{M,\mu}^\pi\left[\sum_{t=0}^{T-1} R_t\right].$$

2.2 Lyapunov Stability and Stability-Constrained Learning

We first define Lyapunov-drift stability and the corresponding optimal stable benchmark, and then formulate the learning objective. Fix a target stability margin $\zeta \in (0, 1)$ and a reference state $s_\circ \in \mathcal{S}$. We use a known potential function $W : \mathcal{S} \to [1, \infty)$ as a scalar measure of state magnitude relative to $s_\circ$. Larger values of $W(s)$ indicate a greater deviation from $s_\circ$, and we normalize $W(s_\circ) = 1$.

Definition 2.1 (Lyapunov-drift stability)**.** A policy π is ζ-stable if

$$\mathbb{E}_M^\pi[W(S_{t+1}) \mid S_t = s] \leq (1 - \zeta)W(s), \quad \forall s \neq s_\circ.$$

This is a standard geometric Foster–Lyapunov drift condition (Foster, 1953; Meyn & Tweedie, 2009). More broadly, Lyapunov functions are classical tools for establishing stability in dynamical systems (Khalil, 2002; Sastry, 1999). Intuitively, at every non-reference state, a ζ-stable policy reduces the expected next-step potential by at least a ζ fraction of its current value.

We use the best ζ-stable policy as the performance benchmark. Fix a reference action $a_\circ$ at $s_\circ$, and let $\Pi_\zeta(M)$ be the set of ζ-stable policies that choose $a_\circ$ at $s_\circ$. As shown in Lemma C.3, $\rho_M^\mu(\pi)$ is the same for every initial-state distribution μ and every $\pi \in \Pi_\zeta(M)$. We denote this common average reward by $\rho_M(\pi)$ and define the optimal ζ-stable average reward as $\rho_\zeta^\star(M) := \sup_{\pi\in\Pi_\zeta(M)} \rho_M(\pi)$.

We assume that the agent has access to a known stabilizing policy $\pi_{\text{stab}} \in \Pi_\zeta(M)$. Related finite-region queueing RL and safe model-based control methods also begin from a known stabilizing or safe policy (Liu et al., 2022; Berkenkamp et al., 2017). Here, it provides stable actions outside the learned core, although it may be highly conservative in performance. The agent seeks to approach the optimal ζ-stable average reward while retaining the slightly weaker stability margin $(1 - \kappa)\zeta$.

Definition 2.2 (Stability-constrained near-optimal policy)**.** For $\varepsilon, \kappa \in (0, 1)$, a policy π is stability-constrained near-optimal for target margin ζ, with accuracy ε and stability slack κ, if

$$\pi \in \Pi_{(1-\kappa)\zeta}(M), \qquad \rho_M(\pi) \geq \rho_\zeta^\star(M) - \varepsilon.$$

3 Occupancy Concentration on Lyapunov Cores

This section explains why Lyapunov stability makes the countable-state problem effectively finite. We show that negative Lyapunov drift concentrates the long-run occupancy of stable policies within a Lyapunov core, which allows their global performance to be approximated from data in the core.

3.1 Lyapunov Cores

We introduce Lyapunov cores to describe where stable policies concentrate. Specifically, for any radius $R \geq 1$, define the core $\mathcal{K}_R := \{s \in \mathcal{S} : W(s) \leq R\}$. We assume that $\mathcal{K}_R$ is finite and enumerable, with $|\mathcal{K}_R| \leq c_W R^\nu$, where $c_W < \infty$ controls the size of the core and $\nu > 0$ describes how quickly it grows with R.

Assumption 3.1 (Bounded one-step movement)**.** *There exist constants $\alpha, \beta \geq 1$ such that, for every $s \neq s_\circ$, $a \in \mathcal{A}$, and $S' \sim P(\cdot \mid s, a)$, we have $W(S') \leq \alpha W(s)$ almost surely. At $s = s_\circ$, the expected potential after one transition satisfies $\mathbb{E}[W(S') \mid S_t = s_\circ, A_t = a_\circ] \leq \beta$.*

This assumption means that the system cannot move arbitrarily far in a single step. The factor α bounds the largest one-step expansion of the potential away from the reference state, while β controls the expected potential after a transition from the reference state. It is satisfied, for example, when arrivals, service, disturbances, or actuation are bounded within one decision interval.

3.2 Occupancy Concentration

We first quantify the occupancy of a ζ-stable policy. For $\pi \in \Pi_\zeta(M)$, let d_M^π denote its long-run state occupancy. By Lemma C.3, this occupancy exists, is independent of the initial distribution, and coincides with the unique stationary distribution induced by π. The following theorem bounds the long-run probability of being outside any Lyapunov core.

Theorem 3.1 (Occupancy concentration on Lyapunov cores)**.** *Under Assumption 3.1, every policy $\pi \in \Pi_\zeta(M)$ satisfies, for every $R \geq 1$,*

$$\Pr_{S \sim d_M^\pi}(S \in \mathcal{K}_R) \geq 1 - \frac{\beta}{\zeta R}.$$

It reveals that, under the long-run occupancy of any ζ-stable policy, the probability of being outside the radius-R core is at most $\beta/(\zeta R)$. Since rewards lie in $[0, 1]$, the reward contributions of states outside the core can be small when R is large. Meanwhile, a larger stability margin ζ permits a smaller effective learning region. The proof is deferred to Appendix C.

3.3 Finite-Core Approximation of Stable Policies

The optimal stable value can be approximated without learning a policy over the entire state space. It is enough to optimize the policy inside a Lyapunov core and use the known stabilizing policy outside it, as in Liu et al. (2022). The following proposition establishes the corresponding approximation guarantee for the Lyapunov-stable policy class considered here. For $\pi \in \Pi_\zeta(M)$, let π_R^{switch} denote the switching policy that follows π inside the core $\mathcal{K}_R$ and the stabilizing policy π_{stab} outside it.

Proposition 3.1 (Core approximation of stable policies)**.** *There exists a universal constant C such that, for every $\pi \in \Pi_\zeta(M)$ and $R \geq e$, the policy π_R^{switch} belongs to $\Pi_\zeta(M)$ and satisfies $\left|\rho_M(\pi) - \rho_M(\pi_R^{\text{switch}})\right| \leq C\beta(1 + \log R)/(\zeta R)$.*

The proposition shows that the approximation error vanishes as $O((1+\log R)/R)$. The proof is given in Appendix C.

4 Lyapunov-Constrained Learning Algorithm Design

The Lyapunov-core principle reduces the infinite problem to a finite region, but it does not reveal which in-core decisions are stable or how they affect rewards and transitions. We introduce LyCore (Lyapunov-Constrained Core Learning), which estimates Lyapunov drift uniformly, learns the

environment model with shell-dependent precision, and combines both estimates through a stability-constrained average-reward linear program. Algorithm 1 summarizes the procedure.

Algorithm 1 LyCore (Lyapunov-Constrained Core Learning)

Require: Accuracy ε, confidence δ, target margin ζ, slack κ, reference pair $(s_\circ, a_\circ)$, potential W, stabilizing policy π_{stab}, generative model, core enumeration, and structural bounds $(\alpha, \beta, c_W, \nu)$
Ensure: A policy using $\widehat{\pi}_R$ inside $\mathcal{K}_R$ and π_{stab} outside it
1: Set R, n_{stab}, $n_\circ$, and $n_{\text{mod}}(s)$ according to Appendix G
2: **for** $s \in \mathcal{K}_R \setminus \{s_\circ\}$ and $a \in \mathcal{A}$ **do**
3: Draw n_{stab} next-state samples and estimate the drift bound $\overline{m}(s,a)$
4: **end for**
5: **for** $s \in \mathcal{K}_R \setminus \{s_\circ\}$ and $a \in \mathcal{A}$ **do**
6: Independently draw $n_{\text{mod}}(s)$ reward–next-state samples and estimate $(\widehat{r}, \widehat{P})$
7: **end for**
8: Draw the prescribed model samples at $(s_\circ, a_\circ)$ and construct the empirical core MDP $\widehat{M}_R$ on $\mathcal{K}_R$
9: **if** the empirical drift-feasible set is empty at any core state **then**
10: **return** the stabilizing policy π_{stab}
11: **end if**
12: Solve the stability-constrained average-reward LP
13: Recover $\widehat{\pi}_R$ from the LP solution and **return** the policy using $\widehat{\pi}_R$ inside $\mathcal{K}_R$ and π_{stab} outside it

4.1 Lyapunov Drift Learning

The first step learns which decisions can be used inside the core without violating the Lyapunov-drift constraint under the unknown dynamics. For each $s \in \mathcal{K}_R \setminus \{s_\circ\}$ and $a \in \mathcal{A}$, the agent draws n_{stab} independent next-state samples $Z'_{s,a,i} \sim P(\cdot \mid s,a)$. The empirical drift estimate and its empirical-Bernstein upper confidence bound are

$$\widehat{\Delta}_W(s,a) := \frac{1}{n_{\text{stab}}} \sum_{i=1}^{n_{\text{stab}}} \frac{W(Z'_{s,a,i}) - W(s)}{W(s)},$$

$$\overline{\Delta}_W(s,a) := \min\left\{ \alpha - 1, \widehat{\Delta}_W(s,a) + \sqrt{\frac{2\widehat{v}_W(s,a)\Lambda}{n_{\text{stab}}}} + \frac{7\alpha\Lambda}{3(n_{\text{stab}}-1)} \right\},$$

Here, $\widehat{v}_W(s,a)$ is the empirical variance of the normalized drift observations, and Λ is the logarithmic confidence factor specified in Appendix G (Maurer & Pontil, 2009). For the planning step, define $\overline{m}(s,a) := 1 + \overline{\Delta}_W(s,a)$. The common sample count n_{stab} is chosen so that these bounds hold simultaneously over the core. Its exact value is given in Appendix G.

With high probability, the true normalized drift is at most $\overline{\Delta}_W(s,a)$ for every sampled state-action pair. Enforcing the planning constraint with $\overline{m}(s,a)$ therefore prevents sampling error from making an unstable decision appear stable.

4.2 Core Model Learning with Shell-Dependent Sampling

The second step estimates rewards and transitions inside the Lyapunov core.

Empirical model. For each $s \in \mathcal{K}_R \setminus \{s_\circ\}$ and $a \in \mathcal{A}$, the agent draws $n_{\text{mod}}(s)$ independent model samples $(R_{s,a,i}, S'_{s,a,i})$, where $S'_{s,a,i} \sim P(\cdot \mid s,a)$ and $\mathbb{E}[R_{s,a,i} \mid s,a] = r(s,a)$. These samples are independent of those used for drift estimation. At the reference state, the agent draws $n_\circ$ samples from $(s_\circ, a_\circ)$ and sets $n_{\text{mod}}(s_\circ) := n_\circ$. For each sampled state-action pair and $x \in \mathcal{K}_R$, define

$$\widehat{r}(s,a) := \frac{1}{n_{\text{mod}}(s)} \sum_{i=1}^{n_{\text{mod}}(s)} R_{s,a,i},$$

$$\widehat{P}(x \mid s,a) := \frac{1}{n_{\text{mod}}(s)} \sum_{i=1}^{n_{\text{mod}}(s)} \left[\mathbf{1}\{S'_{s,a,i} = x\} + \mathbf{1}\{S'_{s,a,i} \notin \mathcal{K}_R\} \mathbf{1}\{x = s_\circ\} \right].$$

To obtain a finite planning problem, we map every sampled transition leaving the core to the reference state $s_\circ$. This defines a finite surrogate MDP on $\mathcal{K}_R$ for computing the in-core policy. The mapping is used only for planning. In the original MDP, the state evolves normally outside the core, where the output policy follows π_{stab} until the process returns to the core.

Shell-dependent sampling. Uniform model sampling would estimate outer states as accurately as central states, even though stability makes their influence on average reward progressively smaller. Let $r_j = 2^j$ and group states with comparable potential into shells $\mathcal{B}_j := \{s \in \mathcal{K}_R \setminus \{s_\circ\} : r_j < W(s) \le 2r_j\}$. By Theorem 3.1, outer shells have progressively less influence on average reward. Balancing this influence with model-estimation error gives the leading allocation $n_{\text{mod}}(s) \propto r_j^{-2}$ for $s \in \mathcal{B}_j$. Since $\mathcal{B}_j$ contains at most $O(r_j^\nu)$ states, its total sampling cost scales as $r_j^{\nu-2}$, which gives the phase transition at $\nu = 2$. The complete allocation and analysis are given in Appendix G and Theorem F.2.

4.3 Stability-Constrained Average-Reward Optimization

The final step optimizes reward and enforces stability simultaneously. Optimizing the empirical reward model alone may select high-performing actions that do not satisfy the drift constraint. Let $x(s,a)$ denote the stationary state-action occupancy measure in the empirical core MDP. The optimal occupancy measure is obtained by solving the following finite linear program:

$$\begin{aligned}
\max_{x \ge 0} \quad & \sum_{s\in\mathcal{K}_R}\sum_{a\in\mathcal{A}} x(s,a)\widehat{r}(s,a) \\
\text{s.t.} \quad & \sum_{a\in\mathcal{A}} x(y,a) = \sum_{s\in\mathcal{K}_R}\sum_{a\in\mathcal{A}} x(s,a)\widehat{P}(y \mid s,a), && \forall y \in \mathcal{K}_R, \\
& \sum_{s\in\mathcal{K}_R}\sum_{a\in\mathcal{A}} x(s,a) = 1, \qquad x(s_\circ,a) = 0, && \forall a \ne a_\circ, \\
& \sum_{a\in\mathcal{A}} x(s,a)\overline{m}(s,a) \le \big[1-(1-\kappa/2)\zeta\big]\sum_{a\in\mathcal{A}} x(s,a), && \forall s \in \mathcal{K}_R \setminus \{s_\circ\}.
\end{aligned} \tag{1}$$

The flow and normalization constraints make x a stationary state-action occupancy measure, while the last constraint enforces the learned Lyapunov-drift condition inside the core. Let $\widehat{x}^\star$ be an optimal solution. The in-core policy is recovered as[1]

$$\widehat{\pi}_R(a \mid s) := \begin{cases} \dfrac{\widehat{x}^\star(s,a)}{\sum_{a'\in\mathcal{A}}\widehat{x}^\star(s,a')}, & \sum_{a'\in\mathcal{A}}\widehat{x}^\star(s,a') > 0, \\ \widehat{p}_s(a), & \sum_{a'\in\mathcal{A}}\widehat{x}^\star(s,a') = 0. \end{cases}$$

The final switching policy on the original MDP is

$$\big(T_R^{\text{stab}}\widehat{\pi}_R\big)(a \mid s) := \begin{cases} \widehat{\pi}_R(a \mid s), & s \in \mathcal{K}_R, \\ \pi_{\text{stab}}(a \mid s), & s \notin \mathcal{K}_R. \end{cases}$$

From finite optimization to a global policy. The occupation-measure LP computes an in-core policy on a finite surrogate MDP (Puterman, 1994; Altman, 1999). The final policy uses this learned policy inside the Lyapunov core and π_{stab} outside it. Transitions leaving the core are redirected to $s_\circ$ only in the surrogate model used for optimization. The true system is never reset. Proposition 3.1 bounds the resulting performance loss. With high probability, the learned drift constraints guarantee stability inside the core, while π_{stab} guarantees stability outside it.

Regeneration replaces contraction. Interestingly, Lyapunov stability provides a different way to control average-reward learning. Average-reward Bellman operators are not generally contractive (Puterman, 1994). In our setting, negative Lyapunov drift repeatedly brings the process back to the reference state. By Lemma C.1, the expected return time from any state in the radius-R core is at most $O((1+\log R)/\zeta)$. Each return therefore ends a regenerative cycle, reducing the continuing average-reward problem to a sequence of random-horizon episodes with controlled expected length. This regenerative structure bounds how model and Bellman errors accumulate without requiring separate assumptions on mixing time, diameter, bias span, or Bellman contraction (Meyn & Tweedie, 2009).

[1] At a zero-occupancy state, $\widehat{p}_s$ is a drift-feasible Bellman-optimal mixture. At $s_\circ$, it selects $a_\circ$. If the LP is infeasible, LyCore returns π_{stab}.

5 Theoretical Guarantees

In this section, we establish upper and minimax lower bounds on the sample complexity of stability-constrained learning, matching up to logarithmic factors.

5.1 Performance Upper Bound

Theorem 5.1 (Sample complexity of LyCore). *Fix $\nu > 0$ and consider the setting of Section 2 under Assumption 3.1 and the core-growth condition in Section 3. Suppose $|\mathcal{A}| \geq 2$, $\beta \geq 1$, and $\alpha \geq 1$. For any $\zeta, \kappa, \varepsilon, \delta \in (0,1)$, LyCore with radius $R = \widetilde{O}_\nu(\beta/(\zeta\varepsilon))$ returns, with probability at least $1-\delta$, a policy $T_R^{\mathrm{stab}}\widehat{\pi}_R$ that is $(1-\kappa)\zeta$-stable and satisfies $\rho_M(T_R^{\mathrm{stab}}\widehat{\pi}_R) \geq \rho_\zeta^\star(M) - \varepsilon$. Its number of generative samples is*[2]

$$N_{\mathrm{up}} = \widetilde{O}_\nu\left[|\mathcal{A}|\, c_W \beta^\nu \left(\frac{1}{\zeta^{\nu+1}\varepsilon^{\max\{2,\nu\}}} + \frac{\alpha - 1 + \zeta}{\kappa^2 \zeta^{\nu+2}\varepsilon^\nu}\right)\right]. \tag{2}$$

Performance learning. The first term $\widetilde{O}_\nu(|\mathcal{A}|c_W\beta^\nu/[\zeta^{\nu+1}\varepsilon^{\max\{2,\nu\}}])$ is the cost of learning rewards and transitions well enough to optimize average reward. The core-growth exponent ν acts as an effective state-space dimension: when $\nu \leq 2$, the usual ε^{-2} estimation cost dominates, whereas when $\nu > 2$, the expanding core determines the $\varepsilon^{-\nu}$ rate. The factor $\zeta^{-(\nu+1)}$ captures the statistical value of stability: stronger drift concentrates long-run behavior within a smaller region and therefore significantly reduces the samples required for performance learning. Note that, if we assign the usual ε^{-2} precision uniformly across an $\varepsilon^{-\nu}$-scale core, it would produce the naive rate $\varepsilon^{-(\nu+2)}$. Our shell-dependent precision avoids this multiplication and attains $\varepsilon^{-\max\{2,\nu\}}$ without knowing the optimal policy's long-run occupancy.

Stability learning. The second term is therefore the cost of learning the drift constraint. This cost increases when the allowed stability slack κ is smaller or when the one-step movement bound α is larger. A larger stability margin ζ makes learning easier by concentrating the process more strongly, while the reward accuracy ε affects this term through the number of core states that must be learned. Thus, performance learning dominates for $\nu < 2$ at high accuracy, whereas stability learning dominates for $\nu \geq 2$ or when drift estimation is particularly demanding.

