TOWARDS A THEORETICAL UNDERSTANDING OF HOW MUTUAL DISTILLATION IMPROVES GENERALIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

We argue that mutual distillation between reinforcement learning policies serves as *implicit regularization* that prevents them from overfitting to irrelevant features. Theoretically, we establish the first trust region lower bound connecting generalization performance with policy robustness, confirming the long-standing assumption that policy robustness underpins generalization. We further characterize the missing connection between deep mutual learning (DML) and this lower bound. Empirically, we demonstrate that mutual distillation between policies promotes such robustness, enabling the spontaneous emergence of *invariant representations* from visual inputs. Ultimately, rather than emphasizing extensive baseline comparisons, our goal is to uncover the underlying principles of generalization and provide theoretical insights into its mechanisms as our foundational contributions.

1 INTRODUCTION

Humans can learn robustly and generalize across diverse environments. Once acquired, a skill often transfers to new contexts that share the same underlying structure, even when their visual appearance differs substantially. For instance, a person proficient at a video game can still perform well when the background graphics or character textures change. This flexibility suggests that human learning is not narrowly tied to specific sensory inputs but instead relies on internal representations that abstract away irrelevant details while preserving task-relevant structure. Neuroscientific studies support this view by linking abstract reasoning to the prefrontal cortex (Bengtsson et al., 2009; Dumontheil, 2014) and the activity of inhibitory neurons to more efficient cognitive processing (Pi et al., 2013).

By contrast, visual reinforcement learning (VRL) agents often struggle to generalize. Although they can learn to solve complex tasks in specific environments, even minor changes, such as shifts in color schemes or background textures, can significantly degrade their performance. This sensitivity suggests that VRL agents tend to overfit to superficial visual features rather than capture the task's underlying structure (Cobbe et al., 2019; 2020). These limitations raise two fundamental questions:

What hinders reinforcement learning agents from generalizing like humans? How can we enable them to learn robust representations that drive human-like generalization behavior?

The limited generalization ability of VRL agents primarily stems from their reliance on convolutional neural networks (CNNs) as visual encoders. Although CNNs are the de facto standard for processing high-dimensional inputs, they are notoriously sensitive to even minor changes (Goodfellow et al., 2014). This vulnerability undermines the robustness of learned policies and limits their ability to generalize. A common mitigation strategy is data augmentation (Shorten & Khoshgoftaar, 2019), which improves robustness by diversifying the training distribution and reducing dataset-induced biases. Alternatively, invariant representation learning addresses generalization from a feature-learning perspective. It seeks representations that remain stable across a wide range of input transformations, thereby promoting both robustness and transferability (Nguyen et al., 2021).

Although data augmentation is an effective bias-mitigation technique, its reliance on task-specific strategies manually designed by experts complicates task-independent solutions. Our method instead enables agents to generalize without augmentations or external priors, relying solely on training experience. Invariant representation learning can likewise improve cross-domain generalization, but it requires transformation correspondences that are fundamentally unavailable in dynamic rein-

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.

forcement learning environments. Moreover, such frameworks separate the encoder from the model and complicate theoretical analysis. Our framework remains end-to-end in theory and practice.

We develop a theory of RL generalization showing that robustness to irrelevant features improves performance. We then provide a limitation-aware connection from DML (Zhang et al., 2018b) to this bound: same-view DML transfers peer robustness through a square-root KL term, while a same-view update analysis identifies explicit robustness-improvement conditions. Experiments indicate that DML mitigates overfitting to irrelevant features in online policies, stabilizes learning, and improves generalization.

The main contributions of this paper are as follows:

- *We establish a quantitative connection between policy robustness and generalization in visual reinforcement learning. Our trust-region analysis quantifies the effect of rendering sensitivity and yields a refined performance lower bound.*
- *We identify cancellation of complementary rendering errors as a robustness mechanism of mutual distillation. We characterize its effect under ideal same-view updates and establish sufficient conditions for robustness improvement under approximate updates.*
- *We demonstrate that MDPO improves generalization and visual robustness on Procgen. Encoder-transfer experiments and resource-scaling ablations support a representation-level explanation of these gains beyond increased training resources alone.*

2 Preliminaries

2.1 Markov Decision Process and Generalization

A Markov decision process (MDP) is a mathematical framework for sequential decision-making. It is defined by a tuple $\mathcal{M} = (\mathcal{S}, \mathcal{A}, r, \mathcal{P}, \rho, \gamma)$, where $\mathcal{S}$ and $\mathcal{A}$ denote the state and action spaces, $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is the reward function, $\mathcal{P} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ is the transition kernel, $\rho : \mathcal{S} \to [0, 1]$ is the initial-state distribution, and $\gamma \in (0, 1)$ is the discount factor. We assume that rewards and all extrema used below are finite; otherwise, the corresponding maxima are understood as suprema.

For a stochastic policy $\mu : \mathcal{S} \to \Delta(\mathcal{A})$, the action-value and value functions are defined as

$$Q^\mu(s_t, a_t) = \mathbb{E}_\mu \left[\sum_{k=0}^{\infty} \gamma^k r(s_{t+k}, a_{t+k}) \mid s_t, a_t \right], \quad V^\mu(s_t) = \mathbb{E}_{a_t \sim \mu(\cdot|s_t)} \left[Q^\mu(s_t, a_t) \right]. \tag{1}$$

Given Q^μ and V^μ, the advantage function can be expressed as $A^\mu(s_t, a_t) = Q^\mu(s_t, a_t) - V^\mu(s_t)$.

In our generalization setting, we use a rendering function $f : \mathcal{S} \to \mathcal{O}_f \subset \mathcal{O}$ to obfuscate the agent's actual observations. This function is a *bijection*[1] from $\mathcal{S}$ to $\mathcal{O}_f$. Whereas Smallwood & Sondik (1973) introduce the POMDP framework, our f does not mask the underlying state information but instead perturbs its style. We denote the MDP induced by the underlying MDP $\mathcal{M}$ and rendering function f as $\mathcal{M}_f = (\mathcal{O}_f, \mathcal{A}, r_f, \mathcal{P}_f, \rho_f, \gamma)$, where $\mathcal{O}_f$ is the observation space, $r_f : \mathcal{O}_f \times \mathcal{A} \to \mathbb{R}$ is the reward function, $\mathcal{P}_f : \mathcal{O}_f \times \mathcal{A} \times \mathcal{O}_f \to [0, 1]$ is the dynamics, and $\rho_f : \mathcal{O}_f \to [0, 1]$ is the initial observation distribution. We make the following assumptions:

Assumption 2.1. Assume that f can be sampled from a distribution $p : \mathcal{F} \to [0, 1]$, where $f \in \mathcal{F}$. This means that $\int_{\mathcal{F}} p(f) \mathrm{d}f = 1$ is naturally satisfied.

Assumption 2.2. Given any $f \in \mathcal{F}, o_0^f, o_t^f, o_{t+1}^f \in \mathcal{O}_f$, assume that $r_f(o_t^f, a_t) = r(f^{-1}(o_t^f), a_t)$, $\mathcal{P}_f(o_{t+1}^f \mid o_t^f, a_t) = \mathcal{P}(f^{-1}(o_{t+1}^f) \mid f^{-1}(o_t^f), a_t)$ and $\rho_f(o_0^f) = \rho(f^{-1}(o_0^f))$.

Remark 2.3. Assumption 2.2 states that all $\mathcal{M}_f$ share a common underlying MDP $\mathcal{M}$. Different rendering functions perturb the agent's observations while leaving all other components unchanged, much like different painters depicting the same scene in their own styles.

An agent following $\pi : \mathcal{O} \times \mathcal{A} \to [0, 1]$ in $\mathcal{M}_f$ generates the trajectory

$$\tau_f = (o_0^f, a_0, r_0^f, o_1^f, a_1, r_1^f, \ldots, o_t^f, a_t, r_t^f, \ldots), \tag{2}$$

[1] We define $\mathcal{O}_f := \{f(s) \mid s \in \mathcal{S}\}$, which means for any $s_1 \neq s_2$, we have $f(s_1) \neq f(s_2)$.

where $o_0^f \sim \rho_f(\cdot)$, $a_t \sim \pi(\cdot \mid o_t^f)$, $r_t^f = r_f(o_t^f, a_t)$, and $o_{t+1}^f \sim \mathcal{P}_f(\cdot \mid o_t^f, a_t)$. We abbreviate this process as $\tau_f \sim \pi$. During training, the agent can access only a subset of the MDPs, $\{\mathcal{M}_f \mid f \in \mathcal{F}_{\text{train}} \subset \mathcal{F}\}$, and its generalization performance is then evaluated across all MDPs. Let $p_{\text{train}} : \mathcal{F}_{\text{train}} \to [0, 1]$ denote the distribution over $\mathcal{F}_{\text{train}}$. The agent's training performance $\eta(\pi)$ and generalization performance $\zeta(\pi)$ are

$$\eta(\pi) = \mathbb{E}_{f \sim p_{\text{train}}(\cdot), \tau_f \sim \pi} \left[\sum_{t=0}^{\infty} \gamma^t r_f(o_t^f, a_t) \right], \quad \zeta(\pi) = \mathbb{E}_{f \sim p(\cdot), \tau_f \sim \pi} \left[\sum_{t=0}^{\infty} \gamma^t r_f(o_t^f, a_t) \right]. \tag{3}$$

The goal of the agent is to learn a policy π that maximizes the generalization performance $\zeta(\pi)$.

2.2 Deep Mutual Learning

Deep mutual learning (DML) (Zhang et al., 2018b) is a mutual distillation technique in supervised learning. Unlike the traditional teacher-student distillation strategy, DML aligns the probability distributions of multiple student networks by minimizing the KL divergence loss during training, allowing them to learn from each other. Specifically,

$$\mathcal{L}_{\text{DML}} = \mathcal{L}_{\text{SL}} + \alpha \mathcal{L}_{\text{KL}}, \tag{4}$$

where $\mathcal{L}_{\text{SL}}$ and $\mathcal{L}_{\text{KL}}$ represent the supervised learning loss and the KL divergence loss, respectively.

3 Theoretical Analysis

3.1 General Bound

This section presents our main results, showing that improving an agent's robustness to irrelevant features improves its generalization performance.

A key issue is that the probability distribution p_{train} is not known exactly. Because $\mathcal{F}_{\text{train}}$ is a subset of $\mathcal{F}$, we assume that p_{train} is obtained by normalizing the probability distribution p:

Assumption 3.1. Let $\mathcal{F}_{\text{eval}} := \mathcal{F} \setminus \mathcal{F}_{\text{train}}$ and $Z := \int_{\mathcal{F}_{\text{train}}} p(f)\, \mathrm{d}f \in (0, 1)$. For any $f \in \mathcal{F}$, assume that

$$p_{\text{train}}(f) = \frac{p(f)\, \mathbb{I}\{f \in \mathcal{F}_{\text{train}}\}}{Z}, \quad p_{\text{eval}}(f) = \frac{p(f)\, \mathbb{I}\{f \in \mathcal{F}_{\text{eval}}\}}{1 - Z}, \tag{5}$$

where Z and $1 - Z$ are the corresponding normalization constants and $\mathbb{I}\{\cdot\}$ denotes the indicator function.

For a given policy π, restricting attention to its interaction with $\mathcal{M}_f$ establishes a bijection between this policy and an underlying policy that interacts directly with $\mathcal{M}$. We denote the latter by $\mu_f(\cdot \mid s_t) = \pi(\cdot \mid f(s_t))$. After defining the normalized discounted visitation distribution $d^\mu(s) = (1 - \gamma) \sum_{t=0}^{\infty} \gamma^t \mathbb{P}(s_t = s \mid \mu)$, we can express the training and generalization performance of π in terms of the underlying policy μ_f. Specifically, the following relation holds:

Lemma 3.2. *For any policy π, define its underlying policy by $\mu_f(\cdot \mid s_t) = \pi(\cdot \mid f(s_t))$. Then*

$$\eta(\pi) = \frac{1}{1 - \gamma} \mathbb{E}_{\substack{f \sim p_{\text{train}}(\cdot) \\ s \sim d^{\mu_f}(\cdot) \\ a \sim \mu_f(\cdot \mid s)}} [r(s, a)], \quad \zeta(\pi) = \frac{1}{1 - \gamma} \mathbb{E}_{\substack{f \sim p(\cdot) \\ s \sim d^{\mu_f}(\cdot) \\ a \sim \mu_f(\cdot \mid s)}} [r(s, a)]. \tag{6}$$

Proof. See Appendix E.1. □

We can therefore analyze generalization through the underlying policy μ_f. Define $L_\pi(\tilde{\pi}) = \eta(\pi) + \frac{1}{1-\gamma} \mathbb{E}_{f \sim p_{\text{train}}(\cdot), s \sim d^{\mu_f}(\cdot), a \sim \tilde{\mu}_f(\cdot \mid s)} [A^{\mu_f}(s, a)]$ as the first-order approximation of η (Schulman et al., 2015). We can then derive the following lower bounds:

Theorem 3.3 (Training performance lower bound)**.** *Given any two policies, $\tilde{\pi}$ and π, the following bound holds:*

$$\eta(\tilde{\pi}) \geq L_\pi(\tilde{\pi}) - \frac{2\gamma \epsilon_{\text{train}}}{(1 - \gamma)^2} \mathbb{E}_{\substack{f \sim p_{\text{train}}(\cdot) \\ s \sim d^{\mu_f}(\cdot)}} [D_{\text{TV}}(\tilde{\mu}_f, \mu_f)[s]], \tag{7}$$

where $\epsilon_{\text{train}} = \max_{f \in \mathcal{F}_{\text{train}}} \left\{ \max_s \left| \mathbb{E}_{a \sim \tilde{\mu}_f(\cdot \mid s)} [A^{\mu_f}(s, a)] \right| \right\}$.

Proof. See Appendix E.3. □

Theorem 3.4 (Generalization performance lower bound). *For any policies $\tilde{\pi}$ and π:*

$$\begin{aligned}\zeta(\tilde{\pi}) \geq L_\pi(\tilde{\pi}) &- \frac{2r_{\max}(1-Z)}{1-\gamma} - \frac{2\gamma\epsilon_{\text{train}}}{(1-\gamma)^2}\mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot)\\ s\sim d^{\mu_f}(\cdot)}}\left[D_{\text{TV}}(\tilde{\mu}_f,\mu_f)[s]\right] \\ &- \frac{2\delta_{\text{train}}(1-Z)}{1-\gamma}\mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot)\\ s\sim d^{\tilde{\mu}_f}(\cdot)}}\left[D_{\text{TV}}(\tilde{\mu}_f,\mu_f)[s]\right] \\ &- \frac{2\delta_{\text{eval}}(1-Z)}{1-\gamma}\mathbb{E}_{\substack{f\sim p_{\text{eval}}(\cdot)\\ s\sim d^{\tilde{\mu}_f}(\cdot)}}\left[D_{\text{TV}}(\tilde{\mu}_f,\mu_f)[s]\right],\end{aligned} \tag{8}$$

where $r_{\max} = \max_{s,a}|r(s,a)|$, $\delta_{\text{train}} = \max_{f\in\mathcal{F}_{\text{train}}}\{\max_{s,a}|A^{\mu_f}(s,a)|\}$ *and* $\delta_{\text{eval}} = \max_{f\in\mathcal{F}_{\text{eval}}}\{\max_{s,a}|A^{\mu_f}(s,a)|\}$.

Proof. See Appendix E.2. □

Remark 3.5. Theorems 3.3 and 3.4 show that, unlike the training performance lower bound, the generalization performance lower bound contains three additional terms scaled by the common coefficient $(1-Z)$. Thus, increasing Z tightens the lower bound. In the degenerate case $Z=1$, the evaluation complement is empty, p_{eval} is unnecessary, and $p_{\text{train}} = p$, so generalization and training performance coincide. Figure 2 of Cobbe et al. (2020) is consistent with this conclusion.

However, once the training level (i.e., $\mathcal{F}_{\text{train}}$) is fixed, Z is constant, and improving generalization performance requires constraining the following three terms:

$$\underbrace{\mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot)\\ s\sim d^{\tilde{\mu}_f}(\cdot)}}\left[D_{\text{TV}}(\tilde{\mu}_f,\mu_f)[s]\right]}_{\text{denote it as }\mathfrak{D}_1},\ \underbrace{\mathbb{E}_{\substack{f\sim p_{\text{eval}}(\cdot)\\ s\sim d^{\tilde{\mu}_f}(\cdot)}}\left[D_{\text{TV}}(\tilde{\mu}_f,\mu_f)[s]\right]}_{\text{denote it as }\mathfrak{D}_2},\ \underbrace{\mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot)\\ s\sim d^{\mu_f}(\cdot)}}\left[D_{\text{TV}}(\tilde{\mu}_f,\mu_f)[s]\right]}_{\text{denote it as }\mathfrak{D}_{\text{train}}}. \tag{9}$$

During training, we can empirically bound only $\mathfrak{D}_{\text{train}}$ because it does not depend on the unseen state distribution $d^{\tilde{\mu}_f}(\cdot)$. We next establish upper bounds on $\mathfrak{D}_1$ and $\mathfrak{D}_2$ in the following theorem:

Theorem 3.6. *Given any two policies, $\tilde{\pi}$ and π, the following bounds hold:*

$$\mathfrak{D}_1 \leq \left(1+\frac{2\gamma\sigma_{\text{train}}}{1-\gamma}\right)\mathfrak{D}_{\text{train}},\quad \mathfrak{D}_2 \leq \left(1+\frac{2\gamma\sigma_{\text{eval}}}{1-\gamma}\right)\underbrace{\mathbb{E}_{\substack{f\sim p_{\text{eval}}(\cdot)\\ s\sim d^{\mu_f}(\cdot)}}\left[D_{\text{TV}}(\tilde{\mu}_f,\mu_f)[s]\right]}_{\textit{denote it as }\mathfrak{D}_{\text{eval}}}. \tag{10}$$

where

$$\sigma_{\text{train}} = \max_{f\in\mathcal{F}_{\text{train}}}\max_{s\in\mathcal{S}} D_{\text{TV}}(\tilde{\mu}_f,\mu_f)[s],\quad \sigma_{\text{eval}} = \max_{f\in\mathcal{F}_{\text{eval}}}\max_{s\in\mathcal{S}} D_{\text{TV}}(\tilde{\mu}_f,\mu_f)[s]. \tag{11}$$

Proof. See Appendix E.4. □

It remains to relate $\mathfrak{D}_{\text{eval}}$ to $\mathfrak{D}_{\text{train}}$. We first define policy robustness to irrelevant features.

Definition 3.7 ($\mathcal{R}$-robust). We say that the policy π is $\mathcal{R}$-robust if it satisfies

$$\sup_{s\in\mathcal{S},\tilde{f},f\in\mathcal{F}} D_{\text{TV}}(\mu_{\tilde{f}},\mu_f)[s] = \mathcal{R}. \tag{12}$$

Remark 3.8. This definition quantifies how two rendering functions, $\tilde{f}$ and f, affect the policy π at any given underlying state s. If $\mathcal{R}=0$, then $D_{\text{TV}}(\mu_{\tilde{f}},\mu_f)[s]\equiv 0$, meaning that irrelevant features no longer affect the policy.

This definition is intended not to yield the tightest possible bound, but to show how policy robustness to irrelevant features can improve generalization. Using Definition 3.7, we establish the following upper bound on $\mathfrak{D}_{\text{eval}}$:

Theorem 3.9. *Given any two policies, $\tilde{\pi}$ and π, assume that $\tilde{\pi}$ is $\mathcal{R}_{\tilde{\pi}}$-robust and π is $\mathcal{R}_{\pi}$-robust. Then the following bound holds:*

$$\mathfrak{D}_{\text{eval}} \leq \left(1 + \frac{2\gamma\sigma_{\text{train}}}{1-\gamma}\right)\mathcal{R}_{\pi} + \mathcal{R}_{\tilde{\pi}} + \mathfrak{D}_{\text{train}}. \tag{13}$$

Proof. See Appendix E.5. □

Combining Theorems 3.4, 3.6, and 3.9 yields the following corollary:

Corollary 3.10. *Given any two policies, $\tilde{\pi}$ and π, the following bound holds:*

$$\zeta(\tilde{\pi}) \geq L_{\pi}(\tilde{\pi}) - C_{\text{train}}\mathfrak{D}_{\text{train}} - C_{\pi}\mathcal{R}_{\pi} - C_{\tilde{\pi}}\mathcal{R}_{\tilde{\pi}} - C. \tag{14}$$

Appendix C gives the explicit forms of C, C_{train}, $C_{\tilde{\pi}}$, and C_{π}.

