arXiv:2603.01365v1 [cs.LG] 2 Mar 2026

对齐与过滤改进异步同策略强化学习的性能

Homayoun Honari [1] **Roger Creus Castanyer** [1 2] **Michael Przystupa** [1 2 3] **Michael Noukhovitch** [1 2] **Pablo Samuel Castro** [1 2] **Glen Berseth** [1 2]

摘要

分布式训练和提高梯度更新频率是加速学习、提升性能的实用策略，但两者都会加剧一个核心挑战：策略滞后，即生成数据的行为策略与正在更新的学习策略之间存在不匹配。策略滞后可能会阻碍同策略学习算法扩展到更大规模的问题。本文指出了由分布式学习和高更新频率导致策略滞后的来源。基于这些发现，我们提出基于总变差的优势对齐约束策略优化（*VACO*），作为缓解策略滞后的实用方法。我们通过实证验证了该方法，并表明它在经典强化学习任务和现代大语言模型数学推理强化学习任务中，对策略滞后具有更强的鲁棒性。

1. 引言

分布式强化学习（RL）通过在去中心化计算系统中同时更新策略和收集数据，提供高度优化且快速的训练流程（Langer等人，2020）。在这种设置中，多个计算节点分别生成数据，然后将数据发送到学习节点进行处理。例如，多个机器人可以在本地运行策略来执行操作或运动任务，并将收集到的经验传输到中央计算节点，以改进策略（Gu等人，2017；Rudin等人，2022）。这种设置中最常见的难题之一，是通信反复失败和延迟，导致数据分布与学习策略的分布不同（Yuan与Mahmood，2022）。这种数据分布不匹配违反了同策略深度强化学习算法的一项关键假设，即行为策略与目标策略相同（Sutton等人，1998年）。同策略方法通过使用当前策略进行环境轨迹采样来生成数据，并进行保守更新，以使其接近数据分布（Schulman 等人，2015年）。例如，近端策略优化（PPO）及其变体使用KL散度来量化分布之间的接近程度（Schulman 等人，2017；Wang 等人，2020）。因此，策略优化的目标是在增加奖励的同时，限制其与行为策略之间的偏差。这使策略能够在数据上进行更多小批量更新，从而提升性能，但也扩大了数据分布与最新策略之间的差距。这种策略差距可能导致性能严重下降，甚至造成策略彻底崩溃（Nikishin等人，2022）。文献中先前将这一局限称为策略滞后，并指出它会阻碍策略改进（Babaeizadeh等人，2016 年；Petrenko等人，2020）。

本文对策略滞后的来源进行了分类：**反向策略滞后**，它源于行为策略与学习策略之间最初的不匹配；以及**前向策略滞后**，它会随着梯度更新导致学习策略偏离数据分布而不断累积。这种前向滞后带来了一个关键的权衡：每个数据批次执行更多更新或许能提升性能，但也会增加策略退化的风险。我们证明，总变差（TV）散度是量化这两种策略滞后的有效工具。

我们提出基于总变差的优势对齐约束策略优化（VACO），以解决异步强化学习中固有存在的前向和反向策略滞后。VACO 基于两个主要思想：**优势重对齐**和**基于总变差散度的过滤**。我们利用优势重对齐，从离策略数据中估计学习策略的优势函数，从而缓解反向策略滞后问题。为控制策略优化过程中可能出现的前向策略滞后，我们随后使用总变差散度，过滤每个小批量中会放大前向策略滞后影响的数据点。我们发现，这种过滤策略能够有效地将期望总变差散度的值维持在特定阈值以内，同时无需为满足约束而额外增加超参数选择的复杂性。

[1]Mila– Quebec AI Institute，加拿大 [2]Universit´e de Montr´eal，加拿大 [3]阿尔伯塔大学，加拿大。通讯作者：Homayoun Honari <homayoun.honari@mila.quebec>。