Which cost dominates. Ignoring the logarithmic factors hidden in $\widetilde{O}_\nu$ and writing $x_+ := \max\{x, 0\}$, define the ratio of the stability-learning cost to the performance-learning cost in (2) as

$$\mathcal{R}_{\mathrm{stab/perf}} := \frac{\alpha - 1 + \zeta}{\kappa^2 \zeta}\varepsilon^{(2-\nu)_+}.$$

When $\nu < 2$, the ratio converges to zero as ε decreases, so performance learning is the asymptotic bottleneck at high accuracy. When $\nu = 2$, both terms scale as ε^{-2}, but the stability-learning term is larger by the factor $(\alpha - 1 + \zeta)/(\kappa^2\zeta) \geq 1/\kappa^2 \geq 1$, so stability learning dominates. When $\nu > 2$, both terms scale as $\varepsilon^{-\nu}$, and the same additional factor again makes stability learning dominant. Even when $\nu < 2$, stability learning can dominate at moderate accuracy if the allowed slack κ is very small or the movement factor α is very large. Increasing the stability margin ζ reduces both costs by concentrating long-run occupancy within a smaller core.

Relation to existing RL theory. For finite-state average-reward MDPs, the optimal generative-model complexity is $\widetilde{O}(|\mathcal{S}||\mathcal{A}|H_{\mathrm{span}}/\varepsilon^2)$, where H_{span} is the optimal bias span. General multichain MDPs additionally require a transient-time parameter (Zurek & Chen, 2024). Our theorem replaces the fixed state count $|\mathcal{S}|$ with an accuracy-dependent Lyapunov core. Perhaps surprisingly, it requires no separate mixing-time, diameter, bias-span, or contraction assumption. Average-reward Bellman operators are not generally contractive (Puterman, 1994), while negative Lyapunov drift controls return times to the reference state and thereby bounds the long-run sensitivity of average reward (Meyn & Tweedie, 2009). When $\nu \leq 2$, the performance-learning term retains the standard ε^{-2} accuracy dependence despite the countable state space. Countable-state queueing results obtain approximation,

[2] $\widetilde{O}_\nu$ and $\widetilde{\Omega}_\nu$ hide logarithmic factors and constants depending only on ν.

regret, or policy-optimization guarantees by exploiting queue-specific dynamics or structured policy classes (Liu et al., 2022; Anselmi et al., 2022; Grosof et al., 2025; Murthy et al., 2024). In contrast, our result gives an end-to-end generative-model PAC characterization for a returned policy that satisfies the prescribed Lyapunov-drift condition under the unknown dynamics. The Lyapunov-based PAC analysis of Wu et al. (2026) instead uses a constructed Lyapunov function as an analysis tool for finite weakly coupled MDPs.

5.2 Matching Lower Bound

Theorem 5.2 (Minimax lower bound)**.** *Fix $\nu > 0$. There are constants $c_\nu, C_\nu > 0$ such that the following holds. Consider any parameters $|\mathcal{A}| \geq 3$, $c_W \geq 1$, $\beta \geq 2$, $\alpha \in [1, 2]$, $\kappa \in (0, 1/2]$, and $0 < \zeta \leq 1/64$ satisfying $c_W(\beta/\zeta)^\nu \geq C_\nu$. For all sufficiently small ε, there exists a family of MDPs satisfying Assumption 3.1, the stabilizing-policy input of Section 2.2, and the core conditions of Section 3 such that any algorithm that, with probability at least $2/3$, returns a policy $\widehat{\pi} \in \Pi_{(1-\kappa)\zeta}(M)$ satisfying $\rho_M(\widehat{\pi}) \geq \rho^\star_\zeta(M) - \varepsilon$ must obey the following bound, where τ is its total number of generative-model calls:*

$$\sup_M \mathbb{E}_M[\tau] \geq \widetilde{\Omega}_\nu\left[|\mathcal{A}|\, c_W \beta^\nu \left(\frac{1}{\zeta^{\nu+1}\varepsilon^{\max\{2,\nu\}}} + \frac{\alpha - 1 + \zeta}{\kappa^2 \zeta^{\nu+2}\varepsilon^\nu}\right)\right].$$

For every fixed ν in the stated parameter regime, Theorems 5.1 and 5.2 match in all displayed polynomial factors, up to logarithmic terms. Together, the bounds identify performance learning and stability learning as the two intrinsic components of the problem.

The proof uses two complementary hard families. In the first, the agent must identify good decisions across many core states, which yields the performance-learning term. In the second, actions are difficult to distinguish by their effect on the potential, forcing the agent to estimate drift accurately enough to guarantee stability. This yields the stability-learning term, including its κ^{-2} dependence. The complete construction and proof are given in Appendix H.

6 Experiments

We use a controlled multi-shell MDP to test the predicted sample-complexity rates and a nonlinear IEEE 33-bus feeder task to evaluate stability-constrained learning under uncertain physical dynamics. Full protocols and additional experiments appear in Appendix J.

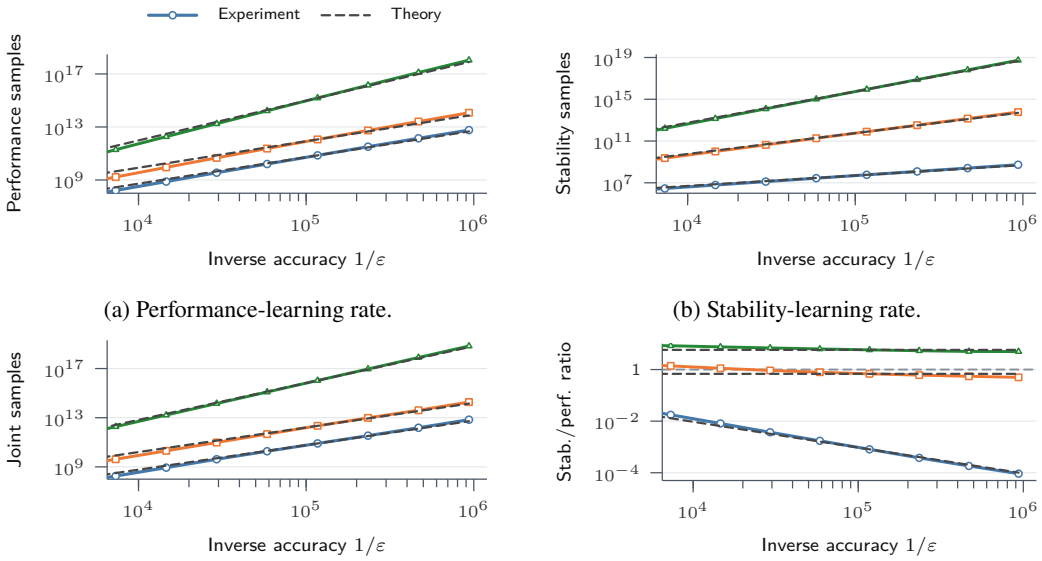
6.1 Controlled Multi-Shell Rate Experiment

A reference state launches the process into dyadic potential shells whose states either self-loop at their level or return to the reference state. We vary the core-growth exponent ν and target accuracy ε, and empirically measure the total samples required to attain 90% success in the performance, stability, and joint learning tasks.

Scaling results. Fitting the four smallest-ε points gives slopes $(2.11, 2.23, 3.16)$ for performance learning, $(1.07, 2.08, 3.09)$ for stability learning, and $(2.14, 2.13, 3.10)$ for joint learning at $\nu = 1, 2, 3$. These approach the respective polynomial exponents $(2, 2, 3)$, $(1, 2, 3)$, and $(2, 2, 3)$ predicted by Theorem 5.1. The remaining deviations are consistent with its suppressed logarithmic factors. At the finest accuracy, the stability-to-performance sample ratios are 9.2×10^{-5}, 0.50, and 4.97, recovering the predicted change in the dominant learning cost.

6.2 IEEE 33-Bus Feeder Voltage Control

We next evaluate LyCore on the widely used IEEE 33-bus feeder, specifically the Baran–Wu system distributed as MATPOWER `case33bw`, with nonlinear AC power flow and uncertain line, load, solar, and inverter dynamics (Baran & Wu, 1989; MATPOWER, 2016). All learned methods use the same total sample budget. Because voltage regulation targets an operating band, we use the reference-set drift guarantee in Proposition I.1.



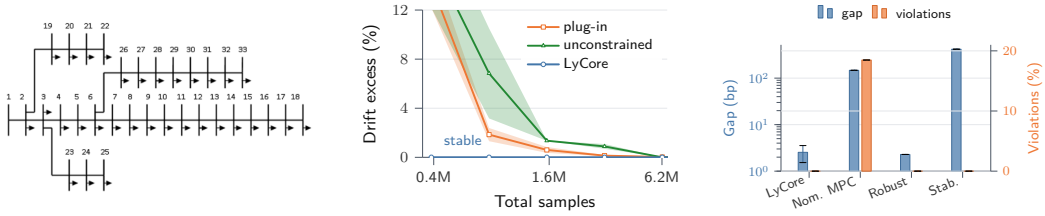
(a) Performance-learning rate.

(b) Stability-learning rate.

(c) Joint learning rate.

(d) Relative learning cost.

Figure 1: Multi-shell sample-complexity scaling at small ε (log–log axes). Curves show empirical means and 95% confidence intervals. Dashed lines are predicted-exponent slopes anchored at the first fitted point. Panels (a)–(c) give samples for 90% success in performance, stability, and joint learning. Panel (d) gives their ratio. Markers denote $\nu = 1, 2, 3$.



(a) IEEE 33-bus feeder.

(b) Stability learning.

(c) Performance–safety tradeoff.

Figure 2: IEEE 33-bus feeder full-AC voltage control. (a) Topology. (b) True-model drift excess versus total samples. (c) Aggregate-MDP optimality gap and stationary voltage violations at 199.07 million samples. Shading and error bars show one standard error over ten replicates. Deterministic baselines have zero error bars.

Results. LyCore has no true-model drift violation at any tested budget, whereas plug-in and unconstrained planning violate the drift constraint at small budgets. At the largest budget, LyCore attains a 2.5-basis-point optimality gap and zero stationary voltage violations. Under joint uncertainty, nominal AC-MPC has a 147.2-basis-point gap and an 18.47% violation rate, robust AC-OPF has a 2.3-basis-point gap and zero violations, and the stabilizing policy eliminates violations but incurs a 418.8-basis-point gap. The complete comparison in Appendix J also reports empirical AC-MPC, which attains a 0.8-basis-point gap and zero violations but does not provide a drift guarantee.

7 Conclusion

We studied stability-constrained reinforcement learning in MDPs with countably many states. Negative Lyapunov drift concentrates long-run occupancy within a finite core, enabling a finite-core algorithm with global stability and near-optimality guarantees. Matching upper and lower bounds characterize the separate costs of learning performance and stability. Extending the framework to online data collection and learning the stabilizing policy remains an important direction.

AI Use Statement

Generative AI tools were used to assist with literature search, experimental code, mathematical checking and revision, LaTeX editing, and prose revision. The authors are responsible for reviewing all AI-assisted text, mathematical statements, proofs, and code before submission.

References

Joshua Achiam, David Held, Aviv Tamar, and Pieter Abbeel. Constrained policy optimization. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 22–31. PMLR, 2017. URL https://proceedings.mlr.press/v70/achiam17a.html.

Saghar Adler and Vijay G. Subramanian. Bayesian learning of optimal policies in markov decision processes with countably infinite state-space. In *Advances in Neural Information Processing Systems*, volume 36, pp. 3979–4033, 2023. doi: 10.52202/075280-0175. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/0c79d6ed1788653643a1ac67b6ea32a7-Abstract-Conference.html.

Mridul Agarwal, Qinbo Bai, and Vaneet Aggarwal. Regret guarantees for model-based reinforcement learning with long-term average constraints. In *Proceedings of the Thirty-Eighth Conference on Uncertainty in Artificial Intelligence*, volume 180 of *Proceedings of Machine Learning Research*, pp. 22–31. PMLR, 2022. URL https://proceedings.mlr.press/v180/agarwal22b.html.

Eitan Altman. *Constrained Markov Decision Processes*. Chapman and Hall/CRC, 1999.

Jonatha Anselmi, Bruno Gaujal, and Louis-Sébastien Rebuffi. Reinforcement learning in a birth and death process: Breaking the dependence on the state space. In *Advances in Neural Information Processing Systems*, volume 35, 2022. doi: 10.52202/068431-1051. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/5d2781cc34f459618a9a504761043055-Abstract-Conference.html.

Mesut E. Baran and Felix F. Wu. Network reconfiguration in distribution systems for loss reduction and load balancing. *IEEE Transactions on Power Delivery*, 4(2):1401–1407, 1989. doi: 10.1109/61.25627.

Felix Berkenkamp, Matteo Turchetta, Angela P. Schoellig, and Andreas Krause. Safe model-based reinforcement learning with stability guarantees. In *Advances in Neural Information Processing Systems*, volume 30, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/766ebcd59621e305170616ba3d3dac32-Abstract.html.

Dimitri P. Bertsekas. *Dynamic Programming and Optimal Control*, volume 2. Athena Scientific, 2012.

Kiante Brantley, Miro Dudik, Thodoris Lykouris, Sobhan Miryoosefi, Max Simchowitz, Aleksandrs Slivkins, and Wen Sun. Constrained episodic reinforcement learning in concave-convex and knapsack settings. In *Advances in Neural Information Processing Systems*, volume 33, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/bc6d753857fe3dd4275dff707dedf329-Abstract.html.

Yinlam Chow, Ofir Nachum, Edgar Duenez-Guzman, and Mohammad Ghavamzadeh. A lyapunov-based approach to safe reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 31, 2018. URL https://arxiv.org/abs/1805.07708.

J. G. Dai and Mark Gluzman. Queueing network controls via deep reinforcement learning. *Stochastic Systems*, 12(1):30–67, 2022. doi: 10.1287/stsy.2021.0081. URL https://doi.org/10.1287/stsy.2021.0081.

Sarah Dean, Stephen Tu, Nikolai Matni, and Benjamin Recht. Safely learning to control the constrained linear quadratic regulator. In *Proceedings of the 2019 American Control Conference*, pp. 5582–5588. IEEE, 2019. URL https://arxiv.org/abs/1809.10121.

Dongsheng Ding, Xiaohan Wei, Zhuoran Yang, Zhaoran Wang, and Mihailo Jovanovic. Provably efficient safe exploration via primal-dual policy optimization. In *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pp. 3304–3312. PMLR, 2021. URL https://proceedings.mlr.press/v130/ding21d.html.

F. G. Foster. On the stochastic matrices associated with certain queuing processes. *The Annals of Mathematical Statistics*, 24(3):355–360, 1953. doi: 10.1214/aoms/1177728976.

Daniel Freund, Thodoris Lykouris, and Wentao Weng. Quantifying the cost of learning in queueing systems. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://papers.nips.cc/paper_files/paper/2023/hash/1502957929fc4257dd1b6daf7d869c2f-Abstract-Conference.html.

Anshul Gandhi, Mor Harchol-Balter, Rajarshi Das, and Charles Lefurgy. Optimal power allocation in server farms. In *Proceedings of the 11th International Joint Conference on Measurement and Modeling of Computer Systems*, pp. 157–168, 2009. doi: 10.1145/1555349.1555368.

Javier García and Fernando Fernández. A comprehensive survey on safe reinforcement learning. *Journal of Machine Learning Research*, 16(42):1437–1480, 2015. URL https://jmlr.org/papers/v16/garcia15a.html.

Isaac Grosof, Siva Theja Maguluri, and R. Srikant. Convergence of natural policy gradient for a family of infinite-state queueing MDPs. *Queueing Systems*, 109:22, 2025. doi: 10.1007/s11134-025-09949-y. URL https://arxiv.org/abs/2402.05274.

Minghao Han, Lixian Zhang, Jun Wang, and Wei Pan. Actor–critic reinforcement learning for control with stability guarantee. *IEEE Robotics and Automation Letters*, 5(4):6217–6224, 2020. doi: 10.1109/LRA.2020.3011351. URL https://doi.org/10.1109/LRA.2020.3011351.

Minghao Han, Yuan Tian, Lixian Zhang, Jun Wang, and Wei Pan. Reinforcement learning control of constrained dynamic systems with uniformly ultimate boundedness stability guarantee. *Automatica*, 129:109689, 2021. doi: 10.1016/j.automatica.2021.109689. URL https://doi.org/10.1016/j.automatica.2021.109689.

Minghao Han, Lixian Zhang, Chenliang Liu, Zhipeng Zhou, Jun Wang, and Wei Pan. Reinforcement learning for control with probabilistic stability guarantee: A finite-sample approach. *Automatica*, 188:112964, 2026. doi: 10.1016/j.automatica.2026.112964. URL https://doi.org/10.1016/j.automatica.2026.112964.

Neharika Jali, Guannan Qu, Weina Wang, and Gauri Joshi. Efficient reinforcement learning for routing jobs in heterogeneous queueing systems. In *Proceedings of the 27th International Conference on Artificial Intelligence and Statistics*, volume 238 of *Proceedings of Machine Learning Research*, pp. 4177–4185. PMLR, 2024. URL https://proceedings.mlr.press/v238/jali24a.html.

Ming Jin and Javad Lavaei. Stability-certified reinforcement learning: A control-theoretic perspective. *IEEE Access*, 8:229086–229100, 2020. doi: 10.1109/ACCESS.2020.3045114. URL https://doi.org/10.1109/ACCESS.2020.3045114.

Ying Jin, Ramki Gummadi, Zhengyuan Zhou, and Jose Blanchet. Feasible Q-learning for average reward reinforcement learning. In *Proceedings of the 27th International Conference on Artificial Intelligence and Statistics*, volume 238 of *Proceedings of Machine Learning Research*, pp. 1630–1638. PMLR, 2024.

Yujia Jin and Aaron Sidford. Efficiently solving MDPs with stochastic mirror descent. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 4890–4900. PMLR, 2020. URL https://proceedings.mlr.press/v119/jin20f.html.

Yujia Jin and Aaron Sidford. Towards tight bounds on the sample complexity of average-reward MDPs. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 5055–5064. PMLR, 2021. URL https://proceedings.mlr.press/v139/jin21b.html.

Hassan K. Khalil. *Nonlinear Systems*. Prentice Hall, 3 edition, 2002.

Sahin Lale, Kamyar Azizzadenesheli, Babak Hassibi, and Animashree Anandkumar. Reinforcement learning with fast stabilization in linear dynamical systems. In *Proceedings of the 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pp. 5354–5390. PMLR, 2022. URL https://proceedings.mlr.press/v151/lale22a.html.