Remark 3.11. Our central theoretical result shows that improving generalization performance requires not only minimizing $\mathfrak{D}_{\text{train}}$ during training but also increasing policy robustness to irrelevant features by reducing $\mathcal{R}_{\pi}$ and $\mathcal{R}_{\tilde{\pi}}$. These results rely only on the mild Assumptions 2.1, 2.2, and 3.1. They therefore constitute a novel contribution applicable to a wide range of algorithms.

3.2 A Direct Bridge from DML to the Performance Lower Bound

The preceding analysis connects rendering robustness to generalization, but it does not establish that peer agreement induces rendering invariance. We now make this missing link precise. For peers $i, j \in \{1, 2\}$, let

$$\mu_{i,f}(\cdot \mid s) := \pi_i(\cdot \mid f(s)), \quad \mathcal{K}_{j\to i} := \sup_{s\in\mathcal{S},\, f\in\mathcal{F}} D_{\text{KL}}(\mu_{j,f} \| \mu_{i,f})[s]. \tag{15}$$

The quantity $\mathcal{K}_{j\to i}$ is the worst-case population analogue of the same-view DML loss.

Proposition 3.12 (Same-view agreement alone is insufficient)**.** *There exist two policies such that $\mathcal{K}_{1\to 2} = \mathcal{K}_{2\to 1} = 0$ but $\mathcal{R}_{\pi_1} = \mathcal{R}_{\pi_2} = 1$. Hence, no unconditional bound of the form $\mathcal{R}_{\pi_i} \leq \varphi(\mathcal{K}_{j\to i})$ with $\varphi(0) = 0$ can hold universally.*

Proof. Consider a single underlying state s, two actions a, b, and two renderings $f(s) = x$ and $g(s) = y$. Set $\pi_1(\cdot \mid x) = \pi_2(\cdot \mid x) = \delta_a$ and $\pi_1(\cdot \mid y) = \pi_2(\cdot \mid y) = \delta_b$. The peers agree exactly at each observation, yet each policy changes its action completely between the two renderings. □

Proposition 3.12 shows that same-view DML can transfer existing robustness, but cannot create robustness when all peers follow the same spurious rule.

Theorem 3.13 (Robustness transfer through DML)**.** *For any learner i and peer j,*

$$\mathcal{R}_{\pi_i} \leq \min\left\{1, \mathcal{R}_{\pi_j} + \sqrt{2\mathcal{K}_{j\to i}}\right\}. \tag{16}$$

Proof. See Appendix E.6. □

To compare updates, fix two peers and let $\bar{\pi} = (\pi_1 + \pi_2)/2$ and $\pi_i^{\lambda} = (1-\lambda)\pi_i + \lambda\pi_{3-i}$, using one weight $0 \leq \lambda \leq 1/2$ at every observation. These are same-view KL barycenters of frozen targets (Appendix E.7), not assumed neural-network updates.

Theorem 3.14 (Mutual distillation and policy robustness)**.** *For the same-view updates π_i^{λ} defined above,*

$$\mathcal{R}_{\pi_i^{\lambda}} \leq (1-2\lambda)\mathcal{R}_{\pi_i} + 2\lambda\mathcal{R}_{\bar{\pi}}, \quad i \in \{1, 2\}. \tag{17}$$

The mean and maximum of the two robustness radii do not increase. For $\lambda > 0$, either quantity strictly decreases if and only if it exceeds $\mathcal{R}_{\bar{\pi}}$ before the update.

Proof. See Appendix E.7. □

Complementary rendering errors can cancel in $\bar{\pi}$ without a robust teacher; shared errors survive. The theorem guarantees collective, not necessarily per-peer, improvement.

Theorem 3.15 (Robustness under an approximate distillation step)**.** *Given realized policies $\tilde{\pi}_i$, write $e_i(\cdot \mid o) = \tilde{\pi}_i(\cdot \mid o) - \pi_i^{\lambda}(\cdot \mid o)$ and define*

$$\varepsilon_i = \frac{1}{2} \sup_{s,f,g} \|e_i(\cdot \mid f(s)) - e_i(\cdot \mid g(s))\|_1 . \tag{18}$$

Then

$$\mathcal{R}_{\tilde{\pi}_i} \le B_i := \min\{1, (1-2\lambda)\mathcal{R}_{\pi_i} + 2\lambda\mathcal{R}_{\bar{\pi}} + \varepsilon_i\}. \tag{19}$$

In particular, $\mathcal{R}_{\tilde{\pi}_i} < \mathcal{R}_{\pi_i}$ whenever $2\lambda(\mathcal{R}_{\pi_i} - \mathcal{R}_{\bar{\pi}}) > \varepsilon_i$.

Proof. See Appendix E.8. □

The residual is not assumed small; the reference λ need not equal MDPO's KL weight α.

Corollary 3.16 (A refined generalization lower bound)**.** *For the policies and B_i above, let $\mathfrak{D}_{\text{train}}^{(i)}$ and $\epsilon_{\text{train}}^{(i)}$ be the quantities in Theorem 3.3 for $(\pi_i, \tilde{\pi}_i)$. With $C = 2r_{\max}(1-Z)/(1-\gamma)$,*

$$\zeta(\tilde{\pi}_i) \ge L_{\pi_i}(\tilde{\pi}_i) - \frac{2\gamma\epsilon_{\text{train}}^{(i)}}{(1-\gamma)^2}\mathfrak{D}_{\text{train}}^{(i)} - C \min\left\{1, \frac{B_i}{1-\gamma}\right\}. \tag{20}$$

This bound is no weaker than Corollary 3.10 with B_i substituted for $\mathcal{R}_{\tilde{\pi}_i}$ at the same policy pair.

Proof. See Appendix E.9. □

This refines the reward-range penalty without coverage or cross-render training, but does not infer actual return improvement from robustness alone.

4 Distillation as Regularization

Despite these theoretical advances, the underlying MDP and rendering function remain unknown in typical generalization settings. We therefore introduce a minimal toy example in Section 4.1, followed by an in-depth analysis in Section 4.2 that motivates our hypothesis.

4.1 Toy Example

We consider a simple environment in which an agent aims to collect coins to obtain rewards (see Figure 1). The agent observes raw pixel inputs. The true objective of the task is clearly to pick up the coins, whereas the background color is a spurious feature. However, in the training data for policy A, a strong correlation emerges: coins always appear on the agent's right in red backgrounds and on its left in cyan backgrounds. Consequently, when trained with standard reinforcement learning algorithms, policy A tends to overfit this spurious correlation, exhibiting behaviors such as moving right in red backgrounds and left in cyan backgrounds. As a result, policy A is not robust to variations in the background color, an irrelevant feature for this task.



Figure 1: This is a toy environment where the gray agent's goal is to pick up coins.

Policy A's overfitting to background color does not arise in policy B's training data, where the coin can appear on either side regardless of whether the background is red or cyan. Through DML, policy B therefore regularizes policy A and mitigates its tendency to overfit to background color. Conversely, any reliance on irrelevant features by policy A would harm policy B's performance,

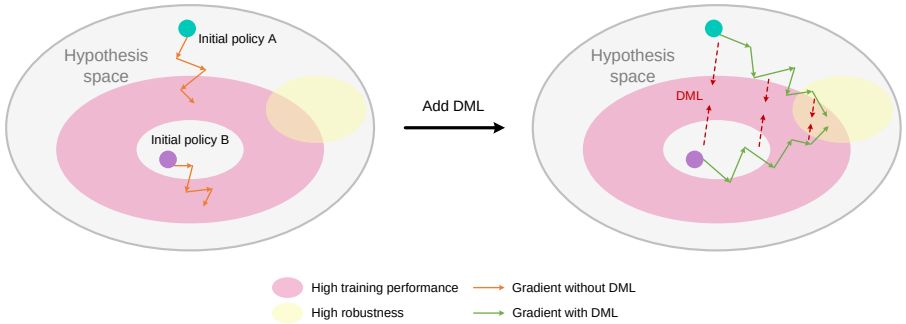


Figure 2: (Left) Two policies trained independently may overfit to irrelevant features, failing to converge to a robust hypothesis space. Thus, high training performance reflects only spurious correlations. (Right) Through mutual distillation, the two policies regularize each other and gradually converge toward a more robust hypothesis space, ultimately improving generalization performance.

and vice versa. We thus hypothesize that mutual distillation encourages both policies to capture the task's true underlying semantics, ultimately improving generalization across environments.

4.2 Central Hypothesis

Motivated by Section 4.1, we view DML as implicit regularization against irrelevant features. Figure 2 depicts two randomly initialized policies trained independently. Because their samples cover only part[2] of the MDP family, these policies can overfit irrelevant features and miss a robust hypothesis space despite high training performance.

Applying DML to the training process of both policies enables mutual distillation, helping to mitigate overfitting to irrelevant features. Due to the randomness in parameter initialization and interactions with the environment, both policies encounter biased training samples, and DML encourages them to make consistent decisions based on the same observations. As discussed in Section 4.1, any irrelevant features learned by one policy can degrade the performance of the other. However, Proposition 3.12 shows that agreement alone is insufficient when both peers share the same spurious rule. Theorem 3.13 gives the precise positive statement: DML transfers a peer's robustness to the learner, with an explicit KL penalty in the lower bound. Over time, DML drives both policies to learn more meaningful representations, gradually reducing the divergence between them. We hypothesize that both policies eventually capture the essential aspects of high-dimensional observations.

5 Experiments

We present the implementation (Section 5.1), generalization results (Section 5.2), robustness test (Section 5.3), and ablations (Section 5.4).

5.1 Implementation Details

We evaluate on Procgen's 16 procedurally generated environments, a standard benchmark for RL sample efficiency and generalization (Cobbe et al., 2019; 2020; Wang et al., 2020; Raileanu & Fergus, 2021; Raileanu et al., 2021; Lyle et al., 2022; Rahman & Xue, 2023; Jesson & Jiang, 2024).

We employ Proximal Policy Optimization (PPO) (Schulman et al., 2017; Cobbe et al., 2020) as our baseline. Specifically, for a parameterized policy π_θ, where θ denotes the policy parameters, the clipped policy objective is

$$J_{\text{clip}}(\theta) = \mathbb{E}_{\substack{o_t \sim q_{\theta_{\text{old}}} \\ a_t \sim \pi_{\theta_{\text{old}}}(\cdot|o_t)}} \left[\min \left\{ r_t(\theta)\hat{A}_t, \tilde{r}_t(\theta)\hat{A}_t \right\} \right], \tag{21}$$

[2]In empirical reinforcement learning, the training data used to update the policy cannot cover all possible trajectories and are therefore inevitably biased, leading to spurious correlations.

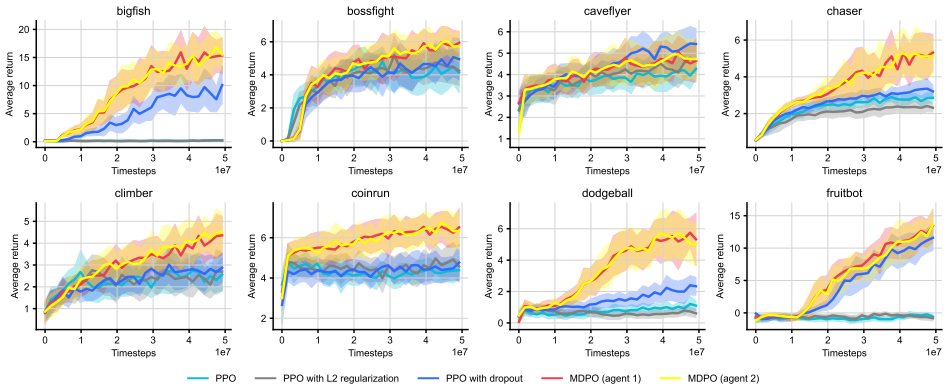


Figure 3: Generalization performance of different methods on 500 levels of the Procgen benchmark. The mean and standard deviation are shown across five random seeds.

Algorithm 1: Mutual Distillation Policy Optimization

Initialize: agents π_1, π_2, PPO $\mathcal{A}$, weight α
while training **do**
  for $i \in \{1, 2\}$ **do**
    Collect $\mathcal{D}_i \sim \pi_i$ with empirical distribution q_i
    $\ell_i \leftarrow \mathbb{E}_{o \sim q_i}[D_{\mathrm{KL}}(\pi_{3-i} \| \pi_i)[o]]$
    $\mathcal{L}_i \leftarrow \mathcal{A}(\mathcal{D}_i) + \alpha \ell_i$
  end for
  Update π_1, π_2 using $\mathcal{L} \leftarrow \frac{1}{2}(\mathcal{L}_1 + \mathcal{L}_2)$
end while

Table 1: Finite-sample robustness proxy $\hat{\mathcal{R}}$ (lower is better)

Environment	PPO	MDPO	Improvement
bossfight	0.9953	**0.9107**	8.50%
caveflyer	0.9999	**0.9031**	9.68%
climber	0.9988	**0.6995**	29.97%
heist	0.7480	**0.4341**	41.97%
jumper	0.9533	**0.6947**	27.13%
maze	0.9923	**0.6447**	35.03%
ninja	0.9342	**0.8586**	8.09%
starpilot	1.0000	**0.9246**	7.54%

where $q_{\theta_{\mathrm{old}}}$ is the rollout observation distribution, $\hat{A}_t = \hat{A}(o_t, a_t)$ is the advantage estimate, $r_t(\theta) := \frac{\pi_\theta(a_t|o_t)}{\pi_{\theta_{\mathrm{old}}}(a_t|o_t)}$ and $\tilde{r}_t(\theta) := \mathrm{clip}(r_t(\theta), 1-\epsilon, 1+\epsilon)$. Thus, for the policy term, we minimize $\mathcal{L}_{\mathrm{RL}} := -J_{\mathrm{clip}}$ (with the standard value and entropy terms omitted here for brevity).

We randomly initialize two agents to interact with the environment and collect data independently. Following the DML objective used in supervised learning, we add KL divergence regularization:

$$\mathcal{L}_{\mathrm{DML}} = \mathcal{L}_{\mathrm{RL}} + \alpha \mathcal{L}_{\mathrm{KL}}, \tag{22}$$

where $\mathcal{L}_{\mathrm{KL}}$ is the KL divergence loss and α is its weight. Algorithm 1 averages the two directional losses into the joint objective of Mutual Distillation Policy Optimization (MDPO).

Rather than exhaustively comparing baselines, we test whether DML reduces overfitting to irrelevant features and improves generalization. Although related to Zhao & Hospedales (2021), our mechanism is distinguished by the theoretical results in Section 3.

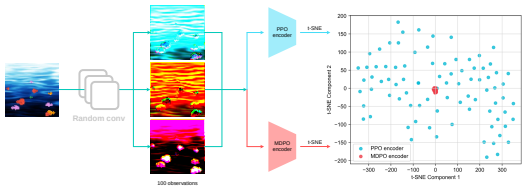


Figure 4: Random-CNN perturbations and encoder features' visualization.

5.2 Empirical Results

Under Procgen's hard-level setting (Cobbe et al., 2020), MDPO consistently outperforms PPO across all environments, with large gains on bigfish, dodgeball, and fruitbot (Figure 3). Appendix D reports the complete training and test curves.

To separate robustness gains from the generic effect of peer gradients, Sections 5.3 and 5.4 provide targeted tests and ablations.

Algo\Env	bigfish	chaser	dodgeball	fruitbot
PPO (PPO encoder)	0.19 ± 0.14	2.57 ± 0.28	0.71 ± 0.34	-0.39 ± 0.46
PPO (MDPO encoder)	$\mathbf{22.67 \pm 6.40}$	$\mathbf{6.22 \pm 1.36}$	$\mathbf{4.70 \pm 1.91}$	$\mathbf{11.22 \pm 2.16}$

Table 2: Generalization performance of PPO *linear probe* on top of the *frozen* encoders across five random seeds.

α in MDPO\Env	bigfish	dodgeball	fruitbot	starpilot
0 (baseline)	0.26 ± 0.23	0.92 ± 0.46	-0.50 ± 0.81	3.99 ± 0.21
0.1	9.87 ± 4.57	4.35 ± 1.63	11.94 ± 2.96	10.97 ± 2.72
1	$\mathbf{16.11 \pm 4.63}$	$\mathbf{5.66 \pm 1.98}$	$\mathbf{13.23 \pm 3.04}$	$\mathbf{11.28 \pm 3.04}$
10	7.69 ± 3.65	4.35 ± 1.48	2.31 ± 2.41	8.54 ± 2.27

Table 3: Generalization performance of MDPO under different KL divergence weights across five random seeds.

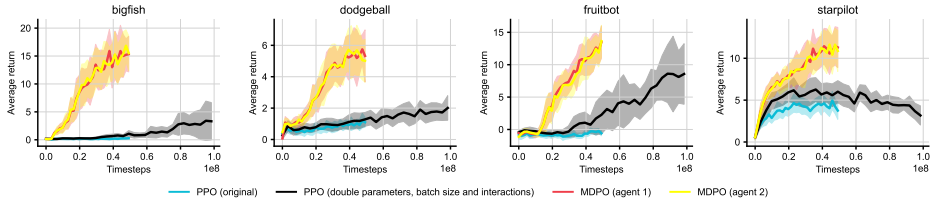


Figure 5: Generalization performance of the PPO baseline with *double* the model size, batch size, and total number of interactions, compared with the original PPO and MDPO methods (for training results, see Figure 6). The mean and standard deviation are shown across five random seeds.

5.3 Robustness Testing

We test robustness by passing observations through random Gaussian-initialized convolutional layers (Figure 4). MDPO representations remain concentrated across perturbations (red), whereas PPO representations are dispersed (blue). We also estimate Definition 3.7 using 100 rollout observations and 100 perturbations $\mathcal{T}_i^j : \mathcal{O} \to \mathcal{O}$ per observation.

$$\hat{\mathcal{R}} := \max_{i,j \in \{1,\dots,100\}} D_{\mathrm{TV}}\big(\pi_\theta(\cdot \mid o_i), \pi_\theta(\cdot \mid \mathcal{T}_i^j(o_i))\big). \tag{23}$$

Unlike the population supremum in Definition 3.7, this is a finite-sample, identity-anchored proxy. MDPO lowers it in all eight environments (Table 1), supporting improved empirical robustness.

5.4 Ablation Study

We conduct additional ablations by *doubling* the number of policy parameters, the training batch size, and the total number of interactions for the PPO baseline. Figure 5 shows that this enlarged baseline still does not match the generalization performance of MDPO, indicating that increased model capacity, batch size, or interaction budget alone does not explain the improvement. We also evaluate the learned representations by retraining a PPO *linear probe* on the *frozen* encoders of the fully trained PPO and MDPO policies. This retraining uses only 2% of the interactions required for the original training. As reported in Table 2, the linear probe built on the MDPO encoder consistently outperforms the one built on the PPO encoder, suggesting that mutual distillation produces more robust representations. Finally, Table 3 reports a sensitivity analysis of the KL divergence weight α.

6 Conclusion

We develop a theoretical framework that relates generalization in deep reinforcement learning to policy robustness against irrelevant features. We then characterize how mutual distillation can act as implicit regularization: same-view DML transfers robustness from a peer under explicit conditions, and approximate same-view updates admit explicit robustness-improvement conditions. Procgen experiments support the predicted link between rendering robustness and generalization.

This benefit entails added cost: MDPO doubles the trainable parameters and environment interactions of a single-policy baseline, potentially limiting deployment under constrained computation or samples. Reducing this overhead remains important for practical RL systems.

AI USE STATEMENT

In this work, we used generative AI tools to aid or polish writing. We have reviewed all AI-assisted work and take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

The algorithms presented in this paper have been submitted as Supplementary Material, enabling anyone to reproduce the results reported in this paper.