Sahin Lale, Yuanyuan Shi, Guannan Qu, Kamyar Azizzadenesheli, Adam Wierman, and Anima Anandkumar. KCRL: Krasovskii-constrained reinforcement learning with guaranteed stability in nonlinear discrete-time systems. In *2023 62nd IEEE Conference on Decision and Control*, pp. 1334–1341, 2023. doi: 10.1109/CDC49753.2023.10384011.

Jongmin Lee, Mario Bravo, and Roberto Cominetti. Near-optimal sample complexity for MDPs via anchoring. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 32907–32929. PMLR, 2025.

Bai Liu, Qiaomin Xie, and Eytan Modiano. RL-QN: A reinforcement learning framework for optimal control of queueing systems. *ACM Transactions on Modeling and Performance Evaluation of Computing Systems*, 7(1):1–35, 2022. doi: 10.1145/3529375. URL https://doi.org/10.1145/3529375.

Shie Mannor and John N. Tsitsiklis. The sample complexity of exploration in the multi-armed bandit problem. *Journal of Machine Learning Research*, 5:623–648, 2004. URL https://www.jmlr.org/papers/v5/mannor04b.html.

MATPOWER. CASE33BW: Power flow data for 33 bus distribution system from Baran and Wu. https://matpower.org/docs/ref/matpower6.0/case33bw.html, 2016. Accessed 2026-08-27.

Andreas Maurer and Massimiliano Pontil. Empirical bernstein bounds and sample variance penalization. In *Proceedings of the 22nd Annual Conference on Learning Theory*, pp. 195–207, 2009. URL https://arxiv.org/abs/0907.3740.

Sean P. Meyn and Richard L. Tweedie. *Markov Chains and Stochastic Stability*. Cambridge University Press, 2 edition, 2009.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves, Martin Riedmiller, Andreas K. Fidjeland, Georg Ostrovski, Stig Petersen, Charles Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dharshan Kumaran, Daan Wierstra, Shane Legg, and Demis Hassabis. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015. doi: 10.1038/nature14236.

Yashaswini Murthy, Isaac Grosof, Siva Theja Maguluri, and R. Srikant. Performance of NPG in countable state-space average-cost RL. *arXiv preprint arXiv:2405.20467*, 2024. URL https://arxiv.org/abs/2405.20467.

Open-Meteo. Open-Meteo Historical Weather API. https://open-meteo.com/en/docs/historical-weather-api, 2026. Accessed 2026-08-24.

Brahma S. Pavse, Matthew Zurek, Yudong Chen, Qiaomin Xie, and Josiah P. Hanna. Learning to stabilize online reinforcement learning in unbounded state spaces. In *Proceedings of the Forty-First International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 40015–40033. PMLR, 2024. URL https://proceedings.mlr.press/v235/pavse24a.html.

Theodore J. Perkins and Andrew G. Barto. Lyapunov design for safe reinforcement learning. *Journal of Machine Learning Research*, 3:803–832, 2002. URL https://www.jmlr.org/papers/v3/perkins02a.html.

Martin L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Wiley, 1994.

Shankar Sastry. *Nonlinear Systems: Analysis, Stability, and Control*. Springer, 1999.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Devavrat Shah, Qiaomin Xie, and Zhi Xu. Stable reinforcement learning with unbounded state space. In *Proceedings of the 2nd Conference on Learning for Dynamics and Control*, volume 120 of *Proceedings of Machine Learning Research*, pp. 581–581. PMLR, 2020. URL https://proceedings.mlr.press/v120/shah20a.html.

David Silver, Aja Huang, Chris J. Maddison, Arthur Guez, Laurent Sifre, George van den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, Sander Dieleman, Dominik Grewe, John Nham, Nal Kalchbrenner, Ilya Sutskever, Timothy Lillicrap, Madeleine Leach, Koray Kavukcuoglu, Thore Graepel, and Demis Hassabis. Mastering the game of Go with deep neural networks and tree search. *Nature*, 529(7587):484–489, 2016. doi: 10.1038/nature16961.

Alexander L. Stolyar. MaxWeight scheduling in a generalized switch: State space collapse and workload minimization in heavy traffic. *The Annals of Applied Probability*, 14(1):1–53, 2004. doi: 10.1214/aoap/1075828046.

Leandros Tassiulas and Anthony Ephremides. Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks. *IEEE Transactions on Automatic Control*, 37(12):1936–1948, 1992. doi: 10.1109/9.182479.

Stephen Tu, Alexander Robey, Tingnan Zhang, and Nikolai Matni. On the sample complexity of stability constrained imitation learning. In *Proceedings of the 4th Annual Learning for Dynamics and Control Conference*, volume 168 of *Proceedings of Machine Learning Research*, pp. 180–191. PMLR, 2022. URL https://proceedings.mlr.press/v168/tu22a.html.

Sharan Vaswani, Lin F. Yang, and Csaba Szepesvari. Near-optimal sample complexity bounds for constrained MDPs. In *Advances in Neural Information Processing Systems*, volume 35, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/14a5ebc9cd2e507cd811df78c15bf5d7-Abstract-Conference.html.

Jinghan Wang, Mengdi Wang, and Lin F. Yang. Near sample-optimal reduction-based policy learning for average reward MDP. *arXiv preprint arXiv:2212.00603*, 2022. URL https://arxiv.org/abs/2212.00603.

Shengbo Wang, Jose Blanchet, and Peter Glynn. Optimal sample complexity for average reward markov decision processes. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/7fde8314b66c570c210f30cbd6c1735d-Abstract-Conference.html.

Honghao Wei, Xin Liu, Weina Wang, and Lei Ying. Sample efficient reinforcement learning in mixed systems through augmented samples and its applications to queueing networks. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/0663a39baab211328fc865f91abc75ab-Abstract-Conference.html.

Tyler Westenbroek, Fernando Castaneda, Ayush Agrawal, Shankar Sastry, and Koushil Sreenath. Lyapunov design for robust and efficient robotic reinforcement learning. In *Proceedings of The 6th Conference on Robot Learning*, volume 205 of *Proceedings of Machine Learning Research*, pp. 2125–2135. PMLR, 2023. URL https://proceedings.mlr.press/v205/westenbroek23a.html.

Adam Wierman, Lachlan L. H. Andrew, and Ao Tang. Power-aware speed scaling in processor sharing systems. In *IEEE INFOCOM 2009*, pp. 2007–2015, 2009. doi: 10.1109/INFCOM.2009.5062123.

Tianhao Wu, Matthew Zurek, Weina Wang, and Qiaomin Xie. Lyapunov-based sample complexity analysis for weakly-coupled MDPs (extended abstract). In *Proceedings of the Thirty-Ninth Conference on Learning Theory*, volume 336 of *Proceedings of Machine Learning Research*, pp. 6852–6857. PMLR, 2026. URL https://proceedings.mlr.press/v336/tianhao26a.html.

Songyuan Zhang and Chuchu Fan. Learning to stabilize high-dimensional unknown systems using Lyapunov-guided exploration. In *Proceedings of the 6th Annual Learning for Dynamics and Control Conference*, volume 242 of *Proceedings of Machine Learning Research*, pp. 52–67. PMLR, 2024. URL https://proceedings.mlr.press/v242/zhang24a.html.

Zihan Zhang and Qiaomin Xie. Sharper model-free reinforcement learning for average-reward markov decision processes. In *Proceedings of the 36th Conference on Learning Theory*, volume 195 of *Proceedings of Machine Learning Research*, pp. 5476–5477. PMLR, 2023.

Matthew Zurek and Yudong Chen. Span-based optimal sample complexity for weakly communicating and general average reward MDPs. In *Advances in Neural Information Processing Systems*, 2024.

Matthew Zurek and Yudong Chen. The plug-in approach for average-reward and discounted MDPs: Optimal sample complexity analysis. In Gautam Kamath and Po-Ling Loh (eds.), *Proceedings of The 36th International Conference on Algorithmic Learning Theory*, volume 272 of *Proceedings of Machine Learning Research*, pp. 1386–1387. PMLR, 2025.

A Appendix Roadmap

The appendix is organized around the two main theoretical guarantees. Appendix B gives the complete proof of the upper bound. It first proves stability of the returned policy and then separates its performance loss into finite-core approximation, projected-model transfer, and finite-core learning errors. The supporting arguments appear next, in the order in which they are used: Appendix C proves occupancy concentration and finite-core approximation, Appendix D proves the drift guarantees, Appendix E establishes the constrained average-reward planning result, and Appendices F and G prove the shell-weighted evaluation and sample counts. Appendix H then proves the matching lower bound. The final sections contain extensions, modeling examples, and experimental details.

Readers interested mainly in the proof logic may read Appendices B and H. The intervening sections provide the technical results used by those two proofs.

B Proof of Theorem 5.1

The theorem has two conclusions. We first prove that the returned policy is globally stable under the true dynamics. We then prove that its average reward is within ε of the optimal ζ-stable value. The proof has five steps, and the supporting results are proved in the sections that follow.

Throughout the upper-bound proof, let $K := \mathcal{K}_R$, let Γ_R map every next state outside K to $s_\circ$, let M_R be the resulting projected core MDP, and let $\widehat{M}_R$ be its empirical model. The policy $\widehat{\pi}_R$ is the empirical optimizer on K. We compare it with a fixed ζ-stable policy $\pi^\dagger$ and write $\pi^\dagger_R$ for its restriction to K.

Step 1: Choose the core and display the performance decomposition.

Set $\bar{\eta}_\kappa := (1 - \kappa/2)\zeta$ and

$$R := \left\lceil C_R \frac{\beta}{\bar{\eta}_\kappa \varepsilon} \left(1 + \log \frac{e\beta}{\bar{\eta}_\kappa \varepsilon}\right)\right\rceil, \tag{3}$$

with C_R sufficiently large. Since $\bar{\eta}_\kappa \geq \zeta/2$, this gives $R = \widetilde{O}(\beta/(\zeta\varepsilon))$. Choose a comparator $\pi^\dagger \in \Pi_\zeta(M)$ satisfying $\rho_M(\pi^\dagger) \geq \rho^\star_\zeta(M) - \varepsilon/16$, which exists by the definition of the supremum. Let $\pi^\dagger_R$ be its restriction to $\mathcal{K}_R$, and let $T^{\mathrm{stab}}_R \pi^\dagger$ be the corresponding switching policy.

For any in-core policy π_R, let $T^{\mathrm{stab}}_R \pi_R$ denote the policy that uses π_R on K and the known stabilizing policy outside K. In particular, the algorithm returns $\widetilde{\pi}^{\mathrm{stab}}_R := T^{\mathrm{stab}}_R \widehat{\pi}_R$.

The performance proof controls three errors. The first approximates the comparator on the finite core, the second transfers between the projected and original MDPs, and the third learns the best policy in

the projected core:

$$\rho_\zeta^\star(M) - \rho_M(\widetilde{\pi}_R^{\mathrm{stab}}) \le \underbrace{\frac{\varepsilon}{16} + \rho_M(\pi^\dagger) - \rho_M(T_R^{\mathrm{stab}}\pi^\dagger)}_{\mathcal{E}_{\mathrm{core}}} + \underbrace{\rho_M(T_R^{\mathrm{stab}}\pi^\dagger) - \rho_{M_R}(\pi_R^\dagger) + \rho_{M_R}(\widehat{\pi}_R) - \rho_M(\widetilde{\pi}_R^{\mathrm{stab}})}_{\mathcal{E}_{\mathrm{transfer}}} + \underbrace{\rho_{M_R}(\pi_R^\dagger) - \rho_{M_R}(\widehat{\pi}_R)}_{\mathcal{E}_{\mathrm{learn}}}. \tag{4}$$

Steps 3 and 4 bound these three terms.

Step 2: Prove global stability.

Set $\eta_{\mathrm{plan}} := (1 - 3\kappa/4)\zeta$ and $\eta_\kappa := (1-\kappa)\zeta$. On the drift event $\mathcal{E}_{\mathrm{drift}}$, every empirically feasible action mixture is truly $\bar{\eta}_\kappa$-stable, and every true ζ-stable comparator remains feasible. Thus $\pi_R^\dagger$ is retained and $\widehat{\pi}_R$ is stable on the core. Outside the core, the output follows the known ζ-stable policy. Therefore the switching policy $\widetilde{\pi}_R^{\mathrm{stab}} := T_R^{\mathrm{stab}}\widehat{\pi}_R$ is globally η_κ-stable. Together with the independent model event $\mathcal{E}_{\mathrm{model}}$, the same drift control gives margin η_{plan} in the empirical projected MDP, which makes the planning problem well posed. These claims are proved in Appendix D.

Step 3: Bound the finite-core and transfer errors.

The finite-core results in Appendix C apply without restricting the number of possible next states. For the radius in (3), Lemma C.4 bounds the policy-approximation term in $\mathcal{E}_{\mathrm{core}}$ and both terms in $\mathcal{E}_{\mathrm{transfer}}$. Increasing C_R if necessary ensures

$$C\frac{\beta(1+\log R)}{R}\left(\frac{1}{\bar{\eta}_\kappa} + \frac{1}{\zeta}\right) \le \frac{\varepsilon}{8}.$$

Step 4: Bound the finite-core learning error.

Set $\varepsilon_{\mathrm{pl}} := \varepsilon/16$. On the Bellman event $\mathcal{E}_{\mathrm{B}}$, the shell-weighted evaluation bound applies to both the retained comparator and the empirical optimizer. The plug-in comparison in Appendix G then gives $\mathcal{E}_{\mathrm{learn}} \le 2\varepsilon_{\mathrm{pl}} = \varepsilon/8$. The proof removes the dependence between the learned policy and each transition row, applies a support-free Bellman bound, and combines the errors across potential shells.

Step 5: Combine the bounds and count samples.

Let $\delta_{\mathrm{drift}} = \delta_{\mathrm{model}} = \delta_{\mathrm{B}} := \delta/4$ and $\mathcal{E} := \mathcal{E}_{\mathrm{drift}} \cap \mathcal{E}_{\mathrm{model}} \cap \mathcal{E}_{\mathrm{B}}$. The logarithmic choice in (47) gives $\mathbb{P}(\mathcal{E}) \ge 1 - \delta$. On this event, Step 2 proves global stability. Equation (4) and Steps 3–4 give

$$\rho_\zeta^\star(M) - \rho_M(\widetilde{\pi}_R^{\mathrm{stab}}) \le \frac{\varepsilon}{16} + \frac{\varepsilon}{8} + \frac{\varepsilon}{4} + \frac{\varepsilon}{8} < \varepsilon.$$

Finally, Lemmas G.1 and G.2 bound the independent model and drift batches by the two terms in (2). This proves Theorem 5.1. □

C Occupancy Concentration and Finite-Core Approximation

This section proves the structural facts used in Step 3 of the upper bound. Negative Lyapunov drift controls return times and the probability of leaving a Lyapunov core. Standard regenerative arguments then give occupancy concentration and the finite-core approximation bounds.

Remark C.1 (Reference-state separation)**.** Since $W \ge 1$, the drift condition implies $W(s) \ge 1/(1-\zeta) > 1$ for every $s \ne s_\circ$. Thus the singleton reference state is separated on the potential scale. Proposition I.1 gives the analogous guarantee for a reference set.

Lemma C.1 (Geometric-drift return moments)**.** *Let $(\mathcal{F}_t)_{t\ge 0}$ be the natural filtration of the controlled process, and suppose a possibly history-dependent controller satisfies $\mathbb{E}[W(S_{t+1}) \mid \mathcal{F}_t] \le (1-\eta)W(S_t)$ for some $\eta \in (0,1)$ whenever $S_t \neq s_\circ$. Let $\tau_\circ = \inf\{t \ge 0 : S_t = s_\circ\}$. There is a universal constant C such that, for every initial state x,*

$$\mathbb{P}_x(\tau_\circ > t) \le \min\{1, W(x)(1-\eta)^t\}, \qquad \mathbb{E}_x\tau_\circ \le C\frac{1+\log W(x)}{\eta}, \qquad \mathbb{E}_x\tau_\circ^2 \le C\frac{(1+\log W(x))^2}{\eta^2}.$$

Proof. For the process killed at $s_\circ$, induction gives $\mathbb{E}_x[W(S_t)\mathbf{1}\{\tau_\circ > t\}] \le (1-\eta)^t W(x)$. Since $W \ge 1$, this proves the tail bound. Summing the tail before and after $t_0 = \lceil \log W(x)/[-\log(1-\eta)]\rceil$ gives the first moment. Applying the same split to $\mathbb{E}\tau_\circ^2 = \sum_{t\ge 0}(2t+1)\mathbb{P}(\tau_\circ > t)$ gives the second. □

Lemma C.2 (Reference-state transition and weighted overshoot)**.** *Let a regenerative cycle start at $s_\circ$, take the fixed reference action, and end at $\tau_\circ^+ = \inf\{t \ge 1 : S_t = s_\circ\}$. Define $\sigma_R = \inf\{t \ge 1 : W(S_t) > R\}$ and $E_R = \{\sigma_R < \tau_\circ^+\}$. If the reference row satisfies $\mathbb{E}[W(S_1) \mid S_0 = s_\circ, A_0 = a_\circ] \le \beta$ and the controller used before $\sigma_R \wedge \tau_\circ^+$ has nonpositive W drift, then*

$$\mathbb{P}_{s_\circ}(E_R) \le \frac{\beta-1}{R-1}, \qquad \mathbb{E}_{s_\circ}[W(S_{\sigma_R})\mathbf{1}_{E_R}] \le \beta,$$
$$\mathbb{E}_{s_\circ}[(1+\log W(S_{\sigma_R}))\mathbf{1}_{E_R}] \le \frac{\beta(1+\log R)}{R}, \qquad R \ge e.$$

Proof. Condition on $S_1 = x$ and let $\tau_n := (\sigma_R \wedge \tau_\circ) \wedge n$. Optional stopping for the nonnegative supermartingale $W(S_t)$, followed by Fatou's lemma, gives

$$W(x) \ge \mathbb{E}_x\Big[\liminf_{n\to\infty} W(S_{\tau_n})\Big] \ge \mathbb{E}_x[W(S_{\sigma_R})\mathbf{1}_{\{\sigma_R<\tau_\circ\}}] + \mathbb{P}_x(\tau_\circ \le \sigma_R).$$

Since $W(S_{\sigma_R}) > R$ on the first event, $1 + (R-1)\mathbb{P}_x(\sigma_R < \tau_\circ) \le W(x)$. Averaging over $S_1 \sim P(\cdot \mid s_\circ, a_\circ)$ yields the tail-hit probability bound. Dropping the nonnegative return term gives the weighted overshoot bound. Finally, $w \mapsto (1+\log w)/w$ is decreasing for $w \ge 1$, so on E_R, $1 + \log W(S_{\sigma_R}) \le [(1+\log R)/R]W(S_{\sigma_R})$. Taking expectations completes the proof. □

Lemma C.3 (Renewal-reward representation)**.** *For any $\eta \in (0,1)$ and $\pi \in \Pi_\eta(M)$, including the switching policies returned by LyCore, the cycles between successive visits to $s_\circ$ are i.i.d. and have finite mean length. Let G and T be the total reward and length of one reference-to-reference cycle, and write $r_\pi(s) := \sum_{a\in\mathcal{A}} \pi(a \mid s)r(s,a)$. The cycles induce a unique long-run state occupancy d_M^π satisfying $\sum_s d_M^\pi(s)W(s) < \infty$. For every initial distribution, the average of the state distributions converges to d_M^π, which coincides with the unique stationary distribution induced by π. Moreover, for every initial distribution μ,*

$$\rho_M^\mu(\pi) = \rho_M(\pi) := \frac{\mathbb{E}G}{\mathbb{E}T} = \sum_{s\in\mathcal{S}} d_M^\pi(s)r_\pi(s). \tag{5}$$

The same conclusions hold for every projected core MDP on $\mathcal{K}_R$.