REFERENCES

Joshua Achiam, David Held, Aviv Tamar, and Pieter Abbeel. Constrained policy optimization. In *International conference on machine learning*, pp. 22–31. Pmlr, 2017.

Sara L Bengtsson, John-Dylan Haynes, Katsuyuki Sakai, Mark J Buckley, and Richard E Passingham. The representation of abstract task rules in the human prefrontal cortex. *Cerebral Cortex*, 19(8):1929–1936, 2009.

Karl Cobbe, Oleg Klimov, Chris Hesse, Taehoon Kim, and John Schulman. Quantifying generalization in reinforcement learning. In *International conference on machine learning*, pp. 1282–1289. PMLR, 2019.

Karl Cobbe, Chris Hesse, Jacob Hilton, and John Schulman. Leveraging procedural generation to benchmark reinforcement learning. In *International conference on machine learning*, pp. 2048–2056. PMLR, 2020.

Karl W Cobbe, Jacob Hilton, Oleg Klimov, and John Schulman. Phasic policy gradient. In *International Conference on Machine Learning*, pp. 2020–2027. PMLR, 2021.

Iroise Dumontheil. Development of abstract thinking during childhood and adolescence: The role of rostrolateral prefrontal cortex. *Developmental cognitive neuroscience*, 10:57–76, 2014.

Jesse Farebrother, Marlos C Machado, and Michael Bowling. Generalization and regularization in dqn. *arXiv preprint arXiv:1810.00123*, 2018.

Yaozhong Gan, Renye Yan, Xiaoyang Tan, Zhe Wu, and Junliang Xing. Transductive off-policy proximal policy optimization. *arXiv preprint arXiv:2406.03894*, 2024.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

Jake Grigsby and Yanjun Qi. Measuring visual generalization in continuous control from pixels. *arXiv preprint arXiv:2010.06740*, 2020.

Shengyi Huang, Rousslan Fernand Julien Dossa, Chang Ye, Jeff Braga, Dipam Chakraborty, Kinal Mehta, and JoÃǪo GM AraÃšjo. Cleanrl: High-quality single-file implementations of deep reinforcement learning algorithms. *Journal of Machine Learning Research*, 23(274):1–18, 2022.

Maximilian Igl, Kamil Ciosek, Yingzhen Li, Sebastian Tschiatschek, Cheng Zhang, Sam Devlin, and Katja Hofmann. Generalization in reinforcement learning with selective noise injection and information bottleneck. *Advances in neural information processing systems*, 32, 2019.

Andrew Jesson and Yiding Jiang. Improving generalization on the procgen benchmark with simple architectural changes and scale. *arXiv preprint arXiv:2410.10905*, 2024.

Yiding Jiang, J Zico Kolter, and Roberta Raileanu. On the importance of exploration for generalization in reinforcement learning. *Advances in Neural Information Processing Systems*, 36: 12951–12986, 2023.

Niels Justesen, Ruben Rodriguez Torrado, Philip Bontrager, Ahmed Khalifa, Julian Togelius, and Sebastian Risi. Illuminating generalization in deep reinforcement learning through procedural level generation. *arXiv preprint arXiv:1806.10729*, 2018.

Sham Kakade and John Langford. Approximately optimal approximate reinforcement learning. In *Proceedings of the nineteenth international conference on machine learning*, pp. 267–274, 2002.

Robert Kirk, Amy Zhang, Edward Grefenstette, and Tim Rocktäschel. A survey of zero-shot generalisation in deep reinforcement learning. *Journal of Artificial Intelligence Research*, 76:201–264, 2023.

Ilya Kostrikov, Denis Yarats, and Rob Fergus. Image augmentation is all you need: Regularizing deep reinforcement learning from pixels. *arXiv preprint arXiv:2004.13649*, 2020.

Kwei-Herng Lai, Daochen Zha, Yuening Li, and Xia Hu. Dual policy distillation. *arXiv preprint arXiv:2006.04061*, 2020.

Misha Laskin, Kimin Lee, Adam Stooke, Lerrel Pinto, Pieter Abbeel, and Aravind Srinivas. Reinforcement learning with augmented data. *Advances in neural information processing systems*, 33: 19884–19895, 2020.

Kimin Lee, Kibok Lee, Jinwoo Shin, and Honglak Lee. Network randomization: A simple technique for generalization in deep reinforcement learning. *arXiv preprint arXiv:1910.05396*, 2019.

Clare Lyle, Mark Rowland, Will Dabney, Marta Kwiatkowska, and Yarin Gal. Learning dynamics and generalization in deep reinforcement learning. In *International conference on machine learning*, pp. 14560–14581. PMLR, 2022.

Guozheng Ma, Zhen Wang, Zhecheng Yuan, Xueqian Wang, Bo Yuan, and Dacheng Tao. A comprehensive survey of data augmentation in visual reinforcement learning. *International Journal of Computer Vision*, 133(10):7368–7405, 2025.

Bogdan Mazoure, Ahmed M Ahmed, Patrick MacAlpine, R Devon Hjelm, and Andrey Kolobov. Cross-trajectory representation learning for zero-shot generalization in rl. *arXiv preprint arXiv:2106.02193*, 2021.

Bhairav Mehta, Manfred Diaz, Florian Golemo, Christopher J Pal, and Liam Paull. Active domain randomization. In *Conference on robot learning*, pp. 1162–1176. PMLR, 2020.

Seungyong Moon, JunYeong Lee, and Hyun Oh Song. Rethinking value function learning for generalization in reinforcement learning. *Advances in Neural Information Processing Systems*, 35: 34846–34858, 2022.

A Tuan Nguyen, Toan Tran, Yarin Gal, and Atilim Gunes Baydin. Domain invariant representation learning with domain density transformations. *Advances in neural information processing systems*, 34:5264–5275, 2021.

Charles Packer, Katelyn Gao, Jernej Kos, Philipp Krähenbühl, Vladlen Koltun, and Dawn Song. Assessing generalization in deep reinforcement learning. *arXiv preprint arXiv:1810.12282*, 2018.

Hyun-Jae Pi, Balázs Hangya, Duda Kvitsiani, Joshua I Sanders, Z Josh Huang, and Adam Kepecs. Cortical interneurons that specialize in disinhibitory control. *Nature*, 503(7477):521–524, 2013.

Md Masudur Rahman and Yexiang Xue. Adversarial style transfer for robust policy optimization in deep reinforcement learning. *arXiv preprint arXiv:2308.15550*, 2023.

Roberta Raileanu and Rob Fergus. Decoupling value and policy for generalization in reinforcement learning. In *International Conference on Machine Learning*, pp. 8787–8798. PMLR, 2021.

Roberta Raileanu, Maxwell Goldstein, Denis Yarats, Ilya Kostrikov, and Rob Fergus. Automatic data augmentation for generalization in reinforcement learning. *Advances in Neural Information Processing Systems*, 34:5402–5415, 2021.

Aravind Rajeswaran, Kendall Lowrey, Emanuel V Todorov, and Sham M Kakade. Towards generalization and simplicity in continuous control. *Advances in neural information processing systems*, 30, 2017.

John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region policy optimization. In *International conference on machine learning*, pp. 1889–1897. Pmlr, 2015.

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

Connor Shorten and Taghi M Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of big data*, 6(1):60, 2019.

Reda Bahi Slaoui, William R Clements, Jakob N Foerster, and Sébastien Toth. Robust visual domain randomization for reinforcement learning. *arXiv preprint arXiv:1910.10537*, 2019.

Richard D Smallwood and Edward J Sondik. The optimal control of partially observable markov processes over a finite horizon. *Operations research*, 21(5):1071–1088, 1973.

Xingyou Song, Yiding Jiang, Stephen Tu, Yilun Du, and Behnam Neyshabur. Observational overfitting in reinforcement learning. *arXiv preprint arXiv:1912.02975*, 2019.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

Miguel Suau, Matthijs TJ Spaan, and Frans A Oliehoek. Bad habits: Policy confounding and out-of-trajectory generalization in rl. *arXiv preprint arXiv:2306.02419*, 2023.

Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. In *2017 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pp. 23–30. IEEE, 2017.

Kaixin Wang, Bingyi Kang, Jie Shao, and Jiashi Feng. Improving generalization in reinforcement learning with mixture regularization. *Advances in Neural Information Processing Systems*, 33: 7968–7978, 2020.

Max Weltevrede, Caroline Horsch, Matthijs TJ Spaan, and Wendelin Böhmer. Exploration implies data augmentation: Reachability and generalisation in contextual mdps. In *NeurIPS 2025 Workshop: Second Workshop on Aligning Reinforcement Learning Experimentalists and Theorists*, 2024.

Max Weltevrede, Moritz Zanger, Matthijs Spaan, and Wendelin Böhmer. How ensembles of distilled policies improve generalisation in reinforcement learning. *Advances in Neural Information Processing Systems*, 38:135932–135972, 2026.

Zhengpeng Xie, Li Lyna Zhang, Zeke Xie, and Mao Yang. Trust region policy distillation. *arXiv preprint arXiv:2607.04751*, 2026.

Xiaohan Xu, Ming Li, Chongyang Tao, Tao Shen, Reynold Cheng, Jinyang Li, Can Xu, Dacheng Tao, and Tianyi Zhou. A survey on knowledge distillation of large language models. *arXiv preprint arXiv:2402.13116*, 2024.

Zhecheng Yuan, Sizhe Yang, Pu Hua, Can Chang, Kaizhe Hu, and Huazhe Xu. Rl-vigen: A reinforcement learning benchmark for visual generalization. *Advances in Neural Information Processing Systems*, 36:6720–6747, 2023.

Xiangyu Yue, Yang Zhang, Sicheng Zhao, Alberto Sangiovanni-Vincentelli, Kurt Keutzer, and Boqing Gong. Domain randomization and pyramid consistency: Simulation-to-real generalization without accessing target domain data. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 2100–2110. IEEE, 2019.

Amy Zhang, Rowan McAllister, Roberto Calandra, Yarin Gal, and Sergey Levine. Learning invariant representations for reinforcement learning without reconstruction. *arXiv preprint arXiv:2006.10742*, 2020.

Chiyuan Zhang, Oriol Vinyals, Remi Munos, and Samy Bengio. A study on overfitting in deep reinforcement learning. *arXiv preprint arXiv:1804.06893*, 2018a.

Hanping Zhang and Yuhong Guo. Generalization of reinforcement learning with policy-aware adversarial data augmentation. *arXiv preprint arXiv:2106.15587*, 2021.

Ying Zhang, Tao Xiang, Timothy M Hospedales, and Huchuan Lu. Deep mutual learning. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4320–4328. IEEE, 2018b.

Chenyang Zhao and Timothy Hospedales. Robust domain randomised reinforcement learning through peer-to-peer distillation. In *Asian Conference on Machine Learning*, pp. 1237–1252. PMLR, 2021.

A Related Work

Generalization in deep reinforcement learning has been widely studied, with overfitting recognized as a key challenge (Rajeswaran et al., 2017; Zhang et al., 2018a; Justesen et al., 2018; Packer et al., 2018; Song et al., 2019; Cobbe et al., 2019; Grigsby & Qi, 2020; Cobbe et al., 2020; Yuan et al., 2023; Suau et al., 2023; Kirk et al., 2023). A common remedy applies regularization techniques from supervised learning, including dropout (Srivastava et al., 2014; Farebrother et al., 2018; Igl et al., 2019), data augmentation (Laskin et al., 2020; Kostrikov et al., 2020; Zhang & Guo, 2021; Raileanu et al., 2021; Ma et al., 2025), and domain randomization (Tobin et al., 2017; Yue et al., 2019; Slaoui et al., 2019; Lee et al., 2019; Mehta et al., 2020). To improve sample efficiency, prior studies encourage policy and value networks to share parameters (Schulman et al., 2017; Huang et al., 2022); others instead decouple these networks and add distillation strategies (Cobbe et al., 2021; Raileanu & Fergus, 2021; Moon et al., 2022). Exploration also improves policy generalization (Jiang et al., 2023; Weltevrede et al., 2024) by changing the initial-state distribution and generating more diverse trajectories during training (Weltevrede et al., 2024).

Representation learning offers another route to generalization. Prior work uses bisimulation metrics to capture invariances by comparing states through reward and transition distributions (Zhang et al., 2020), or self-supervised objectives that align trajectories by behavioral similarity (Mazoure et al., 2021). These approaches let the encoder learn visually robust features without explicit reward signals. However, their separate encoder-pretraining stage can reduce sample efficiency, yield suboptimal downstream representations, and limit end-to-end adaptation to new tasks.

Knowledge distillation transfers knowledge by aligning a student and teacher. It commonly distills a large model into a smaller one to reduce inference costs (Xu et al., 2024; Xie et al., 2026). It can also transfer privileged information to partially informed models or support mutual learning among students through DML (Zhang et al., 2018b). Lai et al. (2020) propose a student-student distillation method that improves performance without a pre-trained teacher. Zhao & Hospedales (2021) show that DML improves RL generalization without explaining why. More recently, Weltevrede et al. (2026) show that distilling policies into an ensemble across diverse training states improves zero-shot generalization, but only in toy environments with symmetric group structures.

B HYPERPARAMETERS

Table 4 summarizes the hyperparameters used in our experiments. Most settings follow the hard-level configuration of Cobbe et al. (2020), except that we train for 50M rather than 200M steps. Policies are trained on the initial 500 levels and evaluated for generalization over the full level distribution.

Table 4: Detailed hyperparameters in Procgen.

Hyperparameter\Algorithm	PPO (Schulman et al., 2017)	MDPO (ours)
Number of workers	64	64
Horizon	256	256
Learning rate	0.0005	0.0005
Learning rate decay	No	No
Optimizer	Adam	Adam
Total interaction steps	50M	50M
Update epochs	3	3
Mini-batches	8	8
Batch size	16384	16384
Mini-batch size	2048	2048
Discount factor γ	0.999	0.999
GAE parameter λ	0.95	0.95
Value loss coefficient c_1	0.5	0.5
Entropy loss coefficient c_2	0.01	0.01
Clipping parameter ϵ	0.2	0.2
KL divergence weight α	-	1.0

C EXPLICIT FORM OF COROLLARY 3.10

Corollary 3.10. *Given any two policies, $\tilde{\pi}$ and π, the following bound holds:*

$$\zeta(\tilde{\pi}) \geq L_{\pi}(\tilde{\pi}) - C_{\text{train}}\mathfrak{D}_{\text{train}} - C_{\pi}\mathcal{R}_{\pi} - C_{\tilde{\pi}}\mathcal{R}_{\tilde{\pi}} - C, \tag{24}$$

where $C = \frac{2r_{\max}(1-Z)}{1-\gamma}$, *and*

$$\begin{aligned} C_{\text{train}} &= \frac{2\delta_{\text{train}}(1-Z)}{1-\gamma}\left(1+\frac{2\gamma\sigma_{\text{train}}}{1-\gamma}\right) + \frac{2\delta_{\text{eval}}(1-Z)}{1-\gamma}\left(1+\frac{2\gamma\sigma_{\text{eval}}}{1-\gamma}\right) + \frac{2\gamma\epsilon_{\text{train}}}{(1-\gamma)^2}, \\ C_{\pi} &= \frac{2\delta_{\text{eval}}(1-Z)}{1-\gamma}\left(1+\frac{2\gamma\sigma_{\text{eval}}}{1-\gamma}\right)\left(1+\frac{2\gamma\sigma_{\text{train}}}{1-\gamma}\right), \\ C_{\tilde{\pi}} &= \frac{2\delta_{\text{eval}}(1-Z)}{1-\gamma}\left(1+\frac{2\gamma\sigma_{\text{eval}}}{1-\gamma}\right). \end{aligned} \tag{25}$$

Remark C.1. Assume additionally that $\tilde{\mu}_f(\cdot \mid s)$ is absolutely continuous with respect to $\mu_f(\cdot \mid s)$ for p_{train}-almost every f and d^{μ_f}-almost every s. Corollary 3.10 then implies that

$$\begin{aligned} \zeta(\tilde{\pi}) &\geq \eta(\pi) + \frac{1}{1-\gamma}\mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot)\\ s\sim d^{\mu_f}(\cdot)\\ a\sim\tilde{\mu}_f(\cdot|s)}}\left[A^{\mu_f}(s,a)\right] - C_{\text{train}}\mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot)\\ s\sim d^{\mu_f}(\cdot)}}\left[D_{\text{TV}}(\tilde{\mu}_f,\mu_f)[s]\right] - C_{\pi}\mathcal{R}_{\pi} - C_{\tilde{\pi}}\mathcal{R}_{\tilde{\pi}} - C \\ &= \underbrace{\eta(\pi)}_{\textit{training performance}} + \underbrace{\frac{1}{1-\gamma}\mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot)\\ s\sim d^{\mu_f}(\cdot)\\ a\sim\mu_f(\cdot|s)}}\left[\frac{\tilde{\mu}_f(a\mid s)}{\mu_f(a\mid s)}A^{\mu_f}(s,a)\right]}_{\textit{surrogate objective}} - \underbrace{\frac{C_{\text{train}}}{2}\mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot)\\ s\sim d^{\mu_f}(\cdot)\\ a\sim\mu_f(\cdot|s)}}\left[\left|\frac{\tilde{\mu}_f(a\mid s)}{\mu_f(a\mid s)}-1\right|\right]}_{\textit{TV distance penalty}} \\ &\quad - \underbrace{C_{\pi}\mathcal{R}_{\pi}}_{\textit{old policy robustness}} - \underbrace{C_{\tilde{\pi}}\mathcal{R}_{\tilde{\pi}}}_{\textit{new policy robustness}} - C. \end{aligned} \tag{26}$$

This decomposition motivates optimizing the surrogate objective while discouraging large policy updates through the TV distance penalty. Ratio clipping in the PPO objective equation 21 is a heuristic for limiting large importance-ratio changes; it does not by itself impose a hard TV constraint. Theorem 3.3 shows why controlling the policy distance improves the training-performance lower bound. Moreover, mutual distillation can improve the generalization lower bound by reducing the robustness terms $\mathcal{R}_{\pi}$ and $\mathcal{R}_{\tilde{\pi}}$ across iterations.

D MORE RESULTS

D.1 MORE ABLATION RESULTS

Figure 6 presents additional training curves from the Ablation Study (Section 5.4).

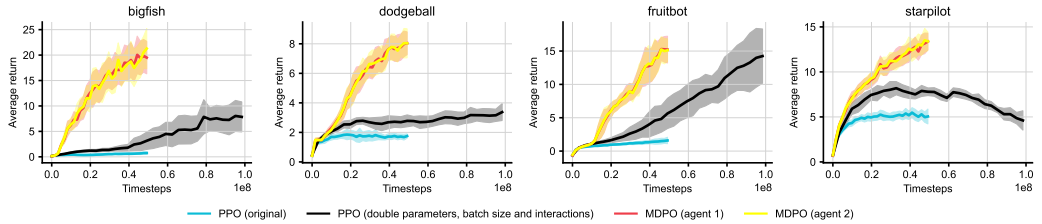


Figure 6: Training performance of PPO baseline with *double* model size, batch size, and total number of interactions, compared to original PPO and MDPO. The mean and standard deviation are shown across five random seeds.