Proof. The fixed reference action and the strong Markov property make successive reference-to-reference cycles i.i.d. By Lemma C.1, their mean length is finite because $1 + \log w \le 1 + w$ and the reference-row moment is at most β. The killed drift recursion also gives

$$\mathbb{E}_x\left[\sum_{t=0}^{\tau_\circ-1} W(S_t)\right] \le \sum_{t\ge 0}(1-\eta)^t W(x) = \frac{W(x)}{\eta}.$$

Thus one cycle has finite expected reward, length, and W-occupation. The standard renewal-reward theorem for regenerative Markov chains (Meyn & Tweedie, 2009) gives (5), the invariant occupancy d_M^π, and its finite W moment. The initial segment before the first visit to $s_\circ$ is almost surely finite by Lemma C.1, so it does not affect the long-run average. Applying the same argument to state-indicator rewards gives the claimed Cesàro convergence and uniqueness. The projected core MDP has the same regenerative structure. □

C.1 Proof of Theorem 3.1

By Lemma C.3, the stable policy admits a long-run state occupancy $d = d_M^\pi$ with finite W moment and $\rho_M(\pi) = \sum_s d(s) r_\pi(s)$. Let $\bar{W} := \sum_s d(s) W(s)$. Because $\bar{W} < \infty$, invariance may be integrated against W. The reference-row moment bound, $W(s_\circ) = 1$, and the ζ-drift condition then give

$$\bar{W} \le \beta d(s_\circ) + (1-\zeta) \sum_{s \ne s_\circ} d(s) W(s) = (1-\zeta)\bar{W} + d(s_\circ)(\beta - 1 + \zeta) \le (1-\zeta)\bar{W} + \beta - 1 + \zeta.$$

Thus $\bar{W} \le (\beta - 1 + \zeta)/\zeta$. Since $W(s) > R$ on $\mathcal{S} \setminus \mathcal{K}_R$, Markov's inequality gives

$$d(\mathcal{S} \setminus \mathcal{K}_R) \le \frac{\beta - 1 + \zeta}{\zeta R} \le \frac{\beta}{\zeta R},$$

where the final inequality uses $\beta \ge 1$ and $\zeta \le 1$. This proves Theorem 3.1. □

C.2 Proof of Proposition 3.1 and Projected-Core Transfer

Both comparisons use the same regenerative coupling, so we state them together.

Lemma C.4 (Core replacement and projected-core transfer)**.** *Let $R \ge e$. First, let π have margin $\eta_\pi > 0$, and let $T_R^{\mathrm{stab}}\pi$ use π on $\mathcal{K}_R$ and an η_{stab}-stable policy outside it. Second, let π_R have margin $\eta_{\mathrm{core}} > 0$ in the projected core MDP M_R, and let $\widetilde{\pi}_R^{\mathrm{stab}}$ be its switching extension. For a universal constant C,*

$$|\rho_M(\pi) - \rho_M(T_R^{\mathrm{stab}}\pi)| \le C \frac{\beta(1+\log R)}{R} \left(\frac{1}{\eta_\pi} + \frac{1}{\eta_{\mathrm{stab}}} \right), \tag{6}$$

$$|\rho_{M_R}(\pi_R) - \rho_M(\widetilde{\pi}_R^{\mathrm{stab}})| \le C \frac{\beta(1+\log R)}{R} \left(\frac{1}{\eta_{\mathrm{core}}} + \frac{1}{\eta_{\mathrm{stab}}} \right). \tag{7}$$

Proof. Couple each pair of regenerative cycles until the first exit from $\mathcal{K}_R$. On the exit event, Lemma C.1 bounds the remaining length under either continuation by the inverse drift margin times $1 + \log W(S_{\sigma_R})$. The weighted overshoot bound in Lemma C.2 therefore gives both displayed bounds for the expected cycle-length difference. Bounded rewards give the same bounds for the cycle-reward difference. Finally, for cycle pairs (G, T) and $(\widetilde{G}, \widetilde{T})$ with rewards bounded by their lengths,

$$\left| \frac{\mathbb{E}G}{\mathbb{E}T} - \frac{\mathbb{E}\widetilde{G}}{\mathbb{E}\widetilde{T}} \right| \le |\mathbb{E}G - \mathbb{E}\widetilde{G}| + |\mathbb{E}T - \mathbb{E}\widetilde{T}|,$$

which completes both comparisons. □

Taking $\eta_\pi = \eta_{\mathrm{stab}} = \zeta$ in (6) proves Proposition 3.1. Stability is preserved because both policies have margin at least ζ.

D Drift Learning and Global Stability

This section proves the drift guarantees used in Step 2 of the upper bound. We use the standard empirical-Bernstein inequality (Maurer & Pontil, 2009): if $Y_1, \dots, Y_n \in [0, \alpha]$ have mean m, empirical mean $\widehat{m}$, and empirical variance $\widehat{v}$, then

$$U_n := \min\left\{ \alpha, \widehat{m} + \sqrt{\frac{2\widehat{v}\ell}{n}} + \frac{7\alpha\ell}{3(n-1)} \right\}$$

satisfies, with probability at least $1 - 3e^{-\ell}$,

$$m \le U_n \le m + c_B \left(\sqrt{\frac{m(\alpha - m)\ell}{n}} + \frac{\alpha\ell}{n} \right)$$

for a universal constant c_B. We only need to transfer this scalar bound to randomized action mixtures.

Lemma D.1 (Drift confidence bounds for randomized policies)**.** *Fix margins $0 < \eta < \zeta < 1$ and a confidence level $\delta \in (0,1)$, and define $\widehat{\mathcal{U}}_\eta(s) := \{\theta \in \Delta(\mathcal{A}) : \sum_{a\in\mathcal{A}} \theta_a \overline{m}(s,a) \le 1-\eta\}$. Let N_R be the number of queried non-reference states and set $\ell = \log(12|\mathcal{A}|\,N_R/\delta)$. There is a universal constant C_{stab} such that, if every queried non-reference row is sampled $n_{\mathrm{stab}} \ge C_{\mathrm{stab}}(\alpha-1+\zeta)\ell/(\zeta-\eta)^2$ times, then one can construct upper confidence bounds $\overline{m}(s,a)$ such that, with probability at least $1-\delta/4$, simultaneously for every queried non-reference state s: every $\theta \in \widehat{\mathcal{U}}_\eta(s)$ satisfies the true η-drift inequality, and every action-probability vector θ satisfying the true ζ-drift inequality belongs to $\widehat{\mathcal{U}}_\eta(s)$.*

Proof. For a fixed state s, write $m_a = \mathbb{E}_{S'\sim P(\cdot|s,a)}[W(S')]/W(s)$ and construct $\overline{m}(s,a) = U_a$ from the empirical-Bernstein bound above using $Y = W(S')/W(s) \in [0,\alpha]$. A union bound gives $m_a \le U_a$ for all queried rows and actions, and

$$0 \le e_a := U_a - m_a \le c_B\left(\sqrt{m_a(\alpha-m_a)u} + \alpha u\right), \qquad u := \ell/n_{\mathrm{stab}}.$$

Soundness is immediate: if $\sum_a \theta_a U_a \le 1-\eta$, then $\sum_a \theta_a m_a \le 1-\eta$.

For retention, fix θ such that $\bar{m} := \sum_a \theta_a m_a \le 1-\zeta$, put $\Delta_\eta := \zeta-\eta$, $d := 1-\eta-\bar{m} \ge \Delta_\eta$, and $c := \alpha-1+\zeta$. Cauchy–Schwarz gives $\sum_a \theta_a\sqrt{m_a(\alpha-m_a)} \le \sqrt{\bar{m}(\alpha-\bar{m})}$. Since $\bar{m} = 1-\eta-d$, we have $\bar{m}(\alpha-\bar{m}) \le c+d$, and $\Delta_\eta^2(c+d)/c \le d^2 + \Delta_\eta d \le 2d^2$. If $u \le c_0\Delta_\eta^2/c$ for a sufficiently small universal c_0, averaging the confidence widths over θ gives $\sum_a \theta_a e_a \le d$. Therefore $\sum_a \theta_a U_a \le 1-\eta$, which proves retention. □

D.1 Simultaneous Drift Events

Let $\Lambda \ge 1$, define $m(s,a) := \mathbb{E}[W(S') \mid s,a]/W(s)$, and let $\overline{m}(s,a)$ be the empirical-Bernstein upper bound computed from the independent drift batch. Set $\widehat{\mathcal{U}}(s) := \{\theta \in \Delta(\mathcal{A}) : \sum_a \theta_a \overline{m}(s,a) \le 1-\bar{\eta}_\kappa\}$. The first line below specifies the sample size. On an event $\mathcal{E}_{\mathrm{drift}}$ of probability at least $1-\delta_{\mathrm{drift}}$, the remaining two lines hold at every queried state:

$$\Lambda \ge \log\frac{12|\mathcal{A}|\,|K|}{\delta_{\mathrm{drift}}}, \qquad n_{\mathrm{stab}} := \left\lceil 16C_{\mathrm{stab}}\frac{(\alpha-1+\zeta)\Lambda}{\kappa^2\zeta^2}\right\rceil, \tag{8}$$

$$\theta \in \widehat{\mathcal{U}}(s) \Longrightarrow \sum_{a\in\mathcal{A}} \theta_a m(s,a) \le 1-\bar{\eta}_\kappa, \tag{9}$$

$$\sum_{a\in\mathcal{A}} \theta_a m(s,a) \le 1-\zeta \Longrightarrow \theta \in \widehat{\mathcal{U}}(s). \tag{10}$$

This follows from Lemma D.1 with $(\eta,\zeta) = (\bar{\eta}_\kappa,\zeta)$ because $\zeta - \bar{\eta}_\kappa = \kappa\zeta/2$.

For the independent model batch, define its projected normalized potential as $\widehat{m}_{\mathrm{mod}}(s,a) := W(s)^{-1}\sum_{x\in K}\widehat{P}(x \mid s,a)W(x)$. Because projection maps every out-of-core next state to $s_\circ$, $\widehat{m}_{\mathrm{mod}}(s,a)$ is no larger than the empirical mean of $W(S'_{s,a,i})/W(s)$ in the model batch. Applying Lemma D.1 again, now with margins $(\eta_{\mathrm{plan}},\bar{\eta}_\kappa)$, gives an independent event $\mathcal{E}_{\mathrm{model}}$ of probability at least $1-\delta_{\mathrm{model}}$ on which

$$\sum_{a\in\mathcal{A}} \theta_a m(s,a) \le 1-\bar{\eta}_\kappa \quad\Longrightarrow\quad \sum_{a\in\mathcal{A}} \theta_a \widehat{m}_{\mathrm{mod}}(s,a) \le 1-\eta_{\mathrm{plan}}. \tag{11}$$

For brevity, write $n(s) := n_{\mathrm{mod}}(s)$ for the model-batch size at state s. The implication holds simultaneously over all queried states if $n(s) \ge n_{\mathrm{stab}}$ and $\Lambda \ge \log(12|\mathcal{A}||K|/\delta_{\mathrm{model}})$. Indeed, $\bar{\eta}_\kappa - \eta_{\mathrm{plan}} = \kappa\zeta/4$, and the projection can only decrease the empirical potential.

On $\mathcal{E}_{\mathrm{drift}}$, every $\widehat{\mathcal{U}}(s)$ contains the true ζ-stable mixture selected by π_{stab}, every mixture in $\widehat{\mathcal{U}}(s)$ has true margin $\bar{\eta}_\kappa$, and every true ζ-stable comparator is retained. On $\mathcal{E}_{\mathrm{drift}} \cap \mathcal{E}_{\mathrm{model}}$, (11) also gives

$$\sum_{a\in\mathcal{A}} \theta_a \widehat{m}_{\mathrm{mod}}(s,a) \le 1-\eta_{\mathrm{plan}}, \qquad \forall\theta \in \widehat{\mathcal{U}}(s). \tag{12}$$

Let $\tau_\circ := \inf\{t \ge 0 : S_t = s_\circ\}$ and set $H := C_H(1+\log R)/\eta_{\mathrm{plan}}$ for a sufficiently large universal C_H. Applying Lemma C.1 to the empirical projected core MDP using Equation (12) and to the projected

core MDP using Equation (9) gives

$$\sup_{x\in K}\sup_{\pi\ \text{feasible}} \mathbb{E}_x^{\pi}\tau_\circ \le H, \qquad \sup_{x\in K}\sup_{\pi\ \text{feasible}} \mathbb{E}_x^{\pi}\tau_\circ^2 \le H^2. \tag{13}$$

Here "feasible" means empirical feasible in $\widehat{M}_R$ and true $\bar{\eta}_\kappa$-stable in M_R, respectively. Since $\eta_{\text{plan}} \ge \zeta/4$, we use $H = \widetilde{O}(\zeta^{-1})$ throughout.

E Constrained Average-Reward Planning

The empirical core is a finite constrained average-reward MDP. On $\mathcal{E}_{\text{drift}} \cap \mathcal{E}_{\text{model}}$, every feasible policy returns to $s_\circ$ with the uniform moment bound (13). The model is therefore regenerative and unichain. Standard finite-MDP and constrained-MDP results imply that the occupation-measure LP in (1) attains the optimal average reward and admits a stationary optimizer (Puterman, 1994; Altman, 1999). We record only the canonical bias properties needed for the statistical proof.

For $s \ne s_\circ$ and $\theta \in \widehat{\mathcal{U}}(s)$, write $\widehat{r}(s,\theta) := \sum_a \theta_a \widehat{r}(s,a)$ and $\widehat{P}_s^{\theta} := \sum_a \theta_a \widehat{P}_{s,a}$. The action at $s_\circ$ is fixed.

Theorem E.1 (Canonical constrained average-reward solution)**.** *Let $\widehat{\rho}$ be the optimum of (1). There is an empirical-feasible optimizer $\widehat{\pi}_R$ and a normalized bias $\widehat{h}$ satisfying*

$$\widehat{h}(s_\circ) = 0, \qquad \left\|\widehat{h}\right\|_\infty \le H, \tag{14}$$

$$\widehat{\rho} + \widehat{h}(s) = \max_{\theta\in\widehat{\mathcal{U}}(s)} \left\{\widehat{r}(s,\theta) + \widehat{P}_s^{\theta}\widehat{h}\right\}, \qquad s \ne s_\circ, \tag{15}$$

$$\widehat{\rho} = \widehat{r}(s_\circ,a_\circ) + \sum_{x\in K} \widehat{P}(x \mid s_\circ,a_\circ)\widehat{h}(x). \tag{16}$$

$$\widehat{\rho} + \widehat{h} = \widehat{r}^{\widehat{\pi}_R} + \widehat{P}^{\widehat{\pi}_R}\widehat{h}. \tag{17}$$

Here $\widehat{\pi}_R$ uses the fixed reference action and, at every non-reference state, any maximizing mixture in Equation (15). It maximizes the average reward in $\widehat{M}_R$ over all empirical-feasible policies.

With a fixed lexicographic tie-breaking rule, $(\widehat{\rho},\widehat{h},\widehat{\pi}_R)$ is a Borel measurable function of the data.

Proof. The occupation-measure correspondence for finite constrained MDPs identifies feasible stationary policies with feasible points of (1), including the statewise drift constraint (Puterman, 1994; Altman, 1999). Thus the LP optimum equals the optimal empirical average reward. The return bound (13) makes every feasible policy proper for the stochastic-shortest-path problem that stops at $s_\circ$. Standard SSP optimality equations then give Equations (15) to (17) (Bertsekas, 2012). The normalized bias is the expected centered reward accumulated before returning to $s_\circ$, so $\left\|\widehat{h}\right\|_\infty \le H$.

The empirical parameters are Borel functions of the data, and each feasible set is a compact polytope. Lexicographic minimization over the compact maximizing face therefore gives a Borel selector. This fixes a measurable version of $(\widehat{\rho},\widehat{h},\widehat{\pi}_R)$. □

Lemma E.1 (Canonical bias for a fixed retained comparator)**.** *Let $\pi^\dagger$ be a policy whose restriction to K is empirical feasible. Its empirical average reward and normalized bias satisfy*

$$\begin{aligned}
\widehat{\rho}^\dagger &:= \rho_{\widehat{M}_R}(\pi^\dagger),\\
\widehat{h}^\dagger(s_\circ) &:= 0,\\
\widehat{h}^\dagger(x) &:= \mathbb{E}_x^{\widehat{M}_R,\pi^\dagger}\sum_{t=0}^{\tau_\circ-1}\left(\widehat{r}^{\pi^\dagger}(S_t) - \widehat{\rho}^\dagger\right), \quad x \ne s_\circ.\\
\left\|\widehat{h}^\dagger\right\|_\infty &\le H,\\
\widehat{\rho}^\dagger + \widehat{h}^\dagger &= \widehat{r}^{\pi^\dagger} + \widehat{P}^{\pi^\dagger}\widehat{h}^\dagger.
\end{aligned} \tag{18}$$

For a fixed model-dependent but sample-independent comparator $\pi^\dagger$, the pair $(\widehat{\rho}^\dagger,\widehat{h}^\dagger)$ is Borel measurable in the samples.

Proof. Uniform properness gives the bound and makes the regenerative ratio finite. The first-step decomposition before hitting $s_\circ$ proves the non-reference Poisson equations. The reference-state equation is the identity $\mathbb{E}[G - \widehat{\rho}^\dagger T] = 0$ for one regenerative cycle. Measurability follows by the same finite-horizon limit argument as in Theorem E.1, now without maximization. □

F Shell-Weighted Plug-In Evaluation

This section proves the finite-core evaluation bound used in Step 4 of the upper-bound proof. The main difficulty is that the empirical bias functions of $\widehat{\pi}_R$ and $\pi^\dagger$ use the same model samples whose error we want to bound. We therefore cannot apply concentration directly to these random functions.

The proof has four steps. We first remove the samples at one state. We then prove a Bernstein bound for each transition row, convert the rowwise errors into an average-reward error, and aggregate the errors across potential shells. The final result is Theorem F.2.