D.2 FULL RESULTS

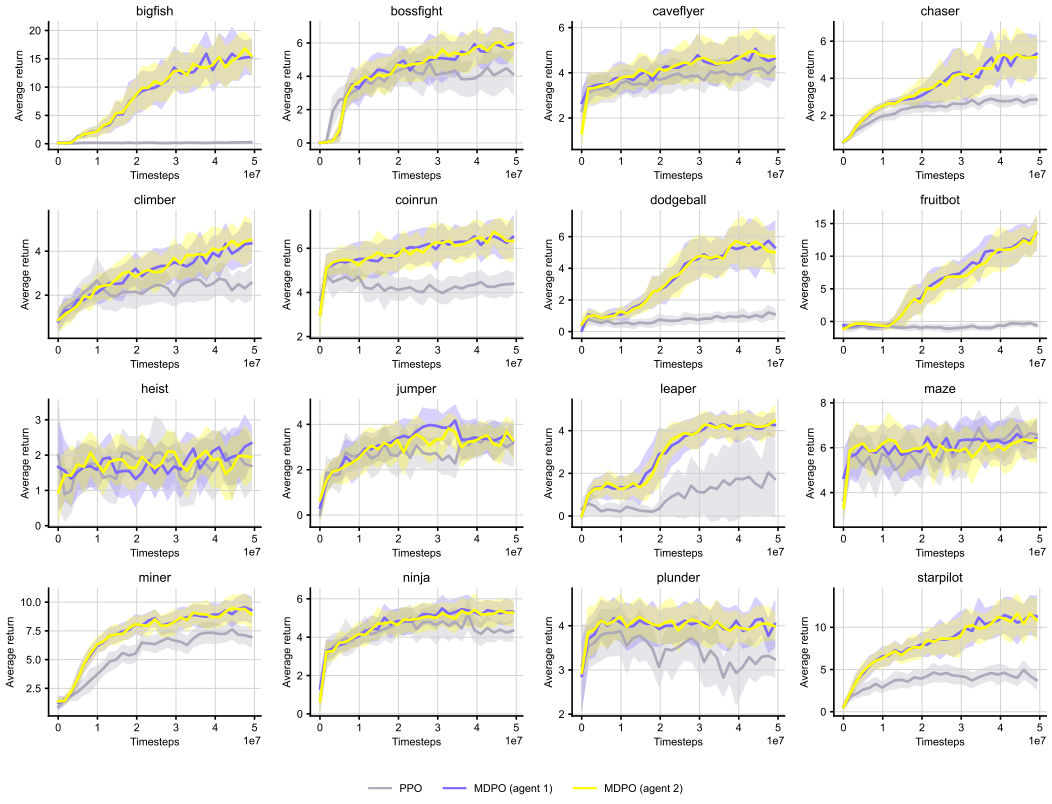


Figure 7: Generalization performance of PPO and MDPO from 500 levels in each environment across five random seeds.

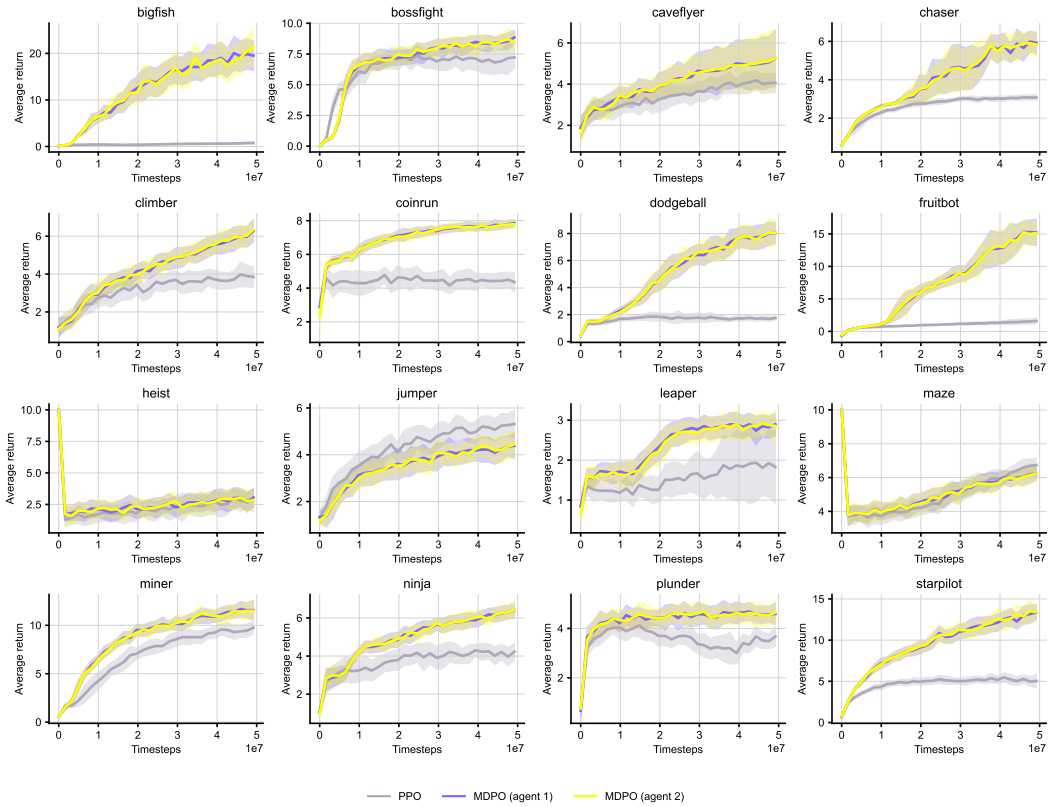


Figure 8: Training performance of PPO and MDPO from 500 levels in each environment across five random seeds.

D.3 Additional Visualizations

We further generate adversarial samples by manipulating the brightness, contrast, saturation, and hue of visual observations. As illustrated in Figure 9, we use these samples to evaluate the robustness of both the PPO encoder and our MDPO encoder.

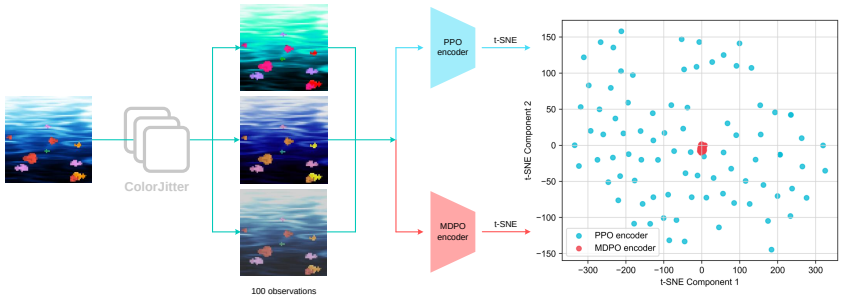


Figure 9: The robustness of PPO and MDPO to brightness, contrast, saturation, and hue.

The MDPO policy exhibits robust representations with respect to these irrelevant features, whereas the PPO policy remains sensitive to such variations. We additionally visualize adversarial samples generated by random CNN renderings (see Figure 10) and by random perturbations of brightness, contrast, saturation, and hue (see Figure 11).

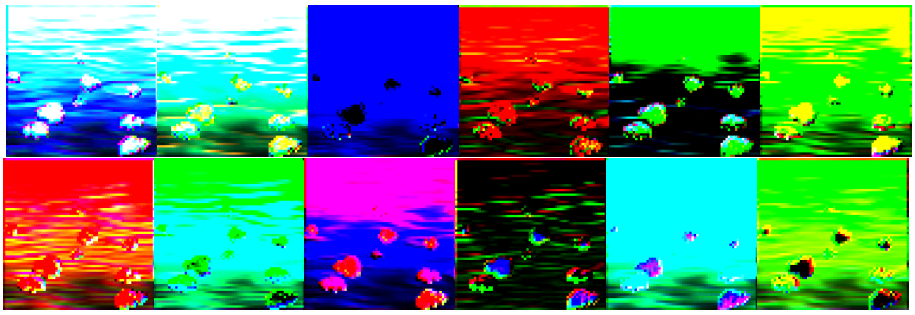


Figure 10: Adversarial samples generated by random CNNs.

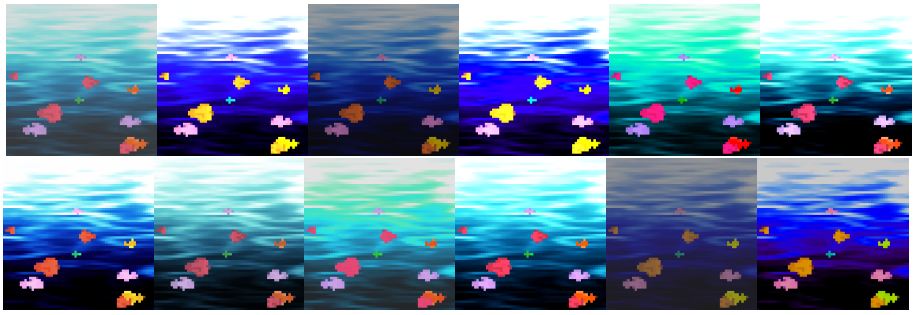


Figure 11: Adversarial samples generated by different brightness, contrast, saturation, and hue.

E PROOFS

We begin with several useful lemmas.

Lemma E.1 (Performance difference)**.** *Let $\mu_f(\cdot \mid s_t) = \pi(\cdot \mid f(s_t))$ and $\tilde{\mu}_f(\cdot \mid s_t) = \tilde{\pi}(\cdot \mid f(s_t))$, and define the training and generalization performance as*

$$\eta(\pi) = \frac{1}{1-\gamma}\mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot)\\ s\sim d^{\mu_f}(\cdot)\\ a\sim\mu_f(\cdot|s)}}\left[r(s,a)\right], \quad \zeta(\pi) = \frac{1}{1-\gamma}\mathbb{E}_{\substack{f\sim p(\cdot)\\ s\sim d^{\mu_f}(\cdot)\\ a\sim\mu_f(\cdot|s)}}\left[r(s,a)\right]. \tag{27}$$

Then the differences in training and generalization performance can be expressed as

$$\eta(\tilde{\pi}) - \eta(\pi) = \frac{1}{1-\gamma}\mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot)\\ s\sim d^{\tilde{\mu}_f}(\cdot)\\ a\sim\tilde{\mu}_f(\cdot|s)}}\left[A^{\mu_f}(s,a)\right], \quad \zeta(\tilde{\pi}) - \zeta(\pi) = \frac{1}{1-\gamma}\mathbb{E}_{\substack{f\sim p(\cdot)\\ s\sim d^{\tilde{\mu}_f}(\cdot)\\ a\sim\tilde{\mu}_f(\cdot|s)}}\left[A^{\mu_f}(s,a)\right]. \tag{28}$$

Proof. The result follows directly from Kakade & Langford (2002). □

Lemma E.2. *The distance between two normalized discounted visitation distributions, $\|d^{\tilde{\mu}} - d^{\mu}\|_1$, is bounded by the average distance between $\tilde{\mu}$ and μ:*

$$\|d^{\tilde{\mu}} - d^{\mu}\|_1 \leq \frac{\gamma}{1-\gamma}\mathbb{E}_{s\sim d^{\mu}(\cdot)}\left[\|\tilde{\mu} - \mu\|_1\right] = \frac{2\gamma}{1-\gamma}\mathbb{E}_{s\sim d^{\mu}(\cdot)}\left[D_{\text{TV}}(\tilde{\mu},\mu)[s]\right], \tag{29}$$

where $D_{\text{TV}}(\tilde{\mu},\mu)[s] = \frac{1}{2}\sum_{a\in\mathcal{A}}|\tilde{\mu}(a \mid s) - \mu(a \mid s)|$ denotes the total variation (TV) distance.

Proof. See Achiam et al. (2017). □

Lemma E.3. *For any state $s \in \mathcal{S}$ and any two policies $\tilde{\mu}$ and μ, the average advantage $\mathbb{E}_{a\sim\tilde{\mu}(\cdot|s)}\left[A^{\mu}(s,a)\right]$ is bounded by*

$$\left|\mathbb{E}_{a\sim\tilde{\mu}(\cdot|s)}\left[A^{\mu}(s,a)\right]\right| \leq 2D_{\mathrm{TV}}(\tilde{\mu},\mu)[s]\cdot\max_{a}\left|A^{\mu}(s,a)\right|. \tag{30}$$

Proof. First, note that

$$\begin{aligned}\mathbb{E}_{a\sim\mu(\cdot|s)}\left[A^{\mu}(s,a)\right] =& \mathbb{E}_{a\sim\mu(\cdot|s)}\left[Q^{\mu}(s,a)-V^{\mu}(s)\right]\\ =& \mathbb{E}_{a\sim\mu(\cdot|s)}\left[Q^{\mu}(s,a)\right]-V^{\mu}(s)\\ =& V^{\mu}(s)-V^{\mu}(s)\\ =& 0,\end{aligned} \tag{31}$$

Therefore,

$$\begin{aligned}\left|\mathbb{E}_{a\sim\tilde{\mu}(\cdot|s)}\left[A^{\mu}(s,a)\right]\right| &= \left|\mathbb{E}_{a\sim\tilde{\mu}(\cdot|s)}\left[A^{\mu}(s,a)\right]-\mathbb{E}_{a\sim\mu(\cdot|s)}\left[A^{\mu}(s,a)\right]\right|\\ &\leq \left\|\tilde{\mu}-\mu\right\|_1\cdot\left\|A^{\mu}(s,a)\right\|_{\infty}\\ &= 2D_{\mathrm{TV}}(\tilde{\mu},\mu)[s]\cdot\max_{a}\left|A^{\mu}(s,a)\right|.\end{aligned} \tag{32}$$

This standard argument is widely used in the literature (Schulman et al., 2015; Gan et al., 2024). □

The preceding lemmas yield the following corollary, which we use repeatedly in the proofs.

Corollary E.4. *Given any two policies, $\tilde{\mu}$ and μ, the following bound holds:*

$$\left|\mathbb{E}_{\substack{s\sim d^{\tilde{\mu}}(\cdot)\\ a\sim\tilde{\mu}(\cdot|s)}}\left[A^{\mu}(s,a)\right]-\mathbb{E}_{\substack{s\sim d^{\mu}(\cdot)\\ a\sim\tilde{\mu}(\cdot|s)}}\left[A^{\mu}(s,a)\right]\right| \leq \frac{2\epsilon\gamma}{1-\gamma}\mathbb{E}_{s\sim d^{\mu}(\cdot)}\left[D_{\mathrm{TV}}(\tilde{\mu},\mu)[s]\right], \tag{33}$$

where $\epsilon=\max_s\left|\mathbb{E}_{a\sim\tilde{\mu}(\cdot|s)}\left[A^{\mu}(s,a)\right]\right|$.

Proof. We rewrite the difference between the expectations as

$$\left|\mathbb{E}_{\substack{s\sim d^{\tilde{\mu}}(\cdot)\\ a\sim\tilde{\mu}(\cdot|s)}}\left[A^{\mu}(s,a)\right]-\mathbb{E}_{\substack{s\sim d^{\mu}(\cdot)\\ a\sim\tilde{\mu}(\cdot|s)}}\left[A^{\mu}(s,a)\right]\right| = \left|\mathbb{E}_{s\sim d^{\tilde{\mu}}(\cdot)}\left\{\mathbb{E}_{a\sim\tilde{\mu}(\cdot|s)}\left[A^{\mu}(s,a)\right]\right\}-\mathbb{E}_{s\sim d^{\mu}(\cdot)}\left\{\mathbb{E}_{a\sim\tilde{\mu}(\cdot|s)}\left[A^{\mu}(s,a)\right]\right\}\right|, \tag{34}$$

Since the expectation $\mathbb{E}_{a\sim\tilde{\mu}(\cdot|s)}\left[A^{\mu}(s,a)\right]$ is a function of s, we have

$$\left|\mathbb{E}_{s\sim d^{\tilde{\mu}}(\cdot)}\left\{\mathbb{E}_{a\sim\tilde{\mu}(\cdot|s)}\left[A^{\mu}(s,a)\right]\right\}-\mathbb{E}_{s\sim d^{\mu}(\cdot)}\left\{\mathbb{E}_{a\sim\tilde{\mu}(\cdot|s)}\left[A^{\mu}(s,a)\right]\right\}\right| \leq \left\|d^{\tilde{\mu}}-d^{\mu}\right\|_1\cdot\left\|\mathbb{E}_{a\sim\tilde{\mu}(\cdot|s)}\left[A^{\mu}(s,a)\right]\right\|_{\infty}. \tag{35}$$

Applying Lemma E.2, we obtain

$$\left\|d^{\tilde{\mu}}-d^{\mu}\right\|_1\cdot\left\|\mathbb{E}_{a\sim\tilde{\mu}(\cdot|s)}\left[A^{\mu}(s,a)\right]\right\|_{\infty} = \left\|d^{\tilde{\mu}}-d^{\mu}\right\|_1\cdot\epsilon \leq \frac{2\epsilon\gamma}{1-\gamma}\mathbb{E}_{s\sim d^{\mu}(\cdot)}\left[D_{\mathrm{TV}}(\tilde{\mu},\mu)[s]\right], \tag{36}$$

This concludes the proof. □

E.1 Proof of Lemma 3.2

Lemma 3.2. *For any policy π, define its underlying policy by $\mu_f(\cdot\mid s_t)=\pi(\cdot\mid f(s_t))$. Then*

$$\eta(\pi)=\frac{1}{1-\gamma}\mathbb{E}_{\substack{f\sim p_{\mathrm{train}}(\cdot)\\ s\sim d^{\mu_f}(\cdot)\\ a\sim\mu_f(\cdot|s)}}\left[r(s,a)\right],\quad \zeta(\pi)=\frac{1}{1-\gamma}\mathbb{E}_{\substack{f\sim p(\cdot)\\ s\sim d^{\mu_f}(\cdot)\\ a\sim\mu_f(\cdot|s)}}\left[r(s,a)\right]. \tag{37}$$

Proof. By the definitions of training and generalization performance in equation 3, we have

$$\eta(\pi)=\mathbb{E}_{f\sim p_{\mathrm{train}}(\cdot),\tau_f\sim\pi}\left[\sum_{t=0}^{\infty}\gamma^t r_f(o_t^f,a_t)\right],\quad \zeta(\pi)=\mathbb{E}_{f\sim p(\cdot),\tau_f\sim\pi}\left[\sum_{t=0}^{\infty}\gamma^t r_f(o_t^f,a_t)\right]. \tag{38}$$

To prove Lemma 3.2, it suffices to show that, for any given $f \in \mathcal{F}$, the following equation holds:

$$\frac{1}{1-\gamma}\mathbb{E}_{\substack{s\sim d^{\mu_f}(\cdot)\\ a\sim\mu_f(\cdot|s)}}[r(s,a)] = \mathbb{E}_{\tau_f\sim\pi}\left[\sum_{t=0}^{\infty}\gamma^t r_f(o_t^f, a_t)\right]. \tag{39}$$

By the definition of the normalized discounted visitation distribution $d^{\mu}(s) = (1-\gamma)\sum_{t=0}^{\infty}\gamma^t\mathbb{P}(s_t = s \mid \mu)$, we have

$$\begin{aligned}\frac{1}{1-\gamma}\mathbb{E}_{\substack{s\sim d^{\mu_f}(\cdot)\\ a\sim\mu_f(\cdot|s)}}[r(s,a)] &= \frac{1}{1-\gamma}\sum_{s\in\mathcal{S}}(1-\gamma)\sum_{t=0}^{\infty}\gamma^t\mathbb{P}(s_t = s \mid \mu_f)\sum_{a\in\mathcal{A}}\mu_f(a\mid s)\cdot r(s,a)\\ &= \sum_{t=0}^{\infty}\sum_{s\in\mathcal{S}}\mathbb{P}(s_t = s\mid\mu_f)\sum_{a\in\mathcal{A}}\mu_f(a\mid s)\cdot\gamma^t r(s,a)\end{aligned} \tag{40}$$

Next, by Assumption 2.2, we obtain

$$\begin{aligned}\frac{1}{1-\gamma}\mathbb{E}_{\substack{s\sim d^{\mu_f}(\cdot)\\ a\sim\mu_f(\cdot|s)}}[r(s,a)] &= \sum_{t=0}^{\infty}\sum_{s\in\mathcal{S}}\mathbb{P}(s_t = s\mid\mu_f)\sum_{a\in\mathcal{A}}\mu_f(a\mid s)\cdot\gamma^t r(s,a)\\ &= \sum_{t=0}^{\infty}\sum_{s\in\mathcal{S}}\mathbb{P}(f(s_t) = f(s)\mid\mu_f)\sum_{a\in\mathcal{A}}\pi(a\mid f(s))\cdot\gamma^t r_f(f(s),a)\\ &\overset{f(s)=o^f, f(s_t)=o_t^f}{=} \sum_{t=0}^{\infty}\sum_{o^f\in\mathcal{O}_f}\mathbb{P}(o_t^f = o^f\mid\pi)\sum_{a\in\mathcal{A}}\pi(a\mid o^f)\cdot\gamma^t r_f(o^f,a)\\ &= \mathbb{E}_{\tau_f\sim\pi}\left[\sum_{t=0}^{\infty}\gamma^t r_f(o_t^f, a_t)\right].\end{aligned} \tag{41}$$

This concludes the proof. □

E.2 Proof of Theorem 3.4

Theorem 3.4. *For any policies $\tilde{\pi}$ and π:*

$$\begin{aligned}\zeta(\tilde{\pi}) \geq L_\pi(\tilde{\pi}) &- \frac{2r_{\max}(1-Z)}{1-\gamma} - \frac{2\gamma\epsilon_{\text{train}}}{(1-\gamma)^2}\mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot)\\ s\sim d^{\mu_f}(\cdot)}}[D_{\text{TV}}(\tilde{\mu}_f,\mu_f)[s]]\\ &- \frac{2\delta_{\text{train}}(1-Z)}{1-\gamma}\mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot)\\ s\sim d^{\tilde{\mu}_f}(\cdot)}}[D_{\text{TV}}(\tilde{\mu}_f,\mu_f)[s]]\\ &- \frac{2\delta_{\text{eval}}(1-Z)}{1-\gamma}\mathbb{E}_{\substack{f\sim p_{\text{eval}}(\cdot)\\ s\sim d^{\tilde{\mu}_f}(\cdot)}}[D_{\text{TV}}(\tilde{\mu}_f,\mu_f)[s]],\end{aligned} \tag{42}$$

where $r_{\max} = \max_{s,a}|r(s,a)|$, $\delta_{\text{train}} = \max_{f\in\mathcal{F}_{\text{train}}}\{\max_{s,a}|A^{\mu_f}(s,a)|\}$ *and* $\delta_{\text{eval}} = \max_{f\in\mathcal{F}_{\text{eval}}}\{\max_{s,a}|A^{\mu_f}(s,a)|\}$.

Proof. We begin with the first-order approximation of the training performance (Schulman et al., 2015), given by

$$L_\pi(\tilde{\pi}) = \eta(\pi) + \frac{1}{1-\gamma}\mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot)\\ s\sim d^{\mu_f}(\cdot)\\ a\sim\tilde{\mu}_f(\cdot|s)}}[A^{\mu_f}(s,a)]. \tag{43}$$

We next bound the difference between $\zeta(\tilde{\pi})$ and $L_\pi(\tilde{\pi})$. By Lemma E.1,

$$
\begin{aligned}
&|\zeta(\tilde{\pi}) - L_\pi(\tilde{\pi})| \\
=&\left|\zeta(\pi) - \eta(\pi) + \frac{1}{1-\gamma}\mathbb{E}_{\substack{f\sim p(\cdot)\\ s\sim d^{\tilde{\mu}_f}(\cdot)\\ a\sim\tilde{\mu}_f(\cdot|s)}}\left[A^{\mu_f}(s,a)\right] - \frac{1}{1-\gamma}\mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot)\\ s\sim d^{\mu_f}(\cdot)\\ a\sim\tilde{\mu}_f(\cdot|s)}}\left[A^{\mu_f}(s,a)\right]\right| \\
=&\frac{1}{1-\gamma}\left|\mathbb{E}_{\substack{f\sim p(\cdot)\\ s\sim d^{\mu_f}(\cdot)\\ a\sim\mu_f(\cdot|s)}}\left[r(s,a)\right] - \mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot)\\ s\sim d^{\mu_f}(\cdot)\\ a\sim\mu_f(\cdot|s)}}\left[r(s,a)\right] + \mathbb{E}_{\substack{f\sim p(\cdot)\\ s\sim d^{\tilde{\mu}_f}(\cdot)\\ a\sim\tilde{\mu}_f(\cdot|s)}}\left[A^{\mu_f}(s,a)\right] - \mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot)\\ s\sim d^{\mu_f}(\cdot)\\ a\sim\tilde{\mu}_f(\cdot|s)}}\left[A^{\mu_f}(s,a)\right]\right| \\
\leq&\frac{1}{1-\gamma}\left\{\left|\mathbb{E}_{\substack{f\sim p(\cdot)\\ s\sim d^{\mu_f}(\cdot)\\ a\sim\mu_f(\cdot|s)}}\left[r(s,a)\right] - \mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot)\\ s\sim d^{\mu_f}(\cdot)\\ a\sim\mu_f(\cdot|s)}}\left[r(s,a)\right]\right| + \left|\mathbb{E}_{\substack{f\sim p(\cdot)\\ s\sim d^{\tilde{\mu}_f}(\cdot)\\ a\sim\tilde{\mu}_f(\cdot|s)}}\left[A^{\mu_f}(s,a)\right] - \mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot)\\ s\sim d^{\mu_f}(\cdot)\\ a\sim\tilde{\mu}_f(\cdot|s)}}\left[A^{\mu_f}(s,a)\right]\right|\right\}.
\end{aligned}
\tag{44}
$$

We bound these two terms separately. To simplify the notation, let

$$
g(f) = \mathbb{E}_{s\sim d^{\mu_f}(\cdot), a\sim\mu_f(\cdot|s)}\left[r(s,a)\right]. \tag{45}
$$

The first term can then be rewritten as

$$
\left|\mathbb{E}_{\substack{f\sim p(\cdot)\\ s\sim d^{\mu_f}(\cdot)\\ a\sim\mu_f(\cdot|s)}}\left[r(s,a)\right] - \mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot)\\ s\sim d^{\mu_f}(\cdot)\\ a\sim\mu_f(\cdot|s)}}\left[r(s,a)\right]\right| = \left|\mathbb{E}_{f\sim p(\cdot)}\left[g(f)\right] - \mathbb{E}_{f\sim p_{\text{train}}(\cdot)}\left[g(f)\right]\right|, \tag{46}
$$

Therefore,

$$
\left|\mathbb{E}_{f\sim p(\cdot)}\left[g(f)\right] - \mathbb{E}_{f\sim p_{\text{train}}(\cdot)}\left[g(f)\right]\right| = \left|\int_{\mathcal{F}} p(f)\cdot g(f)\mathrm{d}f - \int_{\mathcal{F}_{\text{train}}} p_{\text{train}}(f)\cdot g(f)\mathrm{d}f\right|. \tag{47}
$$

Next, by Assumption 3.1,

$$
\begin{aligned}
&\left|\int_{\mathcal{F}} p(f)\cdot g(f)\mathrm{d}f - \int_{\mathcal{F}_{\text{train}}} p_{\text{train}}(f)\cdot g(f)\mathrm{d}f\right| = \left|\int_{\mathcal{F}} p(f)\cdot g(f)\mathrm{d}f - \int_{\mathcal{F}_{\text{train}}} \frac{p(f)}{Z}\cdot g(f)\mathrm{d}f\right| \\
=&\left|\int_{\mathcal{F}_{\text{train}}} p(f)\cdot g(f)\mathrm{d}f - \int_{\mathcal{F}_{\text{train}}} \frac{p(f)}{Z}\cdot g(f)\mathrm{d}f + \int_{\mathcal{F}\setminus\mathcal{F}_{\text{train}}} p(f)\cdot g(f)\mathrm{d}f\right| \\
=&\left|\int_{\mathcal{F}_{\text{train}}} \frac{Z-1}{Z} p(f)\cdot g(f)\mathrm{d}f + \int_{\mathcal{F}\setminus\mathcal{F}_{\text{train}}} p(f)\cdot g(f)\mathrm{d}f\right|,
\end{aligned}
\tag{48}
$$

where $Z = \int_{\mathcal{F}_{\text{train}}} p(f)\mathrm{d}f \leq 1$. Hence,

$$
\begin{aligned}
&\left|\int_{\mathcal{F}_{\text{train}}} \frac{Z-1}{Z} p(f)\cdot g(f)\mathrm{d}f + \int_{\mathcal{F}\setminus\mathcal{F}_{\text{train}}} p(f)\cdot g(f)\mathrm{d}f\right| \\
\leq&\left|\int_{\mathcal{F}_{\text{train}}} \frac{Z-1}{Z} p(f)\cdot g(f)\mathrm{d}f\right| + \left|\int_{\mathcal{F}\setminus\mathcal{F}_{\text{train}}} p(f)\cdot g(f)\mathrm{d}f\right| \\
\leq&\frac{1-Z}{Z}\left|\int_{\mathcal{F}_{\text{train}}} p(f)\cdot g(f)\mathrm{d}f\right| + \left|\int_{\mathcal{F}\setminus\mathcal{F}_{\text{train}}} p(f)\cdot g(f)\mathrm{d}f\right|.
\end{aligned}
\tag{49}
$$

Moreover,

$$
\begin{aligned}
|g(f)| = \left| \mathbb{E}_{\substack{s\sim d^{\mu_f}(\cdot) \\ a\sim\mu_f(\cdot|s)}} [r(s,a)] \right| &= \left| \sum_{s\in\mathcal{S}} (1-\gamma) \sum_{t=0}^{\infty} \gamma^t \mathbb{P}(s_t = s \mid \mu_f) \sum_{a\in\mathcal{A}} \mu_f(a \mid s) \cdot r(s,a) \right| \\
&\leq (1-\gamma) \sum_{t=0}^{\infty} \gamma^t \sum_{s\in\mathcal{S}} \mathbb{P}(s_t = s \mid \mu_f) \sum_{a\in\mathcal{A}} \mu_f(a \mid s) \cdot |r(s,a)| \qquad (50) \\
&\leq (1-\gamma) \sum_{t=0}^{\infty} \gamma^t r_{\max} = r_{\max},
\end{aligned}
$$

where $r_{\max} = \max_{s,a} |r(s,a)|$. We can therefore bound the first term as

$$
\begin{aligned}
\left| \mathbb{E}_{\substack{f\sim p(\cdot) \\ s\sim d^{\mu_f}(\cdot) \\ a\sim\mu_f(\cdot|s)}} [r(s,a)] - \mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot) \\ s\sim d^{\mu_f}(\cdot) \\ a\sim\mu_f(\cdot|s)}} [r(s,a)] \right| &\leq \frac{1-Z}{Z} \left| \int_{\mathcal{F}_{\text{train}}} p(f) \cdot g(f) \mathrm{d}f \right| + \left| \int_{\mathcal{F}\setminus\mathcal{F}_{\text{train}}} p(f) \cdot g(f) \mathrm{d}f \right| \\
&\leq \frac{1-Z}{Z} \int_{\mathcal{F}_{\text{train}}} p(f) \cdot |g(f)| \, \mathrm{d}f + \int_{\mathcal{F}\setminus\mathcal{F}_{\text{train}}} p(f) \cdot |g(f)| \, \mathrm{d}f \\
&\leq \frac{(1-Z) r_{\max}}{Z} \int_{\mathcal{F}_{\text{train}}} p(f) \mathrm{d}f + r_{\max} \int_{\mathcal{F}\setminus\mathcal{F}_{\text{train}}} p(f) \mathrm{d}f \\
&= \frac{(1-Z) r_{\max}}{Z} \cdot Z + r_{\max} \cdot (1-Z) = 2 r_{\max} (1-Z). \qquad (51)
\end{aligned}
$$

We now bound the second term, which can be expressed as

$$
\begin{aligned}
&\left| \mathbb{E}_{\substack{f\sim p(\cdot) \\ s\sim d^{\tilde{\mu}_f}(\cdot) \\ a\sim\tilde{\mu}_f(\cdot|s)}} [A^{\mu_f}(s,a)] - \mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot) \\ s\sim d^{\mu_f}(\cdot) \\ a\sim\tilde{\mu}_f(\cdot|s)}} [A^{\mu_f}(s,a)] \right| \\
&= \left| \mathbb{E}_{\substack{f\sim p(\cdot) \\ s\sim d^{\tilde{\mu}_f}(\cdot) \\ a\sim\tilde{\mu}_f(\cdot|s)}} [A^{\mu_f}(s,a)] - \mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot) \\ s\sim d^{\tilde{\mu}_f}(\cdot) \\ a\sim\tilde{\mu}_f(\cdot|s)}} [A^{\mu_f}(s,a)] + \mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot) \\ s\sim d^{\tilde{\mu}_f}(\cdot) \\ a\sim\tilde{\mu}_f(\cdot|s)}} [A^{\mu_f}(s,a)] - \mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot) \\ s\sim d^{\mu_f}(\cdot) \\ a\sim\tilde{\mu}_f(\cdot|s)}} [A^{\mu_f}(s,a)] \right| \\
&\leq \underbrace{\left| \mathbb{E}_{\substack{f\sim p(\cdot) \\ s\sim d^{\tilde{\mu}_f}(\cdot) \\ a\sim\tilde{\mu}_f(\cdot|s)}} [A^{\mu_f}(s,a)] - \mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot) \\ s\sim d^{\tilde{\mu}_f}(\cdot) \\ a\sim\tilde{\mu}_f(\cdot|s)}} [A^{\mu_f}(s,a)] \right|}_{\text{denote as } \Phi} + \underbrace{\left| \mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot) \\ s\sim d^{\tilde{\mu}_f}(\cdot) \\ a\sim\tilde{\mu}_f(\cdot|s)}} [A^{\mu_f}(s,a)] - \mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot) \\ s\sim d^{\mu_f}(\cdot) \\ a\sim\tilde{\mu}_f(\cdot|s)}} [A^{\mu_f}(s,a)] \right|}_{\text{denote as } \Psi}. \qquad (52)
\end{aligned}
$$

By Corollary E.4, Ψ can be bounded as

$$
\begin{aligned}
\Psi &= \left| \mathbb{E}_{f\sim p_{\text{train}}(\cdot)} \left\{ \mathbb{E}_{\substack{s\sim d^{\tilde{\mu}_f}(\cdot) \\ a\sim\tilde{\mu}_f(\cdot|s)}} [A^{\mu_f}(s,a)] - \mathbb{E}_{\substack{s\sim d^{\mu_f}(\cdot) \\ a\sim\tilde{\mu}_f(\cdot|s)}} [A^{\mu_f}(s,a)] \right\} \right| \\
&\leq \mathbb{E}_{f\sim p_{\text{train}}(\cdot)} \left\{ \left| \mathbb{E}_{\substack{s\sim d^{\tilde{\mu}_f}(\cdot) \\ a\sim\tilde{\mu}_f(\cdot|s)}} [A^{\mu_f}(s,a)] - \mathbb{E}_{\substack{s\sim d^{\mu_f}(\cdot) \\ a\sim\tilde{\mu}_f(\cdot|s)}} [A^{\mu_f}(s,a)] \right| \right\} \qquad (53) \\
&\leq \mathbb{E}_{f\sim p_{\text{train}}(\cdot)} \left\{ \frac{2\epsilon\gamma}{1-\gamma} \mathbb{E}_{s\sim d^{\mu_f}(\cdot)} [D_{\text{TV}}(\tilde{\mu}_f, \mu_f)[s]] \right\},
\end{aligned}
$$

where

$$
\epsilon = \max_s \left| \mathbb{E}_{a\sim\tilde{\mu}_f(\cdot|s)} [A^{\mu_f}(s,a)] \right|. \qquad (54)
$$

Defining

$$
\epsilon_{\text{train}} = \max_{f\in\mathcal{F}_{\text{train}}} \left\{ \max_s \left| \mathbb{E}_{a\sim\tilde{\mu}_f(\cdot|s)} [A^{\mu_f}(s,a)] \right| \right\}, \qquad (55)
$$

we obtain

$$\Psi \leq \frac{2\gamma\epsilon_{\text{train}}}{1-\gamma}\mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot)\\ s\sim d^{\mu_f}(\cdot)}}\left[D_{\text{TV}}(\tilde{\mu}_f,\mu_f)[s]\right]. \tag{56}$$

Next, with a slight abuse of notation, redefine $g(f)$ as

$$g(f) = \mathbb{E}_{\substack{s\sim d^{\tilde{\mu}_f}(\cdot)\\ a\sim\tilde{\mu}_f(\cdot|s)}}\left[A^{\mu_f}(s,a)\right], \tag{57}$$

Then Φ can be rewritten as

$$\Phi = \left|\mathbb{E}_{f\sim p(\cdot)}\left[g(f)\right] - \mathbb{E}_{f\sim p_{\text{train}}(\cdot)}\left[g(f)\right]\right|, \tag{58}$$

Proceeding as in equation 47, equation 48, equation 49, and equation 51, we obtain

$$\Phi \leq \frac{1-Z}{Z}\int_{\mathcal{F}_{\text{train}}} p(f)\cdot|g(f)|\,\mathrm{d}f + \int_{\mathcal{F}\setminus\mathcal{F}_{\text{train}}} p(f)\cdot|g(f)|\,\mathrm{d}f. \tag{59}$$

To bound $g(f)$ using Lemma E.3, first write

$$g(f) = \mathbb{E}_{\substack{s\sim d^{\tilde{\mu}_f}(\cdot)\\ a\sim\tilde{\mu}_f(\cdot|s)}}\left[A^{\mu_f}(s,a)\right] = \mathbb{E}_{s\sim d^{\tilde{\mu}_f}(\cdot)}\left\{\mathbb{E}_{a\sim\tilde{\mu}_f(\cdot|s)}\left[A^{\mu_f}(s,a)\right]\right\}, \tag{60}$$

Therefore,

$$\begin{aligned}|g(f)| &\leq \mathbb{E}_{s\sim d^{\tilde{\mu}_f}(\cdot)}\left[\left|\mathbb{E}_{a\sim\tilde{\mu}_f(\cdot|s)}\left[A^{\mu_f}(s,a)\right]\right|\right]\\ &\leq 2\mathbb{E}_{s\sim d^{\tilde{\mu}_f}(\cdot)}\left[D_{\text{TV}}(\tilde{\mu}_f,\mu_f)[s]\max_a\left|A^{\mu_f}(s,a)\right|\right].\end{aligned} \tag{61}$$

Let $\delta = \max_{s,a}|A^{\mu_f}(s,a)|$. Then

$$|g(f)| \leq 2\delta\mathbb{E}_{s\sim d^{\tilde{\mu}_f}(\cdot)}\left[D_{\text{TV}}(\tilde{\mu}_f,\mu_f)[s]\right], \tag{62}$$

Consequently,

$$\begin{aligned}\Phi \leq& \frac{1-Z}{Z}\int_{\mathcal{F}_{\text{train}}} p(f)\cdot|g(f)|\,\mathrm{d}f + \int_{\mathcal{F}\setminus\mathcal{F}_{\text{train}}} p(f)\cdot|g(f)|\,\mathrm{d}f\\ \leq& \frac{2\delta_{\text{train}}(1-Z)}{Z}\int_{\mathcal{F}_{\text{train}}} p(f)\cdot\mathbb{E}_{s\sim d^{\tilde{\mu}_f}(\cdot)}\left[D_{\text{TV}}(\tilde{\mu}_f,\mu_f)[s]\right]\mathrm{d}f\\ &+ 2\delta_{\text{eval}}\int_{\mathcal{F}\setminus\mathcal{F}_{\text{train}}} p(f)\cdot\mathbb{E}_{s\sim d^{\tilde{\mu}_f}(\cdot)}\left[D_{\text{TV}}(\tilde{\mu}_f,\mu_f)[s]\right]\mathrm{d}f\\ =& 2\delta_{\text{train}}(1-Z)\int_{\mathcal{F}_{\text{train}}}\frac{p(f)}{Z}\cdot\mathbb{E}_{s\sim d^{\tilde{\mu}_f}(\cdot)}\left[D_{\text{TV}}(\tilde{\mu}_f,\mu_f)[s]\right]\mathrm{d}f\\ &+ 2\delta_{\text{eval}}(1-Z)\int_{\mathcal{F}\setminus\mathcal{F}_{\text{train}}}\frac{p(f)}{1-Z}\cdot\mathbb{E}_{s\sim d^{\tilde{\mu}_f}(\cdot)}\left[D_{\text{TV}}(\tilde{\mu}_f,\mu_f)[s]\right]\mathrm{d}f\\ =& 2\delta_{\text{train}}(1-Z)\mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot)\\ s\sim d^{\tilde{\mu}_f}(\cdot)}}\left[D_{\text{TV}}(\tilde{\mu}_f,\mu_f)[s]\right] + 2\delta_{\text{eval}}(1-Z)\mathbb{E}_{\substack{f\sim p_{\text{eval}}(\cdot)\\ s\sim d^{\tilde{\mu}_f}(\cdot)}}\left[D_{\text{TV}}(\tilde{\mu}_f,\mu_f)[s]\right],\end{aligned} \tag{63}$$

where $\delta_{\text{train}} = \max_{f\in\mathcal{F}_{\text{train}}}\{\max_{s,a}|A^{\mu_f}(s,a)|\}$ and $\delta_{\text{eval}} = \max_{f\in\mathcal{F}_{\text{eval}}}\{\max_{s,a}|A^{\mu_f}(s,a)|\}$.