F.1 Step 1: Statewise Leave-One-Out Representation

The first step removes this dependence one state at a time. We omit every action row at the chosen state because the feasible action mixture can depend on all of them. The remaining model determines the bias up to two scalar boundary parameters: the bias at the omitted state and the empirical average reward.

For a non-reference state s, let $\mathcal{F}_{-s}$ contain all drift samples and all model samples except the $|\mathcal{A}|$ batches at (s, a), $a \in \mathcal{A}$. At the reference state, $\mathcal{F}_{-\circ}$ excludes only the reference-row model batch. Any function measurable with respect to these sigma-fields is independent of the omitted samples.

We also need a local feasibility event. For each non-reference state x, let $\mathcal{H}_x$ be the event that $\widehat{\mathcal{U}}(x)$ is nonempty and $\sum_a \theta_a \widehat{m}_{\mathrm{mod}}(x, a) \le 1 - \eta_{\mathrm{plan}}$ for every $\theta \in \widehat{\mathcal{U}}(x)$. The event uses only the data at x and ensures the return-time bounds needed below.

F.1.1 Leave-One-Out Value Functions

Fix $s \neq s_\circ$, a boundary value $u \in [-H, H]$, and an average-reward parameter $\lambda \in [0, 1]$. Let $D_s := \{s, s_\circ\}$ and stop the process at $\tau_{D_s} := \inf\{t \ge 0 : S_t \in D_s\}$. On the event $\bigcap_{x \in K \setminus D_s} \mathcal{H}_x$, define

$$G^\star_{s,u,\lambda}(x) := \sup_\pi \mathbb{E}_x \left[\sum_{t=0}^{\tau_{D_s}-1} (\widehat{r}(S_t, A_t) - \lambda) + u \mathbf{1}\{S_{\tau_{D_s}} = s\} \right].$$

This stopped problem uses only rows and feasible sets outside D_s. Set $G^\star_{s,u,\lambda}(s) = u$ and $G^\star_{s,u,\lambda}(s_\circ) = 0$. When the event fails, set the function to zero. The resulting function is $\mathcal{F}_{-s}$-measurable.

For the fixed comparator, define $G^\dagger_{s,u,\lambda}$ by evaluating the same stopped problem under $\pi^\dagger$. We use this definition when $\mathcal{H}_x$ holds and $\pi^\dagger(\cdot \mid x) \in \widehat{\mathcal{U}}(x)$ for every $x \in K \setminus D_s$. Otherwise, set the function to zero.

The reference state needs no boundary value. For $\lambda \in [0, 1]$, define on the event $\bigcap_{x \in K \setminus \{s_\circ\}} \mathcal{H}_x$

$$G^\star_{\circ,\lambda}(x) := \sup_\pi \mathbb{E}_x \sum_{t=0}^{\tau_\circ - 1} (\widehat{r}(S_t, A_t) - \lambda), \qquad G^\star_{\circ,\lambda}(s_\circ) = 0,$$

Define $G^\dagger_{\circ,\lambda}$ in the same way for the fixed comparator when it is feasible at every non-reference state. Set both functions to zero outside their defining events. Neither function uses the reference-row samples, so both are $\mathcal{F}_{-\circ}$-measurable.

Lemma F.1 (Statewise leave-one-out representation)**.** *On $\mathcal{E}_{\mathrm{drift}} \cap \mathcal{E}_{\mathrm{model}}$, the leave-one-out values recover the empirical biases. For every non-reference state s,*

$$\widehat{h} = G^\star_{s,\widehat{h}(s),\widehat{\rho}}, \tag{19}$$

$$\widehat{h}^{\dagger} = G^{\dagger}_{s,\widehat{h}^{\dagger}(s),\widehat{\rho}^{\dagger}}. \tag{20}$$

$$\widehat{h} = G^{\star}_{\circ,\widehat{\rho}}, \qquad \widehat{h}^{\dagger} = G^{\dagger}_{\circ,\widehat{\rho}^{\dagger}}, \tag{21}$$

Each function is measurable with respect to its leave-one-out sigma-field.

Proof. On $\mathcal{E}_{\text{drift}} \cap \mathcal{E}_{\text{model}}$, every empirically feasible policy satisfies the return bounds in Equation (13). Fix $s \neq s_\circ$. Outside D_s, the canonical bias satisfies Equation (15), with boundary value $\widehat{h}(s)$ at s and zero at $s_\circ$. Stopping this equation at τ_{D_s} gives Equation (19). The same argument applied to the comparator's Poisson equation gives Equation (20). The reference-state identities follow from Equations (14) and (18).

Finite-horizon dynamic programming uses only data outside s. Passing to the uniformly convergent infinite-horizon limit, as in Theorem E.1, proves measurability. Setting the functions to zero outside the stated events preserves this property. □

Lemma F.2 (Bounds for the leave-one-out values)**.** *For every non-reference state s, every $u, u' \in [-H, H]$, and every $\lambda, \lambda' \in [0, 1]$, either $G = G^{\star}$ or $G = G^{\dagger}$ satisfies*

$$\left\|G_{s,u,\lambda}\right\|_\infty \leq 2H, \qquad \left\|G_{\circ,\lambda}\right\|_\infty \leq H, \tag{22}$$

$$\left\|G_{s,u,\lambda} - G_{s,u',\lambda'}\right\|_\infty \leq |u - u'| + H|\lambda - \lambda'|, \tag{23}$$

$$\left\|G_{\circ,\lambda} - G_{\circ,\lambda'}\right\|_\infty \leq H|\lambda - \lambda'|. \tag{24}$$

Proof. Because $\tau_{D_s} \leq \tau_\circ$, the expected magnitude of the running reward is at most H. The terminal payoff also has magnitude at most H. Hence the non-reference value is at most $2H$, while the reference-state value is at most H because it has no terminal payoff.

For a fixed policy, changing (u, λ) to (u', λ') changes the value by at most $|u - u'|\mathbb{P}(S_{\tau_{D_s}} = s) + |\lambda - \lambda'|\mathbb{E}\tau_{D_s}$. The return bound gives Equation (23). Taking the supremum preserves this inequality. At the reference state there is no boundary term, which gives Equation (24). □

This completes the first step: the empirical biases are represented by bounded functions that are independent of the omitted state samples.

F.2 Step 2: Support-Free Bellman Concentration

The second step uses the leave-one-out representation to prove concentration for each Bellman row. We use e to denote one sampled row. For a non-reference row $e = (s, a)$, let $X_{s,a,i} := \Gamma_R(S'_{s,a,i})$ be the projected next state and write $(R_{e,i}, X_{e,i}) := (R_{s,a,i}, X_{s,a,i})$. Let P_e, r_e, $\widehat{P}_e$, $\widehat{r}_e$, and n_e denote the true transition, mean reward, empirical estimates, and sample count for this row. At the reference state, set $e = \circ$, define $X_{\circ,i}$ in the same way, and let $n_e = n_\circ$.

For an independent $X \sim P_e$ and a bounded function $h : K \to \mathbb{R}$, define the Bellman-row error $\Delta_e(h) := (\widehat{r}_e - r_e) + (\widehat{P}_e - P_e)h$. Each model sample contains both a reward and a next state. Therefore, $\widehat{r}_e + \widehat{P}_e h$ is the sample average of $R_{e,i} + h(X_{e,i})$. A single Bernstein inequality can control the reward and transition errors together.

F.2.1 Deterministic Grids

For fixed u and λ, the leave-one-out function is independent of the omitted row. The values of u and λ that recover the empirical bias are random, so we cannot apply a union bound to them directly. We instead cover their ranges by deterministic grids. Let $n_s := n(s)$ for non-reference s. Define a boundary grid $\mathbb{U}_s \subset [-H, H]$ with mesh at most $(8n_s)^{-1}$ and an average-reward grid $\mathbb{L}_s \subset [0, 1]$ with mesh at most $(8Hn_s)^{-1}$. Include both endpoints. These grids can be chosen so that

$$|\mathbb{U}_s| \leq 16Hn_s + 2, \qquad |\mathbb{L}_s| \leq 8Hn_s + 2.$$

At the reference state, only the average-reward grid is needed. Use the same construction with $n_\circ$ in place of n_s.

Fix $\delta_B \in (0,1)$ and a sample-independent comparator $\pi^\dagger$. If its core restriction is not empirically feasible, set $(\widehat{\rho}^\dagger, \widehat{h}^\dagger) := (0,0)$. On $\mathcal{E}_{\text{drift}} \cap \mathcal{E}_{\text{model}}$, these quantities agree with those defined in Lemma E.1.

Let c_B be a sufficiently large universal constant. Define $n_{\max} := \max\{n_\circ, \max_{s \neq s_\circ} n_s\}$ and choose Λ so that

$$\Lambda \geq \log\left(\frac{c_B |\mathcal{A}|\,|K| H^2 (1+n_{\max})^2}{\delta_B}\right), \tag{25}$$

Theorem F.1 (Support-free Bellman concentration)**.** *With probability at least $1-\delta_B$, the following statement holds on $\mathcal{E}_{\text{drift}} \cap \mathcal{E}_{\text{model}}$. For every queried row and for both $h = \widehat{h}$ and $h = \widehat{h}^\dagger$,*

$$|\Delta_e(h)| \leq c_B \left[\sqrt{\frac{(\mathrm{Var}_{P_e}(h(X)) + 1)\Lambda}{n_e}} + \frac{H\Lambda}{n_e}\right]. \tag{26}$$

The bound contains no next-state support cardinality.

Proof. We prove the claim for the optimizer. The proof for the comparator is identical after replacing $G^\star$ with $G^\dagger$.

Fix a non-reference state s and condition on $\mathcal{F}_{-s}$. For each $(u,\lambda) \in \mathbb{U}_s \times \mathbb{L}_s$, the function $G := G^\star_{s,u,\lambda}$ is fixed and independent of the omitted samples at state s. By Equation (22), $\|G\|_\infty \leq 2H$.

For a fixed action a, define $Z_i := R_{s,a,i} + G(X_{s,a,i})$. Then $n_s^{-1}\sum_i Z_i - \mathbb{E}Z_i = \Delta_{s,a}(G)$. The range of Z_i has length at most $1+4H \leq 5H$, and $\mathrm{Var}(Z_i) \leq 2\,\mathrm{Var}_{P_{s,a}}(G(X)) + 1$. Thus, conditional Bernstein gives, for a universal constant c and every $\ell > 0$,

$$\mathbb{P}\left(|\Delta_{s,a}(G)| > c\left[\sqrt{\frac{(\mathrm{Var}_{P_{s,a}} G + 1)\ell}{n_s}} + \frac{H\ell}{n_s}\right] \middle| \mathcal{F}_{-s}\right) \leq 2e^{-\ell}. \tag{27}$$

The same inequality holds when the defining event fails because G is then zero.

Set $\ell = \Lambda$ and take a union bound over the actions and grid points. After removing the conditioning, the failure probability at state s is at most $c|\mathcal{A}|H^2(1+n_s)^2 e^{-\Lambda}$. At the reference state, condition on $\mathcal{F}_{-\circ}$ and use the one-dimensional family $G^\star_{\circ,\lambda}$. Finally, take a union bound over all states and over both the optimizer and comparator families. By Equation (25), the total failure probability is at most δ_B.

It remains to replace the grid function by the random empirical bias. By Lemma F.1, $\widehat{h} = G^\star_{s,\widehat{h}(s),\widehat{\rho}}$. Choose $(u_s,\lambda_s) \in \mathbb{U}_s \times \mathbb{L}_s$ with $|u_s - \widehat{h}(s)| \leq (8n_s)^{-1}$ and $|\lambda_s - \widehat{\rho}| \leq (8Hn_s)^{-1}$. The concentration event is uniform over the grid, so we may choose these points after observing the omitted samples. Set $G_s := G^\star_{s,u_s,\lambda_s}$. The Lipschitz bounds in Equation (23) give

$$\left\|G_s - \widehat{h}\right\|_\infty \leq \frac{1}{4n_s}. \tag{28}$$

$$|\Delta_{s,a}(\widehat{h}) - \Delta_{s,a}(G_s)| = |(\widehat{P}_{s,a} - P_{s,a})(\widehat{h} - G_s)| \leq \frac{1}{2n_s}. \tag{29}$$

$$\sqrt{\mathrm{Var}_P(f)} \leq \sqrt{\mathrm{Var}_P(g)} + 2\,\|f-g\|_\infty. \tag{30}$$

The first two inequalities control the error caused by replacing the empirical bias with a nearby grid function. The third inequality holds for every probability distribution P and bounded functions f, g. Apply it with $f = G_s$ and $g = \widehat{h}$. By Equation (28), the change in the variance term is absorbed by $H\Lambda/n_s$ because $H, \Lambda \geq 1$. Combining Equations (27), (29) and (30) proves Equation (26) at every non-reference row. The reference row follows from the one-dimensional grid and Equation (24). The comparator follows in the same way from Equations (20) and (21). □

Remark F.1 (Scope of Bellman concentration)**.** The theorem is uniform over all rows for the two bias functions used in the plug-in comparison. We do not need a uniform bound over the entire policy class.

This completes the second step: each random bias now satisfies a rowwise concentration bound that is independent of the number of possible next states.

F.3 Step 3: Exact Performance Identity and Variance Control

The third step converts the rowwise errors into an average-reward error. We first derive an exact performance identity. We then control the variance term in the rowwise Bernstein bound.

Recall $\mathcal{E} := \mathcal{E}_{\text{drift}} \cap \mathcal{E}_{\text{model}} \cap \mathcal{E}_{\text{B}}$ and work on this event. Fix $\pi \in \{\widehat{\pi}_R, \pi^\dagger\}$, and let $(\widehat{\rho}^\pi, \widehat{h}^\pi)$ be its empirical average reward and canonical empirical bias. For $\widehat{\pi}_R$, these are $(\widehat{\rho}, \widehat{h})$.

On $\mathcal{E}$, all return-time and bias bounds proved above apply. Let P^π, r^π, and μ^π be the policy kernel, reward, and invariant distribution in the projected core MDP. Define the policy-averaged Bellman error $z(s)$ and one-step variance $v(s)$ by

$$z(s) := \sum_a \pi(a \mid s)\Delta_{s,a}(\widehat{h}^\pi),$$
$$v(s) := \mathrm{Var}_{P^\pi(\cdot|s)}(\widehat{h}^\pi(S')).$$

At $s_\circ$, the definition of $z(s_\circ)$ uses the fixed reference action.

Lemma F.3 (Policy-averaged row error)**.** *For a universal constant c_0, every non-reference state satisfies*

$$|z(s)| \le c_0 \left[\sqrt{\frac{(v(s)+1)\Lambda}{n(s)}} + \frac{H\Lambda}{n(s)} \right]. \tag{31}$$

The same bound holds at the reference state with $n_\circ$ in place of $n(s)$.

Proof. Average Equation (26) over actions. Cauchy–Schwarz and the law of total variance give the result. □

Lemma F.4 (Exact average-reward identity)**.** *The empirical bias satisfies*

$$\widehat{h}^\pi - P^\pi \widehat{h}^\pi = r^\pi - \widehat{\rho}^\pi + z, \tag{32}$$
$$\widehat{\rho}^\pi - \rho_{M_R}(\pi) = \mu^\pi z. \tag{33}$$

Proof. Add and subtract the true reward and transition terms in the empirical Poisson equation to obtain Equation (32). Averaging under μ^π cancels $\mu^\pi(\widehat{h}^\pi - P^\pi \widehat{h}^\pi)$ and gives Equation (33). □

F.3.1 Global Self-Bounding of Quadratic Variation

The identity above reduces the evaluation problem to controlling $\mu^\pi|z|$. The rowwise Bernstein bound depends on $v(s)$, so we first bound its long-run average. This quantity is the average one-step conditional variance accumulated by the chain. Let $q_* := \max_{s \in K} \Lambda/n(s)$, where $n(s_\circ) := n_\circ$. Define $V := \sum_s \mu^\pi(s)v(s)$ and $E_z := \sum_s \mu^\pi(s)|z(s)|$.

Lemma F.5 (Global self-bounded quadratic variation)**.** *There are universal constants $c_{\text{sb}}, C_{\text{V}} > 0$ such that, if $Hq_* \le c_{\text{sb}}$, then*

$$V \le C_{\text{V}} H. \tag{34}$$

Proof. Write $h := \widehat{h}^\pi$, $\widehat{\rho} := \widehat{\rho}^\pi$, and $g := r^\pi - \widehat{\rho} + z$. Equation (32) becomes $h = P^\pi h + g$. Invariance of μ^π gives

$$\begin{aligned} V &= \mu^\pi \left(P^\pi h^2 - (P^\pi h)^2 \right) \\ &= \mu^\pi h^2 - \mu^\pi (h - g)^2 \\ &= 2\mu^\pi (hg) - \mu^\pi g^2 \\ &\le 2H\mu^\pi |g| \\ &\le 2H(1 + E_z), \end{aligned} \tag{35}$$

This first relates the total variance V to the total Bellman error E_z. Both r^π and $\widehat{\rho}$ lie in $[0, 1]$, so $|r^\pi - \widehat{\rho}| \le 1$ in the last line.

We next bound E_z in terms of V. Apply Lemma F.3, Jensen's inequality, and $\Lambda/n(s) \le q_*$:

$$E_z \le c_0\sqrt{q_*}\sum_s \mu^\pi(s)\sqrt{v(s)+1} + c_0 H q_* \le c_0\sqrt{q_*(V+1)} + c_0 H q_*. \tag{36}$$

Combining Equation (36) with Equation (35), and using $\sqrt{V+1} \le \sqrt{V}+1$, gives

$$V \le 2H + 2c_0 H\sqrt{q_* V} + 2c_0 H\sqrt{q_*} + 2c_0 H^2 q_*.$$

Young's inequality gives $2c_0 H\sqrt{q_* V} \le V/2 + 2c_0^2 H^2 q_*$. If $Hq_* \le c_{\mathrm{sb}}$ for a sufficiently small universal constant, all remaining terms are bounded by a constant times H. Moving $V/2$ to the left proves Equation (34). □

F.3.2 Stopped Quadratic Variation from an Arbitrary State

The global lemma controls variance after averaging under the invariant distribution. The shell analysis also needs a bound from the state at which a shell is first reached. We therefore control the variance accumulated before the process returns to the reference state. For $x \in K$, define $Q_x := \mathbb{E}_x^{M_R,\pi}\sum_{t=0}^{\tau_\circ - 1} v(S_t)$ and $Q_* := \max_{x\in K} Q_x$.