Finally, combining equation 44, equation 51, equation 52, equation 56, and equation 63, we have

$$\begin{aligned}|\zeta(\tilde{\pi}) - L_\pi(\tilde{\pi})| \leq& \frac{2r_{\max}(1-Z)}{1-\gamma} + \frac{2\gamma\epsilon_{\text{train}}}{(1-\gamma)^2}\mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot)\\ s\sim d^{\mu_f}(\cdot)}}\left[D_{\text{TV}}(\tilde{\mu}_f,\mu_f)[s]\right]\\ &+ \frac{2\delta_{\text{train}}(1-Z)}{1-\gamma}\mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot)\\ s\sim d^{\tilde{\mu}_f}(\cdot)}}\left[D_{\text{TV}}(\tilde{\mu}_f,\mu_f)[s]\right] + \frac{2\delta_{\text{eval}}(1-Z)}{1-\gamma}\mathbb{E}_{\substack{f\sim p_{\text{eval}}(\cdot)\\ s\sim d^{\tilde{\mu}_f}(\cdot)}}\left[D_{\text{TV}}(\tilde{\mu}_f,\mu_f)[s]\right],\end{aligned} \tag{64}$$

Therefore, the generalization performance satisfies the lower bound

$$\begin{aligned}\zeta(\tilde{\pi}) \geq& L_\pi(\tilde{\pi}) - \frac{2r_{\max}(1-Z)}{1-\gamma} - \frac{2\gamma\epsilon_{\text{train}}}{(1-\gamma)^2}\mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot)\\ s\sim d^{\mu_f}(\cdot)}}\left[D_{\text{TV}}(\tilde{\mu}_f,\mu_f)[s]\right]\\ &- \frac{2\delta_{\text{train}}(1-Z)}{1-\gamma}\mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot)\\ s\sim d^{\tilde{\mu}_f}(\cdot)}}\left[D_{\text{TV}}(\tilde{\mu}_f,\mu_f)[s]\right]\\ &- \frac{2\delta_{\text{eval}}(1-Z)}{1-\gamma}\mathbb{E}_{\substack{f\sim p_{\text{eval}}(\cdot)\\ s\sim d^{\tilde{\mu}_f}(\cdot)}}\left[D_{\text{TV}}(\tilde{\mu}_f,\mu_f)[s]\right],\end{aligned} \tag{65}$$

This concludes the proof. □

E.3 Proof of Theorem 3.3

Theorem 3.3. *Given any two policies, $\tilde{\pi}$ and π, the following bound holds:*

$$\eta(\tilde{\pi}) \geq L_{\pi}(\tilde{\pi}) - \frac{2\gamma\epsilon_{\text{train}}}{(1-\gamma)^2}\mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot)\\ s\sim d^{\mu_f}(\cdot)}}\left[D_{\text{TV}}(\tilde{\mu}_f, \mu_f)[s]\right], \tag{66}$$

where $\epsilon_{\text{train}} = \max_{f\in\mathcal{F}_{\text{train}}}\left\{\max_s \left|\mathbb{E}_{a\sim\tilde{\mu}_f(\cdot|s)}\left[A^{\mu_f}(s,a)\right]\right|\right\}$.

Proof. Using the definition and bound for Ψ above,

$$\begin{aligned}|\eta(\tilde{\pi}) - L_{\pi}(\tilde{\pi})| &= \frac{1}{1-\gamma}\left|\mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot)\\ s\sim d^{\tilde{\mu}_f}(\cdot)\\ a\sim\tilde{\mu}_f(\cdot|s)}}\left[A^{\mu_f}(s,a)\right] - \mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot)\\ s\sim d^{\mu_f}(\cdot)\\ a\sim\tilde{\mu}_f(\cdot|s)}}\left[A^{\mu_f}(s,a)\right]\right| = \frac{\Psi}{1-\gamma}\\ &\leq \frac{2\gamma\epsilon_{\text{train}}}{(1-\gamma)^2}\mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot)\\ s\sim d^{\mu_f}(\cdot)}}\left[D_{\text{TV}}(\tilde{\mu}_f, \mu_f)[s]\right],\end{aligned} \tag{67}$$

which implies

$$\eta(\tilde{\pi}) \geq L_{\pi}(\tilde{\pi}) - \frac{2\gamma\epsilon_{\text{train}}}{(1-\gamma)^2}\mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot)\\ s\sim d^{\mu_f}(\cdot)}}\left[D_{\text{TV}}(\tilde{\mu}_f, \mu_f)[s]\right], \tag{68}$$

This concludes the proof. □

E.4 Proof of Theorem 3.6

Theorem 3.6. *Given any two policies, $\tilde{\pi}$ and π, the following bounds hold:*

$$\mathfrak{D}_1 \leq \left(1 + \frac{2\gamma\sigma_{\text{train}}}{1-\gamma}\right)\mathfrak{D}_{\text{train}}, \quad \mathfrak{D}_2 \leq \left(1 + \frac{2\gamma\sigma_{\text{eval}}}{1-\gamma}\right)\underbrace{\mathbb{E}_{\substack{f\sim p_{\text{eval}}(\cdot)\\ s\sim d^{\mu_f}(\cdot)}}\left[D_{\text{TV}}(\tilde{\mu}_f, \mu_f)[s]\right]}_{\textit{denote it as } \mathfrak{D}_{\text{eval}}}. \tag{69}$$

where

$$\sigma_{\text{train}} = \max_{f\in\mathcal{F}_{\text{train}}}\max_{s\in\mathcal{S}} D_{\text{TV}}(\tilde{\mu}_f, \mu_f)[s], \quad \sigma_{\text{eval}} = \max_{f\in\mathcal{F}_{\text{eval}}}\max_{s\in\mathcal{S}} D_{\text{TV}}(\tilde{\mu}_f, \mu_f)[s]. \tag{70}$$

Proof. By Lemma E.2, we have

$$\begin{aligned}|\mathfrak{D}_1 - \mathfrak{D}_{\text{train}}| &= \left|\mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot)\\ s\sim d^{\tilde{\mu}_f}(\cdot)}}\left[D_{\text{TV}}(\tilde{\mu}_f, \mu_f)[s]\right] - \mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot)\\ s\sim d^{\mu_f}(\cdot)}}\left[D_{\text{TV}}(\tilde{\mu}_f, \mu_f)[s]\right]\right|\\ &= \left|\mathbb{E}_{f\sim p_{\text{train}}(\cdot)}\left\{\mathbb{E}_{s\sim d^{\tilde{\mu}_f}(\cdot)}\left[D_{\text{TV}}(\tilde{\mu}_f, \mu_f)[s]\right] - \mathbb{E}_{s\sim d^{\mu_f}(\cdot)}\left[D_{\text{TV}}(\tilde{\mu}_f, \mu_f)[s]\right]\right\}\right|\\ &\leq \mathbb{E}_{f\sim p_{\text{train}}(\cdot)}\left\{\left|\mathbb{E}_{s\sim d^{\tilde{\mu}_f}(\cdot)}\left[D_{\text{TV}}(\tilde{\mu}_f, \mu_f)[s]\right] - \mathbb{E}_{s\sim d^{\mu_f}(\cdot)}\left[D_{\text{TV}}(\tilde{\mu}_f, \mu_f)[s]\right]\right|\right\}\\ &\leq \mathbb{E}_{f\sim p_{\text{train}}(\cdot)}\left\{\left\|d^{\tilde{\mu}_f} - d^{\mu_f}\right\|_1 \cdot \left\|D_{\text{TV}}(\tilde{\mu}_f, \mu_f)[s]\right\|_{\infty}\right\}\\ &\leq \mathbb{E}_{f\sim p_{\text{train}}(\cdot)}\left\{\frac{2\gamma}{1-\gamma}\mathbb{E}_{s\sim d^{\mu_f}(\cdot)}\left[D_{\text{TV}}(\tilde{\mu}_f, \mu_f)[s]\right] \cdot \max_s D_{\text{TV}}(\tilde{\mu}_f, \mu_f)[s]\right\}\\ &\leq \frac{2\gamma\sigma_{\text{train}}}{1-\gamma}\mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot)\\ s\sim d^{\mu_f}(\cdot)}}\left[D_{\text{TV}}(\tilde{\mu}_f, \mu_f)[s]\right] = \frac{2\gamma\sigma_{\text{train}}}{1-\gamma}\cdot\mathfrak{D}_{\text{train}},\end{aligned} \tag{71}$$

Therefore,

$$\mathfrak{D}_1 \leq \left(1 + \frac{2\gamma\sigma_{\text{train}}}{1-\gamma}\right)\mathfrak{D}_{\text{train}}. \tag{72}$$

Similarly, using Lemma E.2 again, we have

$$
\begin{aligned}
|\mathfrak{D}_2 - \mathfrak{D}_{\text{eval}}| &= \left| \mathbb{E}_{\substack{f\sim p_{\text{eval}}(\cdot)\\ s\sim d^{\tilde{\mu}_f}(\cdot)}} [D_{\text{TV}}(\tilde{\mu}_f, \mu_f)[s]] - \mathbb{E}_{\substack{f\sim p_{\text{eval}}(\cdot)\\ s\sim d^{\mu_f}(\cdot)}} [D_{\text{TV}}(\tilde{\mu}_f, \mu_f)[s]] \right| \\
&= \left| \mathbb{E}_{f\sim p_{\text{eval}}(\cdot)} \left\{ \mathbb{E}_{s\sim d^{\tilde{\mu}_f}(\cdot)} [D_{\text{TV}}(\tilde{\mu}_f, \mu_f)[s]] - \mathbb{E}_{s\sim d^{\mu_f}(\cdot)} [D_{\text{TV}}(\tilde{\mu}_f, \mu_f)[s]] \right\} \right| \\
&\le \mathbb{E}_{f\sim p_{\text{eval}}(\cdot)} \left\{ \left| \mathbb{E}_{s\sim d^{\tilde{\mu}_f}(\cdot)} [D_{\text{TV}}(\tilde{\mu}_f, \mu_f)[s]] - \mathbb{E}_{s\sim d^{\mu_f}(\cdot)} [D_{\text{TV}}(\tilde{\mu}_f, \mu_f)[s]] \right| \right\} \\
&\le \mathbb{E}_{f\sim p_{\text{eval}}(\cdot)} \left\{ \left\| d^{\tilde{\mu}_f} - d^{\mu_f} \right\|_1 \cdot \left\| D_{\text{TV}}(\tilde{\mu}_f, \mu_f)[s] \right\|_\infty \right\} \\
&\le \mathbb{E}_{f\sim p_{\text{eval}}(\cdot)} \left\{ \frac{2\gamma}{1-\gamma} \mathbb{E}_{s\sim d^{\mu_f}(\cdot)} [D_{\text{TV}}(\tilde{\mu}_f, \mu_f)[s]] \cdot \max_s D_{\text{TV}}(\tilde{\mu}_f, \mu_f)[s] \right\} \\
&\le \frac{2\gamma\sigma_{\text{eval}}}{1-\gamma} \mathbb{E}_{\substack{f\sim p_{\text{eval}}(\cdot)\\ s\sim d^{\mu_f}(\cdot)}} [D_{\text{TV}}(\tilde{\mu}_f, \mu_f)[s]] = \frac{2\gamma\sigma_{\text{eval}}}{1-\gamma} \cdot \mathfrak{D}_{\text{eval}},
\end{aligned}
\tag{73}
$$

Therefore,

$$
\mathfrak{D}_2 \le \left(1 + \frac{2\gamma\sigma_{\text{eval}}}{1-\gamma} \right) \mathfrak{D}_{\text{eval}}, \tag{74}
$$

This concludes the proof. □

E.5 Proof of Theorem 3.9

Theorem 3.9. *Given any two policies, $\tilde{\pi}$ and π, assume that $\tilde{\pi}$ is $\mathcal{R}_{\tilde{\pi}}$-robust and π is $\mathcal{R}_{\pi}$-robust. Then the following bound holds:*

$$
\mathfrak{D}_{\text{eval}} \le \left(1 + \frac{2\gamma\sigma_{\text{train}}}{1-\gamma} \right) \mathcal{R}_{\pi} + \mathcal{R}_{\tilde{\pi}} + \mathfrak{D}_{\text{train}}. \tag{75}
$$

Proof. First, rewrite $\mathfrak{D}_{\text{eval}}$ as

$$
\mathfrak{D}_{\text{eval}} = \mathbb{E}_{\substack{\tilde{f}\sim p_{\text{eval}}(\cdot)\\ s\sim d^{\mu_{\tilde{f}}}(\cdot)}} \left[D_{\text{TV}}(\tilde{\mu}_{\tilde{f}}, \mu_{\tilde{f}})[s] \right]. \tag{76}
$$

For another $f \in \mathcal{F}_{\text{train}}$, repeatedly applying the triangle inequality gives the following compact chain. For readability, let $\Delta_f(s) := D_{\text{TV}}(\tilde{\mu}_f, \mu_f)[s]$. We write $\mathbb{E}_{f,\tilde{f}}$ for the product marginal $f \sim p_{\text{train}}$, $\tilde{f} \sim p_{\text{eval}}$; $\mathbb{E}_\star$ augments it with $s \sim d^{\mu_{\tilde{f}}}$, and $\mathbb{E}_\circ$ with $s \sim d^{\mu_f}$.

$$
\begin{aligned}
\mathfrak{D}_{\text{eval}} &= \mathbb{E}_{\tilde{f}\sim p_{\text{eval}},\, s\sim d^{\mu_{\tilde{f}}}} \left[D_{\text{TV}}(\tilde{\mu}_{\tilde{f}}, \mu_{\tilde{f}})[s] \right] \\
&\le \mathbb{E}_{\tilde{f}\sim p_{\text{eval}},\, s\sim d^{\mu_{\tilde{f}}}} \left[D_{\text{TV}}(\tilde{\mu}_{\tilde{f}}, \tilde{\mu}_f)[s] + \Delta_f(s) + D_{\text{TV}}(\mu_f, \mu_{\tilde{f}})[s] \right].
\end{aligned}
\tag{77}
$$

Taking expectation over $f \sim p_{\text{train}}$ gives

$$
\mathbb{E}_{f\sim p_{\text{train}}}[\mathfrak{D}_{\text{eval}}] \le \mathbb{E}_\star \left[D_{\text{TV}}(\tilde{\mu}_{\tilde{f}}, \tilde{\mu}_f)[s] + \Delta_f(s) + D_{\text{TV}}(\mu_f, \mu_{\tilde{f}})[s] \right]. \tag{78}
$$

Because $\mathfrak{D}_{\text{eval}}$ is independent of f, this reduces to

$$
\mathfrak{D}_{\text{eval}} \le \mathbb{E}_\star \left[D_{\text{TV}}(\tilde{\mu}_{\tilde{f}}, \tilde{\mu}_f)[s] + \Delta_f(s) + D_{\text{TV}}(\mu_f, \mu_{\tilde{f}})[s] \right]. \tag{79}
$$

Robustness of $\tilde{\pi}$ bounds the first term:

$$
\begin{aligned}
\mathbb{E}_\star \left[D_{\text{TV}}(\tilde{\mu}_{\tilde{f}}, \tilde{\mu}_f)[s] \right] &= \mathbb{E}_{f,\tilde{f}} \left[\sum_{s\in\mathcal{S}} d^{\mu_{\tilde{f}}}(s) D_{\text{TV}}(\tilde{\mu}_{\tilde{f}}, \tilde{\mu}_f)[s] \right] \\
&\le \mathbb{E}_{f,\tilde{f}} \left[\sum_{s\in\mathcal{S}} d^{\mu_{\tilde{f}}}(s) \mathcal{R}_{\tilde{\pi}} \right] = \mathcal{R}_{\tilde{\pi}}.
\end{aligned}
\tag{80}
$$

Similarly, symmetry of total variation and robustness of π bound the third term:

$$\begin{aligned}\mathbb{E}_\star\left[D_{\mathrm{TV}}(\mu_{\tilde{f}},\mu_f)[s]\right] &= \mathbb{E}_{f,\tilde{f}}\left[\sum_{s\in\mathcal{S}} d^{\mu_{\tilde{f}}}(s) D_{\mathrm{TV}}(\mu_{\tilde{f}},\mu_f)[s]\right] \\ &\le \mathbb{E}_{f,\tilde{f}}\left[\sum_{s\in\mathcal{S}} d^{\mu_{\tilde{f}}}(s)\mathcal{R}_\pi\right] = \mathcal{R}_\pi.\end{aligned} \tag{81}$$

We next bound the second term by relating it to $\mathfrak{D}_{\mathrm{train}}$. Since $\mathfrak{D}_{\mathrm{train}}$ is independent of $\tilde{f}$,

$$\mathfrak{D}_{\mathrm{train}} = \mathbb{E}_{f\sim p_{\mathrm{train}},\, s\sim d^{\mu_f}}[\Delta_f(s)] = \mathbb{E}_\circ[\Delta_f(s)]. \tag{82}$$

Therefore,

$$\begin{aligned}|\mathbb{E}_\star[\Delta_f(s)] - \mathfrak{D}_{\mathrm{train}}| &= |\mathbb{E}_\star[\Delta_f(s)] - \mathbb{E}_\circ[\Delta_f(s)]| \\ &\le \mathbb{E}_{f,\tilde{f}}\left[\left|\mathbb{E}_{s\sim d^{\mu_{\tilde{f}}}}[\Delta_f(s)] - \mathbb{E}_{s\sim d^{\mu_f}}[\Delta_f(s)]\right|\right].\end{aligned} \tag{83}$$

Observe that

$$\left|\mathbb{E}_{s\sim d^{\mu_{\tilde{f}}}}[\Delta_f(s)] - \mathbb{E}_{s\sim d^{\mu_f}}[\Delta_f(s)]\right| \le \|d^{\mu_{\tilde{f}}} - d^{\mu_f}\|_1 \|\Delta_f\|_\infty. \tag{84}$$

By Lemma E.2,

$$\|d^{\mu_{\tilde{f}}} - d^{\mu_f}\|_1 \le \frac{2\gamma}{1-\gamma}\mathbb{E}_{s\sim d^{\mu_f}(\cdot)}\left[D_{\mathrm{TV}}(\mu_{\tilde{f}},\mu_f)[s]\right], \tag{85}$$

Because π is $\mathcal{R}_\pi$-robust,

$$\begin{aligned}\|d^{\mu_{\tilde{f}}} - d^{\mu_f}\|_1 &\le \frac{2\gamma}{1-\gamma}\mathbb{E}_{s\sim d^{\mu_f}(\cdot)}\left[D_{\mathrm{TV}}(\mu_{\tilde{f}},\mu_f)[s]\right] \\ &= \frac{2\gamma}{1-\gamma}\sum_{s\in\mathcal{S}} d^{\mu_f}(s)\cdot D_{\mathrm{TV}}(\mu_{\tilde{f}},\mu_f)[s] \\ &\le \frac{2\gamma}{1-\gamma}\mathcal{R}_\pi.\end{aligned} \tag{86}$$

Therefore,

$$\begin{aligned}|\mathbb{E}_\star[\Delta_f(s)] - \mathfrak{D}_{\mathrm{train}}| &\le \mathbb{E}_{f,\tilde{f}}\left[\|d^{\mu_{\tilde{f}}} - d^{\mu_f}\|_1 \|\Delta_f\|_\infty\right] \\ &\le \frac{2\gamma}{1-\gamma}\mathcal{R}_\pi\, \mathbb{E}_{f\sim p_{\mathrm{train}}}\left[\|\Delta_f\|_\infty\right].\end{aligned} \tag{87}$$