Lemma F.6 (Stopped self-bounded quadratic variation)**.** *There are universal constants $c'_{\mathrm{sb}}, C_Q > 0$ such that, if $Hq_* \le c'_{\mathrm{sb}}$, then*

$$Q_* \le C_Q H^2. \tag{37}$$

Proof. We first use a stopped martingale identity. Write $h := \widehat{h}^\pi$ and define the martingale difference $\xi_{t+1} := h(S_{t+1}) - P^\pi h(S_t)$ for $t < \tau_\circ$. For the truncated return time $\tau_N := \tau_\circ \wedge N$, the martingale isometry gives

$$\mathbb{E}_x\left(\sum_{t<\tau_N}\xi_{t+1}\right)^2 = \mathbb{E}_x\sum_{t<\tau_N} v(S_t). \tag{38}$$

The increments are bounded by $2H$, and $\mathbb{E}\tau_\circ^2 < \infty$. Hence the stopped martingales are uniformly square integrable. Letting $N \to \infty$ in Equation (38) gives $Q_x = \mathbb{E}_x(\sum_{t<\tau_\circ}\xi_{t+1})^2$.

We next express the martingale sum through the perturbed Poisson equation. Summing $h - P^\pi h = r^\pi - \widehat{\rho} + z$ until $\tau_\circ$, and using $h(S_{\tau_\circ}) = 0$, gives

$$\sum_{t<\tau_\circ}\xi_{t+1} = -h(x) + \sum_{t<\tau_\circ}(r^\pi(S_t) - \widehat{\rho}) + \sum_{t<\tau_\circ} z(S_t).$$

The first term is bounded by H. The second sum has absolute value at most $\tau_\circ$, whose second moment is at most H^2. Hence

$$Q_x \le CH^2 + C\mathbb{E}_x\left(\sum_{t<\tau_\circ}|z(S_t)|\right)^2. \tag{39}$$

It remains to control the last term in Equation (39). By Equation (31) and $(a+b)^2 \le 2a^2 + 2b^2$, we have $z(s)^2 \le a_0 v(s) + b_0$, where $a_0 := Cq_*$ and $b_0 := C(q_* + H^2 q_*^2)$. Cauchy–Schwarz gives $(\sum_{t<\tau_\circ}|z(S_t)|)^2 \le \tau_\circ\sum_{t<\tau_\circ} z(S_t)^2$.

Expand this product over pairs of time indices. Conditioning at the earlier time and applying the strong Markov property gives $\mathbb{E}_x[\tau_\circ\sum_{t<\tau_\circ} v(S_t)] \le 2HQ_*$. Together with $\mathbb{E}\tau_\circ^2 \le H^2$, Equation (39) yields

$$Q_* \le CH^2 + Ca_0 HQ_* + Cb_0 H^2.$$

If Hq_* is sufficiently small, then $Ca_0 H \le 1/2$, while $b_0 H^2$ is bounded by a constant times H^2. Absorbing the resulting $Q_*/2$ term into the left side proves Equation (37). □

Together, the two variance bounds complete the third step. They control both stationary variance and the variance accumulated after entering an outer shell.

F.4 Step 4: Aggregation Across Potential Shells

The final step adds the rowwise errors across the core. Outer states receive smaller weights because stable policies reach large potential levels less often. To make this decay explicit, partition the core into one inner block and dyadic potential shells:

$$\begin{aligned} r_{\text{in}} &:= \min\{R, 2\beta H\}, \\ \mathcal{B}_0 &:= \{s \in K : W(s) \le r_{\text{in}}\}, \\ r_j &:= 2^{j-1} r_{\text{in}}, \qquad \mathcal{B}_j := \{s \in K : r_j < W(s) \le \min\{2r_j, R\}\}, \\ p_j &:= \min\left\{1, \frac{2\beta}{r_j}\right\}. \end{aligned}$$

Here $j = 1, \dots, L$, where L is the smallest integer for which these shells cover K. Up to a universal constant, p_j bounds the probability that one regenerative cycle reaches shell j. For each block, write $M_j := \sum_{s\in\mathcal{B}_j} \mu^\pi(s)$ and $V_j := \sum_{s\in\mathcal{B}_j} \mu^\pi(s)v(s)$.

Lemma F.7 (Shell occupancy and shell quadratic variation)**.** *On $\mathcal{E}$, for either $\pi = \widehat{\pi}_R$ or $\pi = \pi^\dagger$,*

$$M_j \le C p_j H, \tag{40}$$

$$V_j \le C p_j H^2, \tag{41}$$

$$M_0 \le 1, \qquad V_0 \le CH.$$

Proof. The inner-block bounds are immediate. We have $M_0 \le 1$ because μ^π has total mass one, and $V_0 \le CH$ follows from Lemma F.5.

Fix an outer shell $\mathcal{B}_j$. Before the process reaches either this shell or the reference state, its Lyapunov drift is nonpositive. Together with the reference-row moment bound and the projection of core exits to the reference state, Lemma C.2 gives

$$\mathbb{P}_{s_\circ}(\text{one cycle hits } \mathcal{B}_j) \le C p_j. \tag{42}$$

After the first hit, the strong Markov property and Equation (13) bound the expected remaining cycle length by H. Thus, the expected number of visits to shell j in one cycle is at most Cp_jH. Dividing by the expected cycle length, which is at least one, proves Equation (40).

The variance bound follows in the same way. After the first hit, Lemma F.6 bounds the expected remaining sum of $v(S_t)$ by $Q_* \le CH^2$. Multiplying by the hitting probability in Equation (42), and then dividing by the expected cycle length, proves Equation (41). □

Theorem F.2 (Shell-weighted plug-in evaluation)**.** *Suppose every state in block $\mathcal{B}_j$ uses a common row sample size n_j. The reference state and the inner block use n_0. If $H\Lambda/n_j$ is below the universal self-bounding threshold for every block, then, on $\mathcal{E}$, each $\pi \in \{\widehat{\pi}_R, \pi^\dagger\}$ satisfies*

$$\begin{aligned} \left|\rho_{\widehat{M}_R}(\pi) - \rho_{M_R}(\pi)\right| \le C &\left[\sqrt{\frac{H\Lambda}{n_0}} + \frac{H\Lambda}{n_0}\right] \\ &+ C\sum_{j=1}^{L}\left[p_j H^{3/2}\sqrt{\frac{\Lambda}{n_j}} + \frac{p_j H^2 \Lambda}{n_j}\right]. \end{aligned} \tag{43}$$

Proof. By Lemma F.4, it is enough to bound $\sum_s \mu^\pi(s)|z(s)|$. We bound this sum one block at a time. Consider a block $\mathcal{B}$ with common sample size n, and write $M_\mathcal{B} := \sum_{s\in\mathcal{B}} \mu^\pi(s)$ and $V_\mathcal{B} := \sum_{s\in\mathcal{B}} \mu^\pi(s)v(s)$. Applying Lemma F.3 and weighted Cauchy–Schwarz gives

$$\begin{aligned} \sum_{s\in\mathcal{B}} \mu^\pi(s)|z(s)| &\le C\sqrt{\frac{\Lambda}{n}} \sum_{s\in\mathcal{B}} \mu^\pi(s)\sqrt{v(s)+1} + C\frac{H\Lambda}{n} M_\mathcal{B} \\ &\le C\sqrt{\frac{\Lambda M_\mathcal{B}(V_\mathcal{B} + M_\mathcal{B})}{n}} + C\frac{H\Lambda}{n} M_\mathcal{B}. \end{aligned}$$

For the inner block, substitute $M_0 \le 1$ and $V_0 \le CH$. For shell j, substitute $M_j \le Cp_jH$ and $V_j \le Cp_jH^2$. Since $H \ge 1$, we have $\sqrt{M_j(V_j + M_j)} \le Cp_jH^{3/2}$. Summing the resulting bounds over the inner block and all shells proves Equation (43). □

G Sample Allocation and Sample Counting

This section chooses the model and drift sample sizes so that the evaluation bound above is at most $\varepsilon/16$. It then compares the empirical optimizer with the retained comparator and sums the samples used for performance learning and stability learning.

Let $L_+ := L + 1$ be the total number of blocks including the inner block, and recall that $\varepsilon_{\mathrm{pl}} = \varepsilon/16$. For a sufficiently large universal constant C_{perf}, define the performance sample requirements

$$n_0^{\mathrm{perf}} := \left\lceil C_{\mathrm{perf}} \Lambda \frac{H L_+^2}{\varepsilon_{\mathrm{pl}}^2} \right\rceil, \tag{44}$$

$$n_j^{\mathrm{perf}} := \left\lceil C_{\mathrm{perf}} \Lambda \left(H + \frac{p_j^2 H^3 L_+^2}{\varepsilon_{\mathrm{pl}}^2} + \frac{p_j H^2 L_+}{\varepsilon_{\mathrm{pl}}} \right) \right\rceil, \qquad j \geq 1. \tag{45}$$

The third term in Equation (45) is an explicit Bernstein-linear-term allocation. It removes any need to infer that the outermost shell has $p_j H \gtrsim \varepsilon$ and is dominated by the final target rate, as shown below.

For every non-reference state s in block j, use the model sample size

$$n_{\mathrm{mod}}(s) := n_j := \max\{n_j^{\mathrm{perf}}, n_{\mathrm{stab}}\}. \tag{46}$$

At the reference state, use $n_{\mathrm{mod}}(s_\circ) = n_\circ = n_0^{\mathrm{perf}}$. Independently, every non-reference row receives n_{stab} drift samples.

G.1 Confidence Level and Feasibility of the Allocation

Because the leave-one-out grid cardinality contains $n_{\max}$, define $n_{\max}(\ell)$ by replacing Λ by ℓ in Equations (8) and (44) to (46). For a sufficiently large universal constant c_Λ, choose Λ to be the smallest integer $\ell \geq 1$ satisfying

$$\ell \geq \log\left(\frac{c_\Lambda |\mathcal{A}|\, |K| H^2 (1 + n_{\max}(\ell))^2}{\delta} \right). \tag{47}$$

Such an integer exists because the right side is $O(\log \ell)$ while the left side is ℓ. Since $n_{\max}(\ell)$ is linear in ℓ times a polynomial in $(H, L_+, \varepsilon^{-1}, \kappa^{-1}, \zeta^{-1}, \alpha)$, the minimal solution satisfies

$$\Lambda = O\left(\log \frac{e|\mathcal{A}|\, c_W R^\nu (H+1)(L_+ + 1)(\alpha + 1)}{\delta \kappa \zeta \varepsilon} \right).$$

Thus it contributes only logarithmic factors. Recall that $\delta_{\mathrm{drift}} = \delta_{\mathrm{model}} = \delta_{\mathrm{B}} = \delta/4$. The constant c_Λ is selected large enough that Equation (47) implies the logarithmic requirements of Lemma D.1 and Theorem F.1. Their total failure probability is at most $3\delta/4$.

For a sufficiently large C_{perf}, the allocation satisfies, on $\mathcal{E}$ and for both $\pi \in \{\widehat{\pi}_R, \pi^\dagger\}$,

$$\max_j \frac{H\Lambda}{n_j} \leq c'_{\mathrm{sb}}, \tag{48}$$

$$\left| \rho_{\widehat{M}_R}(\pi) - \rho_{M_R}(\pi) \right| \leq \varepsilon_{\mathrm{pl}}. \tag{49}$$

The floor $C_{\mathrm{perf}} H \Lambda$ gives (48). For the inner block,

$$\sqrt{\frac{H\Lambda}{n_0}} \leq \frac{\varepsilon_{\mathrm{pl}}}{\sqrt{C_{\mathrm{perf}}} L_+},$$

$$\frac{H\Lambda}{n_0} \leq \frac{\varepsilon_{\mathrm{pl}}^2}{C_{\mathrm{perf}} L_+^2} \leq \frac{\varepsilon_{\mathrm{pl}}}{C_{\mathrm{perf}} L_+}.$$

For shell j, the quadratic and linear terms give the corresponding two terms in (43), each at most a constant times $\varepsilon_{\mathrm{pl}}/L_+$. Summing over at most L_+ blocks proves (49).

Retention in (10) makes the fixed comparator $\pi_R^\dagger$ empirically feasible. Empirical optimality and (49) then give

$$\begin{aligned}\rho_{M_R}(\widehat{\pi}_R) &\geq \rho_{\widehat{M}_R}(\widehat{\pi}_R) - \varepsilon_{\mathrm{pl}} \\ &\geq \rho_{\widehat{M}_R}(\pi_R^\dagger) - \varepsilon_{\mathrm{pl}} \\ &\geq \rho_{M_R}(\pi_R^\dagger) - 2\varepsilon_{\mathrm{pl}}.\end{aligned}$$

The first inequality is valid for the data-dependent optimizer because Theorem F.1 uses statewise leave-one-out.

G.2 Performance and Stability Sample Counts

Lemma G.1 (Support-free performance sample complexity)**.** *With the allocation Equations (44) and (45), the total performance requirement before imposing the model-stability floor n_{stab} is*

$$N_{\mathrm{perf}} = \widetilde{O}\left(\frac{|\mathcal{A}|\, c_W \beta^\nu}{\zeta^{\nu+1}\varepsilon^{\max\{2,\nu\}}}\right). \tag{50}$$

Proof. Logarithmic factors L_+ and Λ are suppressed. Since $r_{\mathrm{in}} \leq 2\beta H$, the number of inner state-action rows is $O(|\mathcal{A}|\, c_W(\beta H)^\nu)$. Their contribution is

$$\widetilde{O}\left(|\mathcal{A}|\, c_W(\beta H)^\nu \frac{H}{\varepsilon^2}\right) = \widetilde{O}\left(|\mathcal{A}|\, c_W \beta^\nu H^{\nu+1}\varepsilon^{-2}\right).$$

For shell j, $|\mathcal{B}_j| \leq c_W(2r_j)^\nu$. When shells exist, $r_j \geq r_{\mathrm{in}} = 2\beta H$, so $p_j \asymp \beta/r_j$. The square-root allocation term contributes

$$\widetilde{O}\left(|\mathcal{A}|\, c_W \frac{\beta^2 H^3}{\varepsilon^2} \sum_{j=1}^{L} r_j^{\nu-2}\right). \tag{51}$$

The dyadic sum is dominated by $r_{\mathrm{in}}^{\nu-2}$ for $\nu < 2$, logarithmic for $\nu = 2$, and dominated by $R^{\nu-2}$ for $\nu > 2$. Therefore Equation (51) is

$$\widetilde{O}\left(|\mathcal{A}|\, c_W \beta^\nu H^{\nu+1}\varepsilon^{-\max\{2,\nu\}}\right), \tag{52}$$

using $R = \widetilde{O}(\beta H/\varepsilon)$.

The explicit Bernstein-linear allocation contributes $\widetilde{O}(|\mathcal{A}|c_W\beta H^2\varepsilon^{-1}\sum_{j=1}^{L} r_j^{\nu-1})$. If $\nu < 1$, this is dominated by the inner endpoint and equals at most $\widetilde{O}(|\mathcal{A}|\, c_W\beta^\nu H^{\nu+1}\varepsilon^{-1})$. If $\nu = 1$, it has only an additional logarithm. If $\nu > 1$, it is dominated by $R^{\nu-1}$ and equals at most

$$\widetilde{O}\left(|\mathcal{A}|\, c_W \beta^\nu H^{\nu+1}\varepsilon^{-\nu}\right).$$

In every case it is bounded by Equation (52) because $\varepsilon < 1$.

Finally, the universal $H\Lambda$ floor over all core rows costs $\widetilde{O}(|\mathcal{A}|c_W R^\nu H) = \widetilde{O}(|\mathcal{A}|c_W\beta^\nu H^{\nu+1}\varepsilon^{-\nu})$, which is again dominated by Equation (52). Substitute $H = \widetilde{O}(\zeta^{-1})$ to obtain Equation (50). □

Lemma G.2 (Stability sample complexity)**.** *The total number of drift samples and the additional model samples induced by the floor n_{stab} satisfy*

$$N_{\mathrm{stab}} = \widetilde{O}\left(|\mathcal{A}|\, c_W \beta^\nu \frac{\alpha - 1 + \zeta}{\kappa^2 \zeta^{\nu+2}\varepsilon^\nu}\right).$$

Proof. There are at most $|\mathcal{A}|c_W R^\nu$ non-reference rows. Each receives n_{stab} drift samples, and imposing the same floor on the independent model batch adds at most another n_{stab} samples per row beyond the performance allocation. Multiplying by the row count, applying Equation (8), and using $R = \widetilde{O}(\beta/(\zeta\varepsilon))$ gives the result. The factor of two is absorbed by the universal constant. □

H Proof of Theorem 5.2

The proof has two common testing ingredients followed by two hard subclasses. The reference-state construction places many independent decision states at a chosen potential radius, and the adaptive direct-sum lemma converts the action mass required by a successful output into a sample lower bound. We then use one subclass for performance learning and one for stability learning before verifying the model conditions and combining the rates. Universal constants may change from line to line.

H.1 Testing Tools

This subsection gives the two tools used by both hard families. The first places many decision states at a prescribed potential level. The second converts the action mass required from a successful output into an adaptive testing lower bound.

Lemma H.1 (Direct transition to a prescribed potential level)**.** *Let $m \geq 2$, $r \geq 2$, and $1 < \beta_0 \leq r/2$. There is a common reference row that stays at $s_\circ$ with probability $1-\lambda$ and transitions directly to each of m leaves with probability λ/m, where $\lambda = (\beta_0 - 1)/(r-1)$. Every leaf has potential value r, the reference-row moment is β_0, each leaf is reached with conditional probability $1/m$ after leaving the reference state, and its unconditional probability per regenerative cycle is $\lambda/m = \Theta((\beta_0 - 1)/(mr))$.*

Proof. Since $W(s_\circ) = 1$, the reference-row moment is $(1-\lambda) + \lambda r = \beta_0$. The condition $\beta_0 \leq r/2$ gives $\lambda \in (0,1)$, while $r/(r-1) \in [1,2]$ gives $\lambda = \Theta((\beta_0 - 1)/r)$. The remaining claims follow from the equal-split definition of the row. □

Theorem H.1 (Permutation-bandit lower bound)**.** *There are universal constants $c_{\mathrm{pb}} > 0$ and $\delta_{\mathrm{pb}} \in (0, 1/2)$ with the following property. For $k \geq 2$, write $[k] := \{1, \ldots, k\}$. For $e \in (0, 1/4)$, consider the k bandit instances indexed by $\theta \in [k]$ in which arm θ has law $\mathrm{Ber}(1/2 + e)$ and every other arm has law $\mathrm{Ber}(1/2 - e)$. Any possibly randomized adaptive procedure with a data-dependent stopping time T and recommendation $\widehat{\theta}$ satisfying $\inf_{\theta \in [k]} \mathbb{P}_\theta(\widehat{\theta} = \theta) \geq 1 - \delta_{\mathrm{pb}}$ must obey $\sup_{\theta \in [k]} \mathbb{E}_\theta[T] \geq c_{\mathrm{pb}} k/e^2$.*

This is the constant-confidence specialization of Mannor & Tsitsiklis (2004, Theorem 13), whose model is a known vector of Bernoulli biases with an unknown permutation. We use it as an external theorem and do not reproduce its proof.

Lemma H.2 (Adaptive direct-sum bound for rare observations)**.** *Let $u \in (0,1)$ and $0 < d < \frac{1}{2}\min\{u, 1-u\}$. Consider m independent tasks with $k \geq 2$ Bernoulli arms each, where $[k] := \{1, \ldots, k\}$. Hidden identities $\Theta_1, \ldots, \Theta_m$ are independent and uniform on $[k]$. Arm Θ_i in task i has mean $u + d$, and every other arm has mean $u - d$. An adaptive algorithm may interleave queries across tasks and stop after a total of N queries. All expectations average over this product prior, the observations, and the algorithm's internal randomization. If it outputs sub-probability vectors $g_i \in [0,1]^k$ with $\sum_{a=1}^k g_{i,a} \leq 1$ and $m^{-1}\sum_i \mathbb{E}[g_{i,\Theta_i}] \geq 13/24$, then, for a universal constant $c_0 > 0$,*

$$\lambda := 2\min\{u, 1-u\}, \qquad e := \frac{d}{\lambda} < \frac{1}{4}, \tag{53}$$

$$\mathbb{E}N \geq c_0 \frac{mk}{\lambda e^2}. \tag{54}$$

Consequently, the same bound holds for $\sup_{\boldsymbol{\theta} \in [k]^m} \mathbb{E}_{\boldsymbol{\theta}} N$.

Proof. After stopping, draw $\widehat{\Theta}_i$ from g_i, sending missing mass to a failure symbol. Writing $p_i = \mathbb{P}(\widehat{\Theta}_i = \Theta_i)$, the average condition and $p_i \leq 1$ imply that at least $m/23$ tasks satisfy $p_i \geq 25/48$.

Fix such a task i, let N_i denote the number of queries sent to that task, and construct a single-task procedure. It draws the other identities $\Theta_{-i} := (\Theta_j)_{j \neq i}$ independently from their uniform prior and simulates all observations requested from those tasks using their known Bernoulli laws. These internally generated observations are independent of Θ_i and require no pulls from the single task. Queries to task i are forwarded to the external bandit instance. Before running this simulation, draw a uniform permutation of the k task-i labels and undo it on the recommendation. For every fixed external hidden

identity, the permuted identity seen by the original algorithm is uniform. Consequently, the single-task procedure has success probability exactly p_i and expected forwarded-query count exactly $\mathbb{E}N_i$, both independent of the fixed external identity. Repeat this symmetrized procedure a universal number L of times and return an identity only when it appears in a strict majority. Since $p_i - 1/2 \geq 1/48$, Hoeffding's inequality permits a universal L for which the amplified error is at most δ_{pb}.

It remains to simulate each Bernoulli observation from a centered coin. Independently activate a query with probability λ. On activation, pull the corresponding centered arm and return its outcome. Without activation, return zero when $u \leq 1/2$ and one when $u > 1/2$. The resulting means are $u + d$ and $u - d$ by Equation (53). If $\mathbb{E}N_i = \infty$, the claim is immediate. Otherwise, conditional expectation at successive queries gives an expected $\lambda\mathbb{E}N_i$ centered-coin pulls per run. Thus the amplified procedure uses $L\lambda\mathbb{E}N_i$ pulls in expectation. Applying Theorem H.1 and absorbing the universal L yields $\mathbb{E}N_i \geq ck/(\lambda e^2)$. Summing this inequality over the at least $m/23$ informative tasks proves Equation (54). The worst-case conclusion follows because the product-prior expectation is at most the supremum over hidden vectors. □

H.2 Common Hard-Instance Geometry

We first specify the reference-state geometry shared by the two hard families. It determines how often the process reaches each hard decision state while keeping the reference-row moment within the model class.

Shared reference-state construction.

In both hard subclasses, take $\beta_0 = (\beta+1)/2$ in Lemma H.1. Since $\beta \geq 2$, one has $\beta/4 \leq \beta_0 - 1 \leq \beta/2$. Therefore, for every radius $r \geq 2\beta$, the probability $\lambda_r = (\beta_0 - 1)/(r - 1)$ of leaving the reference state satisfies

$$\frac{\beta}{4r} \leq \lambda_r \leq \frac{\beta}{r}. \tag{55}$$

The sufficiently-small-ε condition in Theorem 5.2 is chosen below so that every radius used in the proof is at least 2β. All alternatives in each subclass are indexed by a vector of hidden actions, one for each decision state reached from the reference state.

H.3 Performance-Learning Hard Family

This family isolates the cost of performance learning. Every candidate action is stable, but one hidden action at each decision state gives a larger average reward. The proof has three steps. We construct the family, show that near-optimality forces the output to place substantial mass on the hidden actions, and then convert this requirement into a sample lower bound.

Step 1: Construct the hard family.

Choose $c_{1,\nu} \geq 1$, a universal $c_\Delta \geq 5184$, $c_{2,\nu} \leq 1/c_\Delta$, and $c_\nu > 0$ sufficiently small. Fix a radius $r \in [c_{1,\nu}\beta/\zeta, c_{2,\nu}\beta/(\zeta\varepsilon)]$ and let $m = \lfloor c_\nu c_W r^\nu \rfloor$, with constants chosen so that $m = \Theta_\nu(c_W r^\nu)$ and $m \geq 2$ on the displayed radius range. Use Lemma H.1 with the common normalization above. At each leaf x_i with $W(x_i) = r$, there are $k_{\mathrm{perf}} := |\mathcal{A}|$ candidate actions. One candidate is hidden. The hidden action exits to the reference state with probability $q-\Delta$ and self-loops with probability $1-q+\Delta$. Every other action exits with probability $q+\Delta$. Rewards are one on every leaf action. Choose $q = 8\zeta$ and $\Delta = c_\Delta \varepsilon \zeta^2 r/\beta$, and restrict $\varepsilon \leq c_{2,\nu}/c_{1,\nu}$. This makes the radius interval nonempty and ensures $0 < \Delta \leq \zeta$. All actions satisfy at least ζ drift because $1-(q-\Delta)(1-1/r) \leq 1-\zeta$, and every instance-dependent row has support two with $\alpha = 1$. Fix the known stabilizing policy at every performance leaf to candidate action 1. This is valid under every alternative because all leaf actions are ζ-stable. The lower bound does not restrict the algorithm to this architecture. For an arbitrary returned policy, define the hidden-action mass at leaf x_i directly as the probability assigned by $\widehat{\pi}(\cdot \mid x_i)$ to the hidden candidate.

Step 2: Convert near-optimality into hidden-action mass.

Let $\lambda_r = \Theta(\beta/r)$ be the total probability of entering the leaves from the reference state. For leaf i, let g_i denote the output policy's probability of selecting the hidden action at each visit, and set

$\bar{g} = m^{-1}\sum_i g_i$. The policy resamples its action after every self-loop, so the effective exit probability and residence time at leaf i are

$$q_i(g_i) = g_i(q-\Delta) + (1-g_i)(q+\Delta) = q + \Delta - 2\Delta g_i, \qquad h_i(g_i) = \frac{1}{q+\Delta-2\Delta g_i}.$$

Let $h_g = (q-\Delta)^{-1}$ be the pure hidden-action residence time and let $\bar{h} = m^{-1}\sum_i h_i(g_i)$. Since $q = 8\zeta$ and $0 < \Delta \le \zeta$,

$$h_g - h_i(g_i) = \frac{2\Delta(1-g_i)}{(q-\Delta)(q+\Delta-2\Delta g_i)} \ge \frac{2\Delta}{81\zeta^2}(1-g_i), \qquad h_g - \bar{h} \ge \frac{2\Delta}{81\zeta^2}(1-\bar{g}).$$

The cycle reward and length associated with an average leaf residence h are $G(h) = \lambda_r h$ and $T(h) = 1 + \lambda_r h$. Thus $F(h) := G(h)/T(h)$ has derivative $F'(h) = \lambda_r/[1+\lambda_r h]^2$. By Equation (55), $h_g \le 1/(7\zeta)$ and $r \ge c_{1,\nu}\beta/\zeta$ imply $\lambda_r h_g \le 1/7$, so $F'(h) \ge \lambda_r/2 \ge \beta/(8r)$ throughout the interval between $\bar{h}$ and h_g. The mean value theorem gives

$$\rho^\star_\zeta(M) - \rho_M(\widehat{\pi}) \ge \frac{\beta\Delta}{324 r\zeta^2}(1-\bar{g}) \ge 16\varepsilon(1-\bar{g}).$$

Thus ε-optimality forces $\bar{g} \ge 15/16$. Let $\boldsymbol{\theta} \in [k_{\mathrm{perf}}]^m$ collect the hidden-action identities, and let $\mathcal{E}_{\boldsymbol{\theta}}$ be the event that the algorithm's output satisfies the required stability margin and is ε-optimal under $\boldsymbol{\theta}$. By the lower-bound success requirement, $\mathbb{P}_{\boldsymbol{\theta}}(\mathcal{E}_{\boldsymbol{\theta}}) \ge 2/3$ for every $\boldsymbol{\theta}$. On this event, the previous display gives $m^{-1}\sum_i g_{i,\theta_i} \ge 15/16$, where g_{i,θ_i} is the output mass assigned to the hidden action at leaf i. Write $\mathbb{E}_{i,\theta}$ for expectation with $\theta_i = \theta$ and the remaining hidden actions averaged uniformly. Then the uniform prior on hidden actions gives

$$\frac{1}{m}\sum_{i=1}^{m}\frac{1}{k_{\mathrm{perf}}}\sum_{\theta=1}^{k_{\mathrm{perf}}}\mathbb{E}_{i,\theta}[g_{i,\theta}] \ge \frac{2}{3}\cdot\frac{15}{16} = \frac{5}{8}.$$

Step 3: Convert hidden-action mass into samples.

For the self-loop indicator, the hidden and non-hidden means are $u + d$ and $u - d$ with $u = 1 - q$ and $d = \Delta$. Thus Equation (53) gives $\lambda = 2q = 16\zeta$ and $e = \Delta/(2q) \le 1/16$. Write $N_{\mathrm{perf}}(r)$ for the worst-case expected number of queries on this radius-r subclass. Since $5/8 > 13/24$, Lemma H.2 across the m leaves gives

$$N_{\mathrm{perf}}(r) \ge c\, m|\mathcal{A}|\frac{\zeta}{\Delta^2} \ge c|\mathcal{A}|\, c_W \frac{\beta^2}{\varepsilon^2\zeta^3} r^{\nu-2}.$$

Writing N_{perf} for the resulting performance-learning lower bound, taking $r \asymp \beta/\zeta$ when $\nu < 2$, $r \asymp \beta/(\zeta\varepsilon)$ when $\nu > 2$, and either endpoint when $\nu = 2$ yields

$$N_{\mathrm{perf}} = \widetilde{\Omega}\left(\frac{|\mathcal{A}|\, c_W \beta^\nu}{\zeta^{\nu+1}\varepsilon^{\max\{2,\nu\}}}\right).$$

H.4 Stability-Learning Hard Family

This family isolates the cost of stability learning. The rewards favor the candidate actions, but among those candidates only one hidden action at each decision state satisfies the target drift margin. The proof again has three steps. We construct the family, use stability and near-optimality to force mass onto the hidden stable actions, and then apply the testing bound.

Step 1: Construct the hard family.

If $\alpha - 1 + \zeta < 16\kappa\zeta$, then the stability-learning rate is at most a universal constant multiple of the performance rate, because $\kappa > 1/16$ and $\varepsilon^{\max\{2,\nu\}-\nu} \le 1$. In this regime the performance subclass already lower-bounds the sum up to constants. It remains to consider $\alpha - 1 + \zeta \ge 16\kappa\zeta$.

Choose $c'_{R,\nu} \le 2^{-12}$ and set $R_s = c'_{R,\nu}\beta/(\zeta\varepsilon)$ and $m = \lfloor c_\nu c_W R_s^\nu \rfloor$. Restricting the theorem's sufficiently-small-ε regime further to $\varepsilon \le c'_{R,\nu}$ gives $R_s \ge \beta/\zeta \ge 2\beta$ and, after increasing C_ν in the nondegeneracy condition, $m = \Theta_\nu(c_W R_s^\nu) \ge 2$. Use Lemma H.1 with the common reference-row

normalization to transition directly to the m decision states, taking their potential value to be R_s. Let $y_- = 1/2$ and $y_+ = \alpha$. At each decision state x_i, there are a known stabilizing action and $k_{\text{stab}} := |\mathcal{A}| - 1$ candidate actions. The stabilizing action and one hidden candidate have normalized next-W mean $1 - \zeta$, while every other candidate has normalized mean $1 - \zeta + 8\kappa\zeta$. The candidate transition distributions assign potential values $y_- R_s$ and $y_+ R_s$ with the matching Bernoulli probabilities. The hard-regime condition makes the bad mean at most α, so these probabilities are valid and the one-step movement bound holds.

The two candidate next states are common branch states z_- and z_+ with potential values $y_- R_s$ and $y_+ R_s$. At either state all primitive actions have the same row and reward: they give reward one, self-loop with probability $1 - q_0$, and return to the reference state with probability $q_0 = 8\zeta$. The stabilizing action at x_i enters a third common state z_0 with $W(z_0) = (1-\zeta)R_s$. All actions there have the same duration distribution and give reward zero. Every branch state has $W(z) = yR_s$ for some $y \geq 1/2$, so its normalized next-state moment is $1 - q_0 + q_0/(yR_s) \leq 1 - \zeta$ whenever $R_s \geq 16/7$. This condition follows from the preceding parameter choice. The three common branch states add only a constant to every core volume.

Step 2: Convert the output guarantees into hidden-action mass.

For one decision state, let u_i be the total output mass on bad candidates, g_i the mass on the hidden good candidate, and $c_i = g_i + u_i$ the total candidate mass. The required output margin $(1-\kappa)\zeta$ gives $1 - \zeta + 8\kappa\zeta u_i \leq 1 - \zeta + \kappa\zeta$, hence $u_i \leq 1/8$. Let $\bar{c} = m^{-1} \sum_i c_i$ and let $\pi^\star$ be the comparator that always chooses the hidden good candidate. Write λ_{R_s} for the probability of transitioning from the reference state to radius R_s and $h_0 = 1/q_0$ for the mean branch residence time. One cycle under $\pi^\star$ has expected reward $\lambda_{R_s} h_0$ and expected length $1 + \lambda_{R_s}(1 + h_0)$, so

$$\rho_M(\pi^\star) = \frac{\lambda_{R_s} h_0}{1 + \lambda_{R_s}(1 + h_0)} \geq \frac{\varepsilon}{64 c'_{R,\nu}} \geq 64\varepsilon.$$

Indeed, Equation (55) gives $\lambda_{R_s} h_0 \geq \varepsilon/(32 c'_{R,\nu})$, while $\varepsilon \leq c'_{R,\nu}$ and $\zeta \leq 1/64$ give $1 + \lambda_{R_s}(1 + h_0) < 2$. The reference-state transition law and all branch durations are the same under $\pi^\star$ and the agent's output. Candidate branches earn reward one and the stabilizing branch earns reward zero. Hence the output earns average reward $\bar{c}\,\rho_M(\pi^\star)$ on this subclass. If the output is ε-optimal, then $(1 - \bar{c})\rho_M(\pi^\star) \leq \varepsilon$, so $\bar{c} \geq 15/16$. Therefore $m^{-1} \sum_i g_i \geq 15/16 - 1/8 = 13/16$.

Let $\boldsymbol{\theta} \in [k_{\text{stab}}]^m$ collect the hidden good-candidate identities, and let $\mathcal{E}_{\boldsymbol{\theta}}$ be the event that the output satisfies the required stability margin and is ε-optimal under $\boldsymbol{\theta}$. The success requirement gives $\mathbb{P}_{\boldsymbol{\theta}}(\mathcal{E}_{\boldsymbol{\theta}}) \geq 2/3$ for every $\boldsymbol{\theta}$, and on this event the preceding lower bound holds. With $\mathbb{E}_{i,\theta}$ defined analogously for the stability tasks, the uniform prior on hidden good candidates gives

$$\frac{1}{m} \sum_{i=1}^{m} \frac{1}{k_{\text{stab}}} \sum_{\theta=1}^{k_{\text{stab}}} \mathbb{E}_{i,\theta}[g_{i,\theta}] \geq \frac{2}{3} \cdot \frac{13}{16} = \frac{13}{24}.$$

Step 3: Convert hidden-action mass into samples.

Let $D = \alpha - 1/2$. For the indicator of the low-potential next state z_-, the hidden-good and bad candidate means are $u_s + d_s$ and $u_s - d_s$, where $u_s = (\alpha - 1 + \zeta - 4\kappa\zeta)/D$ and $d_s = 4\kappa\zeta/D$. With $\lambda_s = 2\min\{u_s, 1 - u_s\}$ and $e_s = d_s/\lambda_s$, the hard-regime condition gives $e_s \leq 1/6$ and, since $D \in [1/2, 3/2]$, $\lambda_s = \Theta(\alpha - 1 + \zeta)$. Therefore $1/(\lambda_s e_s^2) = \Theta((\alpha - 1 + \zeta)/(\kappa^2\zeta^2))$. Writing N_{stab} for the worst-case expected number of queries on this subclass, Lemma H.2 gives

$$N_{\text{stab}} = \widetilde{\Omega}\left(\frac{|\mathcal{A}|\, c_W \beta^\nu (\alpha - 1 + \zeta)}{\kappa^2 \zeta^{\nu+2} \varepsilon^\nu} \right).$$

H.5 Combining the Bounds

It remains to verify that both hard families belong to the model class and to combine their two lower-bound terms.

Unspecified rows have zero reward and return deterministically to the reference state. Every alternative-dependent row has support at most two. The common reference-state transition row has

moment $\beta_0 \le \beta$ and may have support $m+1$, but it is identical across alternatives and may be revealed to the agent. Choose $c_\nu \le 2^{-(\nu+3)}$ and then choose C_ν in Theorem 5.2 sufficiently large that every core at the first non-reference potential level has room for the reference state and the three common branch states. In the performance subclass $m \le c_\nu c_W r^\nu$, and in the stability subclass $m \le c_\nu c_W R_s^\nu$. These choices give $m \ge 2$ and $|\mathcal{K}_t| \le c_W t^\nu$ at every radius t: below the hard-state level only constantly many common states appear, and at or above that level the m hard states use at most a fixed fraction of the volume budget. Thus the finite construction satisfies the countability, enumerability, reward, movement, stabilizing-policy, and nonemptiness requirements.

Both reductions depend only on the returned action probabilities at the hard decision states and therefore apply to arbitrary outputs. The performance subclass gives the first term. If $\alpha - 1 + \zeta \ge 16\kappa\zeta$, the stability subclass gives the second, and the maximum of the two bounds is at least half their sum. Otherwise the second term is at most a universal constant times the first. This proves Theorem 5.2.□

I Extensions and Modeling Examples

The main theory uses a singleton reference state. We first record the corresponding drift guarantee for a reference set, which is useful for applications with a stable operating region. We then give concrete examples of the potential function, bounded one-step movement condition, and known stabilizing policy.