We previously defined $\sigma_{\mathrm{train}} = \max_{f\in\mathcal{F}_{\mathrm{train}}}\|\Delta_f\|_\infty$, so

$$|\mathbb{E}_\star[\Delta_f(s)] - \mathfrak{D}_{\mathrm{train}}| \le \frac{2\gamma\sigma_{\mathrm{train}}}{1-\gamma}\mathcal{R}_\pi. \tag{88}$$

Thus, the second term is bounded by

$$\mathbb{E}_\star[\Delta_f(s)] \le \frac{2\gamma\sigma_{\mathrm{train}}}{1-\gamma}\mathcal{R}_\pi + \mathfrak{D}_{\mathrm{train}}. \tag{89}$$

Finally, combining equation 80, equation 81 and equation 89, we have

$$\mathfrak{D}_{\mathrm{eval}} \le \left(1 + \frac{2\gamma\sigma_{\mathrm{train}}}{1-\gamma}\right)\mathcal{R}_\pi + \mathcal{R}_{\tilde{\pi}} + \mathfrak{D}_{\mathrm{train}}, \tag{90}$$

This concludes the proof. □

E.6 Proof of Theorem 3.13

Theorem 3.13. *For any learner i and peer j,*

$$\mathcal{R}_{\pi_i} \le \min\left\{1, \mathcal{R}_{\pi_j} + \sqrt{2\mathcal{K}_{j\to i}}\right\}. \tag{91}$$

Proof. Fix any $s \in \mathcal{S}$ and $f, g \in \mathcal{F}$. The triangle inequality for total variation gives

$$D_{\mathrm{TV}}(\mu_{i,f}, \mu_{i,g})[s] \le D_{\mathrm{TV}}(\mu_{i,f}, \mu_{j,f})[s] + D_{\mathrm{TV}}(\mu_{j,f}, \mu_{j,g})[s] + D_{\mathrm{TV}}(\mu_{j,g}, \mu_{i,g})[s]. \tag{92}$$

Pinsker's inequality yields

$$D_{\mathrm{TV}}(P, Q) \le \sqrt{\frac{1}{2} D_{\mathrm{KL}}(P \| Q)}. \tag{93}$$

While KL divergence is asymmetric, total variation is symmetric. Thus,

$$D_{\mathrm{TV}}(\mu_{i,f}, \mu_{j,f})[s] = D_{\mathrm{TV}}(\mu_{j,f}, \mu_{i,f})[s] \le \sqrt{\frac{\mathcal{K}_{j \to i}}{2}}, \tag{94}$$

The same argument bounds the third term in equation 92. Moreover, by Definition 3.7, the middle term is at most $\mathcal{R}_{\pi_j}$. Hence,

$$D_{\mathrm{TV}}(\mu_{i,f}, \mu_{i,g})[s] \le \mathcal{R}_{\pi_j} + \sqrt{2\mathcal{K}_{j \to i}}. \tag{95}$$

Finally, taking the supremum over (s, f, g) and using $D_{\mathrm{TV}} \le 1$ proves equation 91. □

E.7 Proof of Theorem 3.14

Theorem 3.14. *For the same-view updates π_i^λ defined above,*

$$\mathcal{R}_{\pi_i^\lambda} \le (1 - 2\lambda)\mathcal{R}_{\pi_i} + 2\lambda \mathcal{R}_{\bar{\pi}}, \quad i \in \{1, 2\}.$$

The mean and maximum of the two robustness radii do not increase. For $\lambda > 0$, either quantity strictly decreases if and only if it exceeds $\mathcal{R}_{\bar{\pi}}$ before the update.

Proof. Fix the two policies before the update. In the notation of the original setting, define

$$\bar{\mu}_f(\cdot \mid s) = \frac{\mu_{1,f}(\cdot \mid s) + \mu_{2,f}(\cdot \mid s)}{2}, \quad \mu_{i,f}^\lambda(\cdot \mid s) = (1 - \lambda)\mu_{i,f}(\cdot \mid s) + \lambda \mu_{3-i,f}(\cdot \mid s).$$

These are the underlying policies induced by $\bar{\pi}$ and π_i^λ, respectively. The same number $\lambda \in [0, 1/2]$ is used at every observation, and both updates use the same frozen pair (π_1, π_2). In particular,

$$\mu_{i,f}^\lambda = (1 - 2\lambda)\mu_{i,f} + 2\lambda \bar{\mu}_f, \quad \frac{\mu_{1,f}^\lambda + \mu_{2,f}^\lambda}{2} = \bar{\mu}_f.$$

The coefficients in the first identity are nonnegative. For any fixed $s \in \mathcal{S}$ and $f, g \in \mathcal{F}$, the triangle inequality for the action-distribution difference therefore gives

$$\begin{aligned} D_{\mathrm{TV}}(\mu_{i,f}^\lambda, \mu_{i,g}^\lambda)[s] &= \frac{1}{2} \left\| (1 - 2\lambda)(\mu_{i,f} - \mu_{i,g})(\cdot \mid s) + 2\lambda(\bar{\mu}_f - \bar{\mu}_g)(\cdot \mid s) \right\|_1 \\ &\le (1 - 2\lambda) D_{\mathrm{TV}}(\mu_{i,f}, \mu_{i,g})[s] + 2\lambda D_{\mathrm{TV}}(\bar{\mu}_f, \bar{\mu}_g)[s]. \end{aligned}$$

Taking the supremum over (s, f, g) and applying Definition 3.7 proves the asserted bound. This step only uses an upper bound on each pointwise distance; it does not require the two suprema on the right to be attained at the same state and rendering pair.

Applying the same convexity argument to the average policy, before and after the update, yields

$$\mathcal{R}_{\bar{\pi}} \le \frac{\mathcal{R}_{\pi_1} + \mathcal{R}_{\pi_2}}{2}, \quad \mathcal{R}_{\bar{\pi}} \le \frac{\mathcal{R}_{\pi_1^\lambda} + \mathcal{R}_{\pi_2^\lambda}}{2}.$$

The second inequality uses the fact that the updated average policy is still $\bar{\pi}$. Combining these inequalities with the individual bounds gives the sandwich

$$\begin{aligned} \mathcal{R}_{\bar{\pi}} &\le \frac{\mathcal{R}_{\pi_1^\lambda} + \mathcal{R}_{\pi_2^\lambda}}{2} \\ &\le (1 - 2\lambda)\frac{\mathcal{R}_{\pi_1} + \mathcal{R}_{\pi_2}}{2} + 2\lambda \mathcal{R}_{\bar{\pi}} \le \frac{\mathcal{R}_{\pi_1} + \mathcal{R}_{\pi_2}}{2}. \end{aligned}$$

Similarly, since a maximum is at least the corresponding mean,

$$\begin{aligned}\mathcal{R}_{\bar{\pi}} &\le \max_{i\in\{1,2\}} \mathcal{R}_{\pi_i^\lambda} \\ &\le (1-2\lambda) \max_{i\in\{1,2\}} \mathcal{R}_{\pi_i} + 2\lambda \mathcal{R}_{\bar{\pi}} \le \max_{i\in\{1,2\}} \mathcal{R}_{\pi_i}.\end{aligned}$$

For either quantity, if its original value exceeds $\mathcal{R}_{\bar{\pi}}$ and $\lambda > 0$, its upper bound in the middle is strictly smaller than its original value. Conversely, if its original value equals $\mathcal{R}_{\bar{\pi}}$, the lower and upper bounds coincide, so it cannot strictly decrease. An original value below $\mathcal{R}_{\bar{\pi}}$ is excluded by the first sandwich inequalities. This proves both directions of the strict-decrease statements, without any assumption that a worst-case rendering pair exists. At $\lambda = 0$ both policies are unchanged; at $\lambda = 1/2$ both equal $\bar{\pi}$, so both collective radii equal $\mathcal{R}_{\bar{\pi}}$ exactly.

For completeness, the distributional update also has the stated same-view KL interpretation. Fix an observation o and minimize over an action distribution $\pi(\cdot \mid o)$, keeping the two original distributions fixed. Suppressing this common conditioning argument within the following display, one has

$$\begin{aligned}&(1-\lambda) D_{\mathrm{KL}}(\pi_i \| \pi) + \lambda D_{\mathrm{KL}}(\pi_{3-i} \| \pi) \\ &\quad = (1-\lambda) D_{\mathrm{KL}}(\pi_i \| \pi_i^\lambda) + \lambda D_{\mathrm{KL}}(\pi_{3-i} \| \pi_i^\lambda) + D_{\mathrm{KL}}(\pi_i^\lambda \| \pi).\end{aligned}$$

To verify the identity, insert $\pi_i^\lambda(a \mid o)$ into each logarithmic ratio. The terms depending on the candidate distribution combine as

$$\begin{aligned}&\sum_{a\in\mathcal{A}} \big[(1-\lambda)\pi_i(a \mid o) + \lambda \pi_{3-i}(a \mid o)\big] \log \frac{\pi_i^\lambda(a \mid o)}{\pi(a \mid o)} \\ &\quad = \sum_{a\in\mathcal{A}} \pi_i^\lambda(a \mid o) \log \frac{\pi_i^\lambda(a \mid o)}{\pi(a \mid o)} = D_{\mathrm{KL}}(\pi_i^\lambda \| \pi)[o].\end{aligned}$$

The remaining terms are independent of π. Nonnegativity of KL divergence, with equality only for identical distributions, thus makes $\pi_i^\lambda(\cdot \mid o)$ the unique minimizer over the full action simplex.

This argument also covers the support boundary. For $0 < \lambda \le 1/2$, the support of π_i^λ is the union of the two target supports, and

$$\pi_i^\lambda(a \mid o) \ge (1-\lambda)\pi_i(a \mid o), \;\; \pi_i^\lambda(a \mid o) \ge \lambda \pi_{3-i}(a \mid o).$$

Consequently, the two target-to-mixture divergences are bounded by $-\log(1-\lambda)$ and $-\log\lambda$, respectively. They remain finite even when the targets have disjoint supports. If a candidate assigns zero probability to an action having positive mixture probability, its objective and its divergence from the mixture are both infinite, as required by the identity. More generally, on a countable action space the identity holds with extended KL values: the target-to-mixture terms are finite, and an infinite remaining divergence cannot be a minimizer. Coordinates outside the mixture support contribute zero under the usual convention; assigning mass there cannot yield zero KL unless the whole candidate equals the mixture. At $\lambda = 0$, the zero-weighted peer term is omitted rather than interpreted as $0 \cdot \infty$, and the objective is simply $D_{\mathrm{KL}}(\pi_i \| \pi)$.

The reason that no robust teacher is needed can be seen directly from signed rendering errors. For a finite action set and a fixed (s, f, g), write

$$\Delta_{i,a} = \mu_{i,f}(a \mid s) - \mu_{i,g}(a \mid s), \;\; \sum_{a\in\mathcal{A}} \Delta_{i,a} = 0.$$

For each action, the elementary absolute-value identity is

$$|\Delta_{1,a} + \Delta_{2,a}| = |\Delta_{1,a}| + |\Delta_{2,a}| - 2\mathbb{I}\{\Delta_{1,a}\Delta_{2,a} < 0\} \min\{|\Delta_{1,a}|, |\Delta_{2,a}|\}.$$

Summing over actions gives the exact cancellation formula

$$\begin{aligned}D_{\mathrm{TV}}(\bar{\mu}_f, \bar{\mu}_g)[s] &= \frac{1}{4} \sum_{a\in\mathcal{A}} |\Delta_{1,a} + \Delta_{2,a}| \\ &= \frac{D_{\mathrm{TV}}(\mu_{1,f}, \mu_{1,g})[s] + D_{\mathrm{TV}}(\mu_{2,f}, \mu_{2,g})[s]}{2} \\ &\quad - \frac{1}{2} \sum_{\substack{a\in\mathcal{A} \\ \Delta_{1,a}\Delta_{2,a} < 0}} \min\{|\Delta_{1,a}|, |\Delta_{2,a}|\}.\end{aligned}$$

Thus the pointwise convexity inequality is strict exactly when at least one action has nonzero rendering errors of opposite signs. Opposite errors disappear partially or completely in the average; errors of the same sign do not cancel at that action. This is an action-by-action statement at a fixed rendering pair. Cancellation at only one pair need not give strict improvement of a global supremum, which is why the theorem states its global criterion in terms of $\mathcal{R}_{\bar{\pi}}$ rather than a particular pair.

For example, take one underlying state, two renderings $x = f(s)$ and $y = g(s)$, and two actions. For any $0 < d < 1/2$, set

$$\pi_1(1 \mid x) = \tfrac{1}{2} + d, \quad \pi_1(1 \mid y) = \tfrac{1}{2} - d,$$
$$\pi_2(1 \mid x) = \tfrac{1}{2} - d, \quad \pi_2(1 \mid y) = \tfrac{1}{2} + d.$$

The other action has the complementary probability. Both original radii are $2d$, but the average policy is independent of the rendering. Direct substitution therefore gives

$$\mathcal{R}_{\bar{\pi}} = 0, \;\; \mathcal{R}_{\pi_1^\lambda} = \mathcal{R}_{\pi_2^\lambda} = (1 - 2\lambda)2d.$$

Both peers improve for every $\lambda > 0$ even though neither was initially invariant. In contrast, if the peers have identical rendering differences at every (s, f, g), mixing leaves all those differences unchanged. The theorem consequently gives a collective guarantee, not an assertion that every peer improves individually: a peer whose radius is below $\mathcal{R}_{\bar{\pi}}$ need not improve.

Finally, all these identities concern frozen, synchronous mixing of action distributions. Allowing the mixing weight to depend on the observation would introduce different coefficients at $f(s)$ and $g(s)$, so the displayed pointwise convexity calculation would no longer have the stated form. Updating one peer first and using that new peer as the other target would also lose the preserved-average identity. A neural parameter update need not realize the pointwise KL minimizer, and a mixture of parameters need not be a mixture of action distributions. These distinctions do not alter the result proved here; they motivate the explicit residual in Theorem 3.15. □

E.8 Proof of Theorem 3.15

Theorem 3.15. *Given realized policies $\tilde{\pi}_i$, write $e_i(\cdot \mid o) = \tilde{\pi}_i(\cdot \mid o) - \pi_i^\lambda(\cdot \mid o)$ and define*

$$\varepsilon_i = \frac{1}{2} \sup_{s,f,g} \left\| e_i(\cdot \mid f(s)) - e_i(\cdot \mid g(s)) \right\|_1 .$$

Then

$$\mathcal{R}_{\tilde{\pi}_i} \le B_i := \min\{1, (1 - 2\lambda)\mathcal{R}_{\pi_i} + 2\lambda\mathcal{R}_{\bar{\pi}} + \varepsilon_i\}.$$

In particular, $\mathcal{R}_{\tilde{\pi}_i} < \mathcal{R}_{\pi_i}$ whenever $2\lambda(\mathcal{R}_{\pi_i} - \mathcal{R}_{\bar{\pi}}) > \varepsilon_i$.

Proof. Fix a realized update and its frozen reference pair. Define $\tilde{\mu}_{i,f}(\cdot \mid s) = \tilde{\pi}_i(\cdot \mid f(s))$. For any $s \in \mathcal{S}$ and $f, g \in \mathcal{F}$, subtracting the two endpoint identities for e_i gives

$$\begin{aligned}(\tilde{\mu}_{i,f} - \tilde{\mu}_{i,g})(\cdot \mid s) = (\mu_{i,f}^\lambda - \mu_{i,g}^\lambda)(\cdot \mid s) \\ + e_i(\cdot \mid f(s)) - e_i(\cdot \mid g(s)).\end{aligned}$$

The triangle inequality and its reverse form immediately imply

$$\begin{aligned}&\left| D_{\mathrm{TV}}(\tilde{\mu}_{i,f}, \tilde{\mu}_{i,g})[s] - D_{\mathrm{TV}}(\mu_{i,f}^\lambda, \mu_{i,g}^\lambda)[s] \right| \\ &\le \frac{1}{2} \left\| e_i(\cdot \mid f(s)) - e_i(\cdot \mid g(s)) \right\|_1 \le \varepsilon_i.\end{aligned}$$

Both pointwise distances are bounded functions of the same triple (s, f, g). Taking their suprema in each direction therefore gives

$$\left| \mathcal{R}_{\tilde{\pi}_i} - \mathcal{R}_{\pi_i^\lambda} \right| \le \varepsilon_i.$$

Combining the upper direction with Theorem 3.14, and independently using $\mathcal{R}_{\tilde{\pi}_i} \le 1$, proves the stated bound with B_i. Moreover,

$$\mathcal{R}_{\tilde{\pi}_i} \le \mathcal{R}_{\pi_i} - 2\lambda(\mathcal{R}_{\pi_i} - \mathcal{R}_{\bar{\pi}}) + \varepsilon_i < \mathcal{R}_{\pi_i}$$

under the strict inequality in the theorem. This is sufficient, not necessary: the two triangle inequalities can be strict, and a signed endpoint error can cancel an existing rendering difference instead of reinforcing it.

The residual is a rendering-dependent discrepancy, rather than the full size of a policy update. Indeed, a further endpoint triangle inequality yields

$$\begin{aligned}\varepsilon_i &\le \frac{1}{2}\sup_{s,f,g}\big(\|e_i(\cdot \mid f(s))\|_1 + \|e_i(\cdot \mid g(s))\|_1\big)\\ &\le 2\sup_{o\in\mathcal{O}} D_{\mathrm{TV}}(\tilde{\pi}_i, \pi_i^{\lambda})[o] \le 2.\end{aligned}$$

Here the observation supremum may be restricted to observations of the form $f(s)$. Although the difference of the two robustness radii is at most one, ε_i itself can be as large as two, since it compares two signed errors. Conversely, $\varepsilon_i = 0$ whenever the error is identical at all renderings of each underlying state, even if its size at an individual observation is nonzero. Keeping the endpoint difference in its definition avoids charging such a rendering-independent error.

The same argument also gives collective error bounds. The inequality for individual radii implies

$$\begin{aligned}\left|\frac{\mathcal{R}_{\tilde{\pi}_1}+\mathcal{R}_{\tilde{\pi}_2}}{2} - \frac{\mathcal{R}_{\pi_1^{\lambda}}+\mathcal{R}_{\pi_2^{\lambda}}}{2}\right| &\le \frac{\varepsilon_1+\varepsilon_2}{2},\\ \left|\max_{i\in\{1,2\}}\mathcal{R}_{\tilde{\pi}_i} - \max_{i\in\{1,2\}}\mathcal{R}_{\pi_i^{\lambda}}\right| &\le \max_{i\in\{1,2\}}\varepsilon_i.\end{aligned}$$

For the maximum, each radius on one side is bounded by its counterpart plus the largest residual, and the reverse inequality follows by exchanging the sides. Thus the collective bounds in the previous proof give

$$\begin{aligned}\frac{\mathcal{R}_{\tilde{\pi}_1}+\mathcal{R}_{\tilde{\pi}_2}}{2} &\le \min\left\{1, (1-2\lambda)\frac{\mathcal{R}_{\pi_1}+\mathcal{R}_{\pi_2}}{2} + 2\lambda\mathcal{R}_{\bar{\pi}} + \frac{\varepsilon_1+\varepsilon_2}{2}\right\},\\ \max_{i\in\{1,2\}}\mathcal{R}_{\tilde{\pi}_i} &\le \min\left\{1, (1-2\lambda)\max_{i\in\{1,2\}}\mathcal{R}_{\pi_i} + 2\lambda\mathcal{R}_{\bar{\pi}} + \max_{i\in\{1,2\}}\varepsilon_i\right\}.\end{aligned}$$

In particular, strict decrease of the mean or maximum is ensured, respectively, by

$$\begin{aligned}2\lambda\left(\frac{\mathcal{R}_{\pi_1}+\mathcal{R}_{\pi_2}}{2} - \mathcal{R}_{\bar{\pi}}\right) &> \frac{\varepsilon_1+\varepsilon_2}{2},\\ 2\lambda\left(\max_{i\in\{1,2\}}\mathcal{R}_{\pi_i} - \mathcal{R}_{\bar{\pi}}\right) &> \max_{i\in\{1,2\}}\varepsilon_i.\end{aligned}$$

These conditions too are sufficient rather than necessary. Unlike exact mixing, a realized update need not preserve the average policy, so the equality-based converse from Theorem 3.14 cannot be imposed on this residual bound.

All inequalities so far hold conditionally on the actual realized policies. They require neither a particular optimizer nor a claim that its residual is small. The residual includes the effects of the RL objective, shared parameterization, and the optimization step. In particular, the reference mixing weight λ is not the KL coefficient α of MDPO. A small mean KL loss cannot by itself establish a small uniform residual: Pinsker's inequality followed by Jensen's inequality controls the mean TV distance under the same observation distribution, not its supremum over observations. An observation with zero training mass is unconstrained by that mean; even positive but arbitrarily small mass can hide a large pointwise error. No coverage assumption converting such a mean into a supremum is used here.

An ordinary logit calculation makes explicit why descent of a same-view KL objective cannot replace the residual argument. Consider one underlying state, two renderings x, y, and actions $0, 1$. Give each learner an independent logit at each observation, and set

$$\begin{aligned}\pi_1(1 \mid x) &= 0.95, \quad & \pi_1(1 \mid y) &= 0.9,\\ \pi_2(1 \mid x) &= 0.01, \quad & \pi_2(1 \mid y) &= 0.02.\end{aligned}$$

For $o \in \{x, y\}$, write $u_o = \pi_1(1 \mid o)$ and $v_o = \pi_2(1 \mid o)$, and parameterize them by $u_o = (1 + e^{-\theta_{1,o}})^{-1}$ and $v_o = (1 + e^{-\theta_{2,o}})^{-1}$. Consider the joint, non-detached objective

$$J(\theta) = \sum_{o \in \{x,y\}} \left[D_{\mathrm{KL}}(\pi_1 \| \pi_2)[o] + D_{\mathrm{KL}}(\pi_2 \| \pi_1)[o] \right].$$

An average over the two observations only rescales the step size. Define the scalar log-odds difference $L_o = \log \frac{u_o(1-v_o)}{v_o(1-u_o)}$. Expanding the two binary divergences shows that their sum at observation o is $(u_o - v_o)L_o$. Differentiating with respect to both logits, including both appearances of each policy, gives the exact joint gradients

$$\frac{\partial J}{\partial \theta_{1,o}} = u_o(1-u_o)L_o + (u_o - v_o), \quad \frac{\partial J}{\partial \theta_{2,o}} = -v_o(1-v_o)L_o - (u_o - v_o).$$

Along the simultaneous gradient step $\theta(h) = \theta - h\nabla J(\theta)$, let a dot denote differentiation at $h = 0$. The action probabilities then satisfy

$$\dot{u}_o = -[u_o(1-u_o)]^2 L_o - u_o(1-u_o)(u_o - v_o),$$
$$\dot{v}_o = [v_o(1-v_o)]^2 L_o + v_o(1-v_o)(u_o - v_o).$$

At the specified starting point, the two radii are $\mathcal{R}_{\pi_1} = u_x - u_y = 1/20$ and $\mathcal{R}_{\pi_2} = v_y - v_x = 1/100$. These signs remain unchanged for sufficiently small h. The factors entering the derivatives are

$$u_x(1-u_x) = \frac{19}{400}, \qquad u_y(1-u_y) = \frac{9}{100},$$
$$v_x(1-v_x) = \frac{99}{10000}, \qquad v_y(1-v_y) = \frac{49}{2500},$$
$$u_x - v_x = \frac{47}{50}, \qquad u_y - v_y = \frac{22}{25},$$
$$L_x = \log 1881, \qquad L_y = \log 441.$$

In particular, $0 < L_x < 2L_y$ because $1 < 1881 < 441^2$. Substitution into the radius derivatives gives

$$\frac{\mathrm{d}}{\mathrm{d}h}\mathcal{R}_{\pi_1(h)}\bigg|_{h=0} = \frac{691}{20000} + \frac{81}{10000}L_y - \frac{361}{160000}L_x > \frac{691}{20000} + \frac{287}{80000}L_y > 0,$$
$$\frac{\mathrm{d}}{\mathrm{d}h}\mathcal{R}_{\pi_2(h)}\bigg|_{h=0} = \frac{3971}{500000} + \frac{2401}{6250000}L_y - \frac{9801}{100000000}L_x > \frac{3971}{500000} + \frac{9407}{50000000}L_y > 0.$$

On the other hand, the gradient is nonzero and

$$\frac{\mathrm{d}}{\mathrm{d}h}J(\theta - h\nabla J(\theta))\bigg|_{h=0} = -\|\nabla J(\theta)\|_2^2 < 0.$$

All probabilities are strictly interior, so smoothness and the strict derivative signs imply that, for every sufficiently small positive step, the joint KL objective decreases while both robustness radii increase. The example involves only two equally weighted observations and unrestricted ordinary logits; its failure is not caused by an unseen observation or a singular support.

For comparison, the same starting policies also exhibit this failure under a separate frozen-target convention. If each learner minimizes the sum of $D_{\mathrm{KL}}(\pi_{3-i}^{\mathrm{old}} \| \pi_i)[o]$ with its peer detached, the simultaneous negative logit gradients at the starting point are $\dot{\theta}_{1,o} = v_o - u_o$ and $\dot{\theta}_{2,o} = u_o - v_o$. The corresponding derivatives are therefore

$$\frac{\mathrm{d}}{\mathrm{d}h}\mathcal{R}_{\pi_1(h)}\bigg|_{h=0} = \frac{9}{100}\frac{22}{25} - \frac{19}{400}\frac{47}{50} = \frac{691}{20000} > 0,$$
$$\frac{\mathrm{d}}{\mathrm{d}h}\mathcal{R}_{\pi_2(h)}\bigg|_{h=0} = \frac{49}{2500}\frac{22}{25} - \frac{99}{10000}\frac{47}{50} = \frac{3971}{500000} > 0.$$

Each fixed-target loss decreases under its own sufficiently small gradient step, but both rendering radii again increase. This separate calculation is not an assumption about whether the original MDPO implementation detaches its peers; the preceding joint calculation already differentiates through both policies. Neither calculation is an additional hypothesis of the theorem. They only rule out an unconditional inference from ordinary same-view KL descent to robustness improvement, and explain why an actual parameter update must retain its residual rather than being identified with the exact distributional mixture. □

E.9 Proof of Corollary 3.16

Corollary 3.16. *For the policies and B_i above, let $\mathfrak{D}^{(i)}_{\text{train}}$ and $\epsilon^{(i)}_{\text{train}}$ be the quantities in Theorem 3.3 for $(\pi_i, \tilde{\pi}_i)$. With $C = 2r_{\max}(1-Z)/(1-\gamma)$,*

$$\zeta(\tilde{\pi}_i) \geq L_{\pi_i}(\tilde{\pi}_i) - \frac{2\gamma\epsilon^{(i)}_{\text{train}}}{(1-\gamma)^2}\mathfrak{D}^{(i)}_{\text{train}} - C\min\left\{1, \frac{B_i}{1-\gamma}\right\}.$$

This bound is no weaker than Corollary 3.10 with B_i substituted for $\mathcal{R}_{\tilde{\pi}_i}$ at the same policy pair.

Proof. Using the normalized discounted occupancies of the original setting, fix a candidate policy $\tilde{\pi}$ and define its underlying policies and rendering-specific returns by

$$\tilde{\mu}_f(\cdot \mid s) = \tilde{\pi}(\cdot \mid f(s)), \quad \eta_f(\tilde{\pi}) = \frac{1}{1-\gamma}\mathbb{E}_{\substack{s\sim d^{\tilde{\mu}_f}(\cdot)\\ a\sim\tilde{\mu}_f(\cdot|s)}}[r(s,a)].$$

We use the state and action sums of that setting throughout. By Assumption 2.2, all these policies act on the same underlying transition law, reward function, and initial-state distribution. Thus a comparison between f and g is a comparison of two policies in one MDP, not of rewards or initial conditions from different MDPs.

For fixed $f, g \in \mathcal{F}$, apply the performance-difference identity to $\tilde{\mu}_f$ as the reference policy and $\tilde{\mu}_g$ as the new policy. This yields

$$\eta_g(\tilde{\pi}) - \eta_f(\tilde{\pi}) = \frac{1}{1-\gamma}\mathbb{E}_{\substack{s\sim d^{\tilde{\mu}_g}(\cdot)\\ a\sim\tilde{\mu}_g(\cdot|s)}}\left[A^{\tilde{\mu}_f}(s,a)\right].$$

In particular, the occupancy in this identity belongs to the new policy induced by g. To see the identity directly, along a trajectory generated by $\tilde{\mu}_g$ the conditional expectation of $r(s_t,a_t) + \gamma V^{\tilde{\mu}_f}(s_{t+1}) - V^{\tilde{\mu}_f}(s_t)$ is $A^{\tilde{\mu}_f}(s_t,a_t)$. Multiplying by γ^t and summing telescopes the value terms. The terminal value contribution tends to zero because $|V^{\tilde{\mu}_f}| \leq r_{\max}/(1-\gamma)$, while the initial value has expectation $\eta_f(\tilde{\pi})$ under the common initial distribution. This proves the displayed identity with the normalized occupancy and its factor $1/(1-\gamma)$.

We next control the action expectation of this advantage without paying twice for an arbitrary value-function offset. Since the reference policy has zero expected advantage at each state,

$$\begin{aligned}\mathbb{E}_{a\sim\tilde{\mu}_g(\cdot|s)}[A^{\tilde{\mu}_f}(s,a)] &= \sum_{a\in\mathcal{A}}\big(\tilde{\mu}_g(a\mid s) - \tilde{\mu}_f(a\mid s)\big)Q^{\tilde{\mu}_f}(s,a)\\ &= \sum_{a\in\mathcal{A}}\big(\tilde{\mu}_g(a\mid s) - \tilde{\mu}_f(a\mid s)\big)\big(Q^{\tilde{\mu}_f}(s,a) - c_f(s)\big),\end{aligned}$$

where $c_f(s) = \frac{1}{2}\big(\sup_a Q^{\tilde{\mu}_f}(s,a) + \inf_a Q^{\tilde{\mu}_f}(s,a)\big)$. Subtraction of this midpoint is valid because the action-distribution difference sums to zero. Also, bounded rewards give

$$|Q^{\tilde{\mu}_f}(s,a)| \leq \frac{r_{\max}}{1-\gamma}, \quad \sup_a Q^{\tilde{\mu}_f}(s,a) - \inf_a Q^{\tilde{\mu}_f}(s,a) \leq \frac{2r_{\max}}{1-\gamma}.$$

The midpoint argument therefore bounds the expectation by the action-value span times TV:

$$\begin{aligned}\left|\mathbb{E}_{a\sim\tilde{\mu}_g(\cdot|s)}[A^{\tilde{\mu}_f}(s,a)]\right| &\leq \frac{\sup_a Q^{\tilde{\mu}_f}(s,a) - \inf_a Q^{\tilde{\mu}_f}(s,a)}{2}\sum_{a\in\mathcal{A}}|\tilde{\mu}_g(a\mid s) - \tilde{\mu}_f(a\mid s)|\\ &\leq \frac{2r_{\max}}{1-\gamma}D_{\text{TV}}(\tilde{\mu}_g,\tilde{\mu}_f)[s] \leq \frac{2r_{\max}}{1-\gamma}\mathcal{R}_{\tilde{\pi}}.\end{aligned}$$

Since $d^{\tilde{\mu}_g}$ is normalized, substitution into the performance-difference identity gives

$$|\eta_g(\tilde{\pi}) - \eta_f(\tilde{\pi})| \leq \frac{2r_{\max}}{(1-\gamma)^2}\mathcal{R}_{\tilde{\pi}}.$$

Independently, every rendering-specific return lies in $[-r_{\max}/(1-\gamma), r_{\max}/(1-\gamma)]$. Hence its difference from any other such return is at most $2r_{\max}/(1-\gamma)$. Combining these two independent bounds, rather than enlarging the value-span estimate, gives

$$|\eta_g(\tilde{\pi}) - \eta_f(\tilde{\pi})| \leq \frac{2r_{\max}}{1-\gamma}\min\left\{1, \frac{\mathcal{R}_{\tilde{\pi}}}{1-\gamma}\right\}.$$

The same uniform bound holds for every pair of renderings, even though their induced occupancies may differ substantially. No comparison of a training occupancy with an evaluation occupancy is required: each pointwise TV distance was bounded before taking its expectation under the new occupancy.

To pass from a rendering pair to the train–evaluation gap, write

$$\eta_{\text{eval}}(\tilde{\pi}) = \mathbb{E}_{g\sim p_{\text{eval}}(\cdot)}[\eta_g(\tilde{\pi})], \;\; \eta(\tilde{\pi}) = \mathbb{E}_{f\sim p_{\text{train}}(\cdot)}[\eta_f(\tilde{\pi})].$$

Assumption 3.1 gives the exact mixture identity

$$p = Zp_{\text{train}} + (1-Z)p_{\text{eval}}, \;\; \zeta(\tilde{\pi}) = Z\eta(\tilde{\pi}) + (1-Z)\eta_{\text{eval}}(\tilde{\pi}).$$

Averaging the pairwise return difference over independent $f \sim p_{\text{train}}$ and $g \sim p_{\text{eval}}$ therefore yields

$$\begin{aligned}
|\zeta(\tilde{\pi}) - \eta(\tilde{\pi})| &= (1-Z)|\eta_{\text{eval}}(\tilde{\pi}) - \eta(\tilde{\pi})| \\
&\le (1-Z)\mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot)\\ g\sim p_{\text{eval}}(\cdot)}}[|\eta_g(\tilde{\pi}) - \eta_f(\tilde{\pi})|] \\
&\le \frac{2r_{\max}(1-Z)}{1-\gamma}\min\left\{1, \frac{\mathcal{R}_{\tilde{\pi}}}{1-\gamma}\right\}.
\end{aligned}$$

The factor $1 - Z$ comes from this exact decomposition, not from a bound on the distance between rendering distributions. The only rendering comparison used in its derivation is the candidate policy's own robustness radius.

Now restore the learner index and use Theorem 3.3 for the actual pair $(\pi_i, \tilde{\pi}_i)$. Its training discrepancy is

$$\mathfrak{D}^{(i)}_{\text{train}} = \mathbb{E}_{\substack{f\sim p_{\text{train}}(\cdot)\\ s\sim d^{\mu_{i,f}}(\cdot)}}\left[D_{\text{TV}}(\tilde{\mu}_{i,f}, \mu_{i,f})[s]\right],$$

and $\epsilon^{(i)}_{\text{train}}$ is the corresponding maximal absolute expected reference advantage. The preceding gap bound gives

$$\begin{aligned}
\zeta(\tilde{\pi}_i) &\ge \eta(\tilde{\pi}_i) - C\min\left\{1, \frac{\mathcal{R}_{\tilde{\pi}_i}}{1-\gamma}\right\} \\
&\ge L_{\pi_i}(\tilde{\pi}_i) - \frac{2\gamma\epsilon^{(i)}_{\text{train}}}{(1-\gamma)^2}\mathfrak{D}^{(i)}_{\text{train}} - C\min\left\{1, \frac{\mathcal{R}_{\tilde{\pi}_i}}{1-\gamma}\right\} \\
&\ge L_{\pi_i}(\tilde{\pi}_i) - \frac{2\gamma\epsilon^{(i)}_{\text{train}}}{(1-\gamma)^2}\mathfrak{D}^{(i)}_{\text{train}} - C\min\left\{1, \frac{B_i}{1-\gamma}\right\}.
\end{aligned}$$

The last inequality uses $\mathcal{R}_{\tilde{\pi}_i} \le B_i$ from Theorem 3.15, monotonicity of the capped function, and $C \ge 0$. This proves the first assertion with exactly the constants stated in the corollary.

For the comparison with Corollary 3.10, evaluate all coefficients at this same policy pair. The explicit formula in Appendix C gives

$$\begin{aligned}
C_{\text{train}} - \frac{2\gamma\epsilon^{(i)}_{\text{train}}}{(1-\gamma)^2} &= \frac{2\delta_{\text{train}}(1-Z)}{1-\gamma}\left(1 + \frac{2\gamma\sigma_{\text{train}}}{1-\gamma}\right) \\
&\quad + \frac{2\delta_{\text{eval}}(1-Z)}{1-\gamma}\left(1 + \frac{2\gamma\sigma_{\text{eval}}}{1-\gamma}\right) \ge 0.
\end{aligned}$$

Here the δ and σ quantities are precisely those of the original bound for this pair and are nonnegative. Its coefficients C_π and $C_{\tilde{\pi}}$ are also nonnegative, as are both radii, B_i, and $\mathfrak{D}^{(i)}_{\text{train}}$. Since the capped factor is at most one, the two lower-bound expressions satisfy

$$\begin{aligned}
&L_{\pi_i}(\tilde{\pi}_i) - \frac{2\gamma\epsilon^{(i)}_{\text{train}}}{(1-\gamma)^2}\mathfrak{D}^{(i)}_{\text{train}} - C\min\left\{1, \frac{B_i}{1-\gamma}\right\} \\
&\ge L_{\pi_i}(\tilde{\pi}_i) - \frac{2\gamma\epsilon^{(i)}_{\text{train}}}{(1-\gamma)^2}\mathfrak{D}^{(i)}_{\text{train}} - C \\
&\ge L_{\pi_i}(\tilde{\pi}_i) - C_{\text{train}}\mathfrak{D}^{(i)}_{\text{train}} - C_\pi\mathcal{R}_{\pi_i} - C_{\tilde{\pi}}B_i - C.
\end{aligned}$$

The last line is exactly the original corollary after substituting B_i for the candidate radius. Thus the comparison is pointwise at a fixed pair, not a comparison that silently changes coefficients or policies. It does not assert dominance over every possible use of the original bound with a different, tighter radius estimate.

The cap also describes the scope of the robustness refinement. Its natural threshold is $\mathcal{R}_{\mathrm{crit}} = 1-\gamma$: if $B_i \geq 1-\gamma$, the reward-range penalty is still C, whereas if $B_i < 1-\gamma$ it is $CB_i/(1-\gamma)$. When γ is close to one, the threshold is small, so a reduction in the radius bound can remain entirely in the capped regime. The overall lower bound can also be loose because of its training discrepancy or its residual-based radius estimate. Neither a smaller radius nor a higher lower bound alone proves an increase of the actual return.

For a precise comparison of two certificates, suppose a fixed reference policy π is used for candidates $\tilde{\pi}^{(0)}$ and $\tilde{\pi}^{(1)}$, and the same valid $\epsilon_{\mathrm{train}}$ is used to bound the expected advantages of both pairs. Keep C and γ fixed, and let B_0, B_1 be their respective valid radius bounds; here B_0 denotes a radius bound, not the reward-range constant C. With $\Delta L = L_\pi(\tilde{\pi}^{(1)}) - L_\pi(\tilde{\pi}^{(0)})$ and $\Delta\mathfrak{D}_{\mathrm{train}} = \mathfrak{D}^{(1)}_{\mathrm{train}} - \mathfrak{D}^{(0)}_{\mathrm{train}}$, subtracting the two right-hand sides gives exactly

$$\Delta L - \frac{2\gamma\epsilon_{\mathrm{train}}}{(1-\gamma)^2}\Delta\mathfrak{D}_{\mathrm{train}} + C\left[\min\left\{1, \frac{B_0}{1-\gamma}\right\} - \min\left\{1, \frac{B_1}{1-\gamma}\right\}\right].$$

If $B_1 \leq B_0$, the last term is nonnegative but may vanish under the cap; the surrogate and discrepancy changes must still be included. If different advantage coefficients are used, one must instead subtract the full coefficient–discrepancy products. A positive difference in these expressions orders only the certified lower bounds, not the two unknown returns. This preserves the distinction between the proved refinement and an unsupported claim of automatic performance improvement. □