I.1 Reference-Set Drift Guarantee

Proposition I.1 (Reference-set drift guarantee)**.** *Fix $\eta \in (0,1)$ and let $C \subseteq K \subset \mathcal{S}$, where C is a designated reference set and K is a finite set of learned states. Suppose a policy π uses empirically feasible action mixtures on $K \setminus C$ and the known stabilizing policy outside K, where it satisfies the true η-drift inequality. On the simultaneous upper-confidence event of Lemma D.1,*

$$\mathbb{E}_M^\pi[W(S_{t+1}) \mid S_t = s] \le (1-\eta)W(s), \qquad \forall s \notin C,$$
$$\mathbb{P}_x^\pi(\tau_C > t) \le \min\{1, W(x)(1-\eta)^t\}, \qquad \tau_C := \inf\{t \ge 0 : S_t \in C\}.$$

Proof. For states in $K \setminus C$, Lemma D.1 transfers empirical feasibility to the true η-drift inequality. Outside K, the same inequality holds by the stabilizing-policy assumption. Applying this drift inequality to the process killed upon entering C gives $\mathbb{E}_x[W(S_t)\mathbf{1}\{\tau_C > t\}] \le (1-\eta)^t W(x)$ by induction. Since $W \ge 1$, Markov's inequality gives the return-time bound. □

I.2 Modeling Examples

Queueing and data-center scheduling. Let Q_t be a single-server queue. At $q \ge 1$, action a selects a service mode and $Q_{t+1} = q - D_t^a + A_t$, where $A_t \sim$ Bernoulli(λ) and $D_t^a \sim$ Bernoulli(μ_a) are conditionally independent. Taking the empty queue as the reference state and choosing a parameter $\chi > 1$ with $W(q) = \chi^q$, the one-step drift ratio under action a is $(1 - \lambda + \lambda\chi)(1 - \mu_a + \mu_a/\chi)$. Hence a high-service mode with $\mu_{\text{stab}} > \lambda$ is stabilizing for χ close enough to one and defines a valid stabilizing policy. Since $Q_{t+1} \in \{q-1, q, q+1\}$ away from the empty queue, bounded one-step movement holds with $\alpha = \chi$. The reference-row moment is $\beta = 1 - \lambda + \lambda\chi$. The core $\{q : \chi^q \le R\}$ has size $O(\log R)$. A pooled datacenter workload scheduler gives the same structure with Q_t interpreted as backlog or workload and actions interpreted as server-capacity choices. Finite per-period arrivals and service capacity give bounded movement.

Robotics stabilization. After discretizing tracking error, balance error, or deviation from a nominal operating point, the state space can be modeled by countable error shells. The potential function can be chosen as a monotone function of this deviation, such as a quadratic or exponential potential on the shell index. The known stabilizing policy can be implemented by a recovery or balance controller. The drift condition states that the closed-loop controller reduces the expected deviation outside the reference region, while bounded movement follows from finite sampling time, bounded velocities, and actuator limits.

Thermal and energy control. For thermal, storage, or inventory-control systems, the state can record deviation from a desired operating band, stored-energy imbalance, or unmet demand. A natural potential function grows with this deviation. Actions choose heating, cooling, charging, discharging, or shedding levels. A conservative controller that pushes the system back toward the operating band provides a known stabilizing policy. Bounded movement is natural when one-period disturbances, thermal changes, and charging or discharging rates are finite.

J Experimental Details

Purpose. The controlled experiments test the two terms in the sample-complexity bound, their joint rate, and the complete algorithm on a multi-step MDP. This section specifies the synthetic transition and reward laws, exact sample accounting, the treatment of core exits and boundaries, and the numerical accuracy of the planning solvers. It then documents the information available to each voltage controller.

Persistent multi-shell construction. The phase-transition experiment has a singleton reference state and twenty dyadic shells with radii $r_j = 2^j$. For $\nu \in \{1, 2, 3\}$, shell j contains $\lceil r_j^\nu - r_{j-1}^\nu \rceil$ states, so a radius-R core contains exactly R^ν states. The fixed reference action enters shell j uniformly with total probability $0.4/r_j$ and otherwise remains at the reference state. Each shell row has two next states: it either self-loops or returns to the reference state. The performance diagnostic uses equal probabilities for these events. The stability diagnostic chooses the self-loop probability to realize the normalized moments specified below. The construction is therefore multi-step, every learned transition row has support two, and its reference moment is $\beta = 8.6000$. We use radii $R \in \{2^4, \dots, 2^{16}\}$ and target accuracy $\varepsilon = 0.07/R$, leaving at least four outer shells beyond every tested core.

The performance diagnostic supplies the exact drift and hides two Bernoulli reward means under an independent action-label permutation at every core state. The reward gap in shell j is chosen so that selecting the worse action throughout any one shell contributes $2\varepsilon/D$, where $D = \log_2 R$. A fixed 0.1 reward gap outside the core supplies the truncation error. The agent selects actions from empirical rewards, using a common number of samples within each shell and different numbers across shells. The stability diagnostic instead supplies exact rewards and hides the transition moments. Its stable candidate has normalized moment $1 - \zeta = 0.8$, while its higher-reward unsafe candidate has moment $1 - \zeta + 1.25\kappa\zeta = 0.925$, with $\kappa = 0.5$. A next-state sample produces normalized potential $1/r_j$ or 1, and feasibility is decided by the same empirical-Bernstein upper-confidence rule used by LyCore. The known stabilizing action is selected whenever neither candidate satisfies the empirical drift constraint.

For each (ν, R), we search fixed geometric grids of model and drift sample allocations. Performance success requires average-reward error at most ε, stability success requires both the output drift condition and reward error at most ε, and joint success requires both events on paired independent trials. Because states within a shell are exchangeable, we simulate their binomial and multinomial sufficient statistics. This is distributionally identical to explicit row-wise sampling, and all reported budgets count the underlying samples. We record the smallest total budget attaining 90% empirical success. Means and standard errors use 32 independent batches of 240 trials. Log–log fits over the four smallest-ε points give performance slopes 2.11, 2.23, and 3.16, stability slopes 1.07, 2.08, and 3.09, and joint slopes 2.14, 2.13, and 3.10 for $\nu = 1, 2, 3$, respectively.

Stochastic birth–death construction. The joint experiment uses states $q \in \{0, \dots, 80\}$, Bernoulli arrivals with probability 0.48, and service actions with probabilities 0.59, 0.70, and 0.88. At $q = 0$, the fixed reference action moves to 1 on an arrival and otherwise remains at 0. At $q = 80$, an upward move is clipped at 80, giving a finite reflecting boundary. At all other states, independent arrival and service events produce upward moves, downward moves, and self-loops. Each generative call returns this next state and an independent Bernoulli reward with mean $[0.96 - 0.22 \min\{q, 30\}/30 - c_a]_{[0.02,0.98]}$, where $(c_1, c_2, c_3) = (0.01, 0.055, 0.17)$.

We use $W(q) = 1.08^q$, learn on $q \le 28$, and follow the stabilizing action with service probability 0.88 outside the core. The target and output margins are 0.012 and 0.0072. For each displayed n, the reference row and every (q, a) with $1 \le q \le 28$ receive n independent model samples and

n independent drift samples. Thus LyCore and the plug-in ablation use $2n(1 + 28 \times 3)$ samples. Unconstrained planning uses one model batch of size $2n$ on the same rows. A model sample above 28 is projected to 0 only in the finite planning model. The deployed policy instead uses the stabilizing action until the process re-enters the core. The stabilizing action enters planning through its known drift bound $1 - \zeta$, not its unknown exact moment.

The in-core policy is computed by the occupation-measure LP in (1) using HiGHS. Across all 3,601 solves, the maximum flow residual is 1.0×10^{-7}, the maximum drift-constraint violation is 6.2×10^{-8}, and the maximum solver count is 72 iterations. Median solve times range from 0.42 to 0.61 ms. The stable benchmark is computed by the same LP on the true 81-state model. We report the average-reward gap to this benchmark and the percentage of outputs violating the true drift condition. Every point averages 120 independent runs.

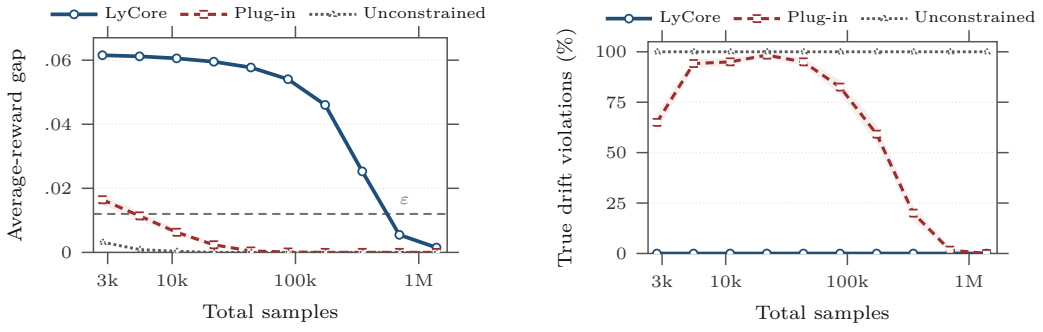
At 348,160 samples, LyCore remains stable with reward gap 0.0253, while plug-in violates drift in 20% of runs. At 696,320 samples, LyCore reaches gap 0.0055 without violations. Plug-in still violates drift in 1.7% of runs, and unconstrained planning remains unstable in every run.

Distribution-grid voltage-control model. The simulator uses the Baran–Wu 33-bus radial distribution system and the line and load data in MATPOWER `case33bw` (Baran & Wu, 1989; MATPOWER, 2016). It has one slack bus, 32 PQ buses, and 32 radial branches. At every state-action pair, we solve the balanced nonlinear AC equations, where P_i and Q_i are the active and reactive power injections at bus i, V_i and θ_i are its voltage magnitude and angle, and $Y = G + \mathrm{i}B$ is the bus-admittance matrix with conductance matrix G and susceptance matrix B:

$$P_i = \sum_j V_i V_j \big(G_{ij}\cos(\theta_i - \theta_j) + B_{ij}\sin(\theta_i - \theta_j)\big)\,,$$
$$Q_i = \sum_j V_i V_j \big(G_{ij}\sin(\theta_i - \theta_j) - B_{ij}\cos(\theta_i - \theta_j)\big)\,.$$

An independent Newton power-flow check of the nominal case gives a minimum voltage of 0.913090 p.u. at bus 18 and active loss of 202.677 kW. The residual of the implemented equations is below 1.3×10^{-11} p.u.

Sequential state and data. PV generation is placed at buses 6, 14, 18, 25, 30, and 33, with reactive-power control at buses 18, 25, 30, and 33. Nine actions encode spatial reactive-power patterns. The state records a seven-level voltage-deviation bin, one of six weather regimes, a five-level inverter thermal state, and the previous action, giving 1,890 states. Weather regimes are constructed from hourly 2023 Fresno temperature and solar-radiation data from the Open-Meteo Historical Weather API (Open-Meteo, 2026). One regime is reserved for the highest 2.5% of net-load hours so rare stress events are not averaged away. The transition includes weather persistence, inverter response



(a) Reward optimality.

(b) True-model stability.

Figure 3: Complete-algorithm evaluation on the stochastic birth–death MDP. The panels report reward suboptimality and true drift violations at equal total-sample budgets. Shading is one standard error.

lag, action-dependent heating, thermal derating, and switching costs. Rewards in $[0, 1]$ combine voltage quality, feeder loss, reactive effort, switching, thermal stress, and curtailment.

The potential is $W(s) = \exp(\max\{0, |b(s)| - 1\})$, where $b(s) \in \{-3, \dots, 3\}$ is the voltage bin. Bins $|b| \le 1$ cover the normal and warning bands, $|b| = 2$ denotes a standard-limit violation, and $|b| = 3$ a severe excursion. The experimental drift condition is imposed on violation states $|b| \ge 2$ and measures contraction toward the operating set $|b| \le 1$ according to Proposition I.1. The learned core contains the 1,350 states with $|b| \le 2$, while severe states use the known stabilizing policy. We set the target drift margin to 0.03 and enforce an output margin of 0.015. The reference-set result provides the reported drift guarantee. The matching minimax theorems are established for the singleton-reference setting.

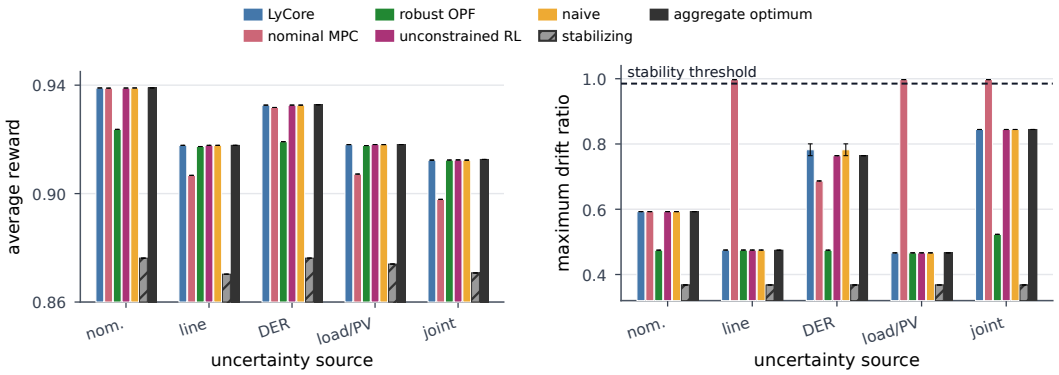
Uncertainty suite. The line-mismatch plant increases resistance by 10%, reactance by 8%, and load by 4%. The DER plant reduces reactive response to 76% and increases action-induced heating by 24%. The operating-shift plant raises load by 10%, PV by 12%, and weather-regime persistence. The joint plant combines line, DER, and operating shifts. These parameters and the scenario label are hidden from the learning methods. The robust AC-OPF baseline is the exception: it receives all five physical plants and optimizes its one-step action against their worst case.

Voltage policies and evaluation. The known stabilizing policy is a fixed conservative reactive-support controller and is not optimized using any plant model. Nominal AC-OPF enumerates the nine actions using the nominal AC equations. Nominal AC-MPC applies six-step dynamic programming to the nominal aggregate model. Empirical AC-MPC estimates a model from generative samples and then uses the same six-step horizon. The unconstrained model-based RL baseline uses the projected core MDP and average-reward solver of LyCore but removes the drift constraint. At an equal total-sample budget, it assigns all samples to model estimation. The naive plug-in uses independent drift and model batches but replaces upper confidence bounds by empirical means. For each plant scenario, all three learning methods are trained and evaluated on that plant. Only the fixed nominal policies are transferred across plants. The full-information optimal stable policy is computed from the plant's complete aggregate kernel by solving the constrained average-reward problem on all 1,890 states.

For learned benchmarks, LyCore and the plug-in ablation use independent model and drift batches of 8,192 samples per core state-action pair. Empirical AC-MPC and unconstrained RL instead use one model batch of 16,384 samples per pair. Since the core has 1,350 states and nine actions, every learned method therefore uses 199,065,600 total samples per replicate. All results use ten replicates. The confidence level is $\delta = 0.05$. For a state-action pair with n normalized-potential observations, empirical variance $\widehat{v}$, and observation range bounded by b, the confidence radius is $\sqrt{2\widehat{v}\log(4|\mathcal{K}_R||\mathcal{A}|/\delta)/n} + 3b\log(4|\mathcal{K}_R||\mathcal{A}|/\delta)/n$. Average-reward policies are computed by relative value iteration with span tolerance 10^{-9} and an iteration cap of 2,000. Every drift-feasible solve reaches the tolerance. The largest final span residual is 9.9972×10^{-10}, the maximum iteration count is 713, and median solve times range from 0.17 to 1.20 seconds across planners. Infeasible low-budget constrained problems return the known stabilizing policy. We evaluate stationary metrics from each induced transition matrix and additionally report four year-long stress-test rollouts driven by the original, unaggregated weather sequence. The formal drift guarantee applies to the aggregate MDP used for learning. These rollouts use the 2023 weather record at hourly resolution. Their standard errors reflect thermal-transition randomness rather than variation across weather years.

Table 1: Year-long closed-loop evaluation under joint uncertainty and the unaggregated 2023 weather sequence. The aggregate optimum is the full-information optimum of the aggregate training MDP evaluated under the same replay. Entries are means with standard errors over four stochastic thermal-transition rollouts.

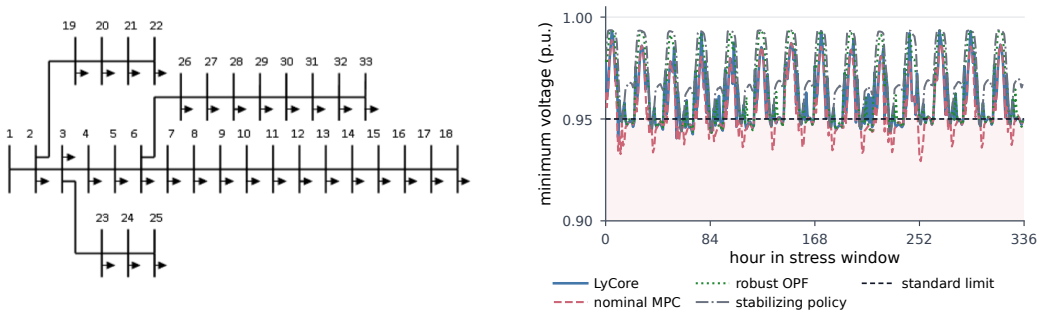
Controller	Average reward	Violation (%)	P99 excursion (p.u.)
LyCore	0.87366 ± 0.00012	26.26 ± 0.06	0.00988 ± 0.00007
Nominal AC-MPC	0.85687 ± 0.00017	42.67 ± 0.05	0.01777 ± 0.00005
Robust AC-OPF	0.87618 ± 0.00015	25.05 ± 0.08	0.00818 ± 0.00004
Stabilizing policy	0.86696 ± 0.00000	0.98 ± 0.00	0.00000 ± 0.00000
Aggregate optimum	0.87236 ± 0.00007	26.91 ± 0.05	0.01185 ± 0.00000



(a) Average reward.

(b) True-model drift ratio.

Figure 4: Full-AC voltage-control benchmark across uncertain plants. The dashed line in (b) is the output stability threshold, and lower drift ratios indicate stronger contraction. LyCore approaches the aggregate-MDP optimum while satisfying the drift constraint. The stabilizing policy is conservative, and the same-planner ablations isolate the effect of drift constraints and confidence bounds.



(a) Baran–Wu 33-bus feeder.

(b) Two-week voltage trace.

Figure 5: Supporting voltage-control visuals. The left panel shows the Baran–Wu 33-bus feeder, and the right panel reports a chronological stress window under the joint-uncertainty plant.