预印本。*2026*年*3*月*3*日。

为研究反向策略滞后对各种同策略强化学习算法的影响，我们提出了一个用于强化学习的模拟异步框架（Howes 等人，2025年）。我们使用该框架，在各种 MuJoCo（Todorov 等人，2012年）机器人任务中模拟高度并行化环境。我们发现，与最常用的同策略算法之一近端策略优化（Schulman 等人，2017）相比，VACO 对反向策略滞后表现出更强的鲁棒性。随后，我们研究了利用强化学习对大语言模型进行数学推理微调时存在的前向策略滞后。我们将 VACO 应用于 GRPO（Shao 等人，2024年），这是大语言模型强化学习中的标准方法；结果发现，对于这些设置中常见的前向滞后，VACO 显著提升了鲁棒性。

总而言之，我们首先对异步强化学习中的策略滞后进行理论分析，并将其不同来源分类。在此分析的基础上，我们提出了 VACO，这是一种旨在提升对前向和后向滞后鲁棒性的策略优化算法。随后，我们在两个精心构建的设置中对该方法进行了实证验证：高度并行化的机器人任务，以及大语言模型的推理。在这两种场景中，与强有力的标准强化学习基线方法相比，我们都显著提升了对策略滞后的鲁棒性。

2. 相关工作

我们的工作探讨了同策略强化学习算法在学习器智能体异步接收数据时的学习动态问题。同策略方法的一个关键假设是，数据由当前策略采集，这使得重复利用数据或从异策略样本中学习颇具挑战性（Sutton 等人，1998年）。缓解这一问题的一种标准机制是定义信赖域约束，以保守的方式更新策略，从而允许对给定的数据批次进行多次更新（Schulman等人，2015a）。此前已有研究探讨了其他近似方法，以减少优化策略时对二阶优化器的需求（Schulman等人，2015a；2017；Wang 等人，2020；Xie 等人，2025年；Achiam 等，2017b；Vuong 等人，2019年）。研究者还提出了异策略变体，在目标函数中引入自适应机制，以调整信赖域正则化项的重要性（Meng 等人，2023；Fakoor 等人，2020；Queeney 等人，2021年）。我们的研究建立在此前提出的全变差散度度量之上，该度量用于分析同策略学习中数据分布的变化（Schulman等人, 2015a；Achiam 等，2017a）。

尽管近期取得了进展，同策略方法通常仍需要大量数据批次才能实现稳定优化（Andrychow- icz 等人；Singla 等人，2024年）。分布式学习架构通过并行化数据收集和策略更新来解决这一问题（Liu 等人，2024年；Czech，2021年；Langer 等人，2020）。在这些设置中，中心策略要么将其权重发送给并行执行器以收集数据（Nair 等人，2015；Babaeizadeh等人，2016 年；Mnih等人，2016 年），要么由执行器更新局部策略并将梯度发送回中心学习器（Kyzy等人，2025年）。异步处理方法的一个权衡是会引入执行策略与正在更新的策略之间的不匹配，这通常称为策略滞后（Espeholt 等人，2018年；Babaeizadeh等人，2016 年）。我们的方法通过有选择地应用仅与优势函数方向一致的更新，来缓解策略滞后。

此外，在需要同时进行推理和学习的现实场景中，同步策略更新往往并不实用。这一挑战普遍存在于多个领域，包括实时机器人、设备端移动应用以及大规模语言模型的训练；在这些场景中，系统的复杂性使异步学习成为必要（Yuan与Mahmood，2022；Gu等人，2017；Radac与Chirla，2025年）。我们的算法通过有效利用异步样本来满足这一需求，使其成为应对这些要求严苛的真实应用场景的实用解决方案。

3. 背景

强化学习（RL）问题通常被形式化为马尔可夫决策过程（MDP）（Puterman，2014）。MDP是一个六元组 $(\mathcal{S}, \mathcal{A}, \mathcal{R}, \mathcal{P}, \mu, \gamma)$，由状态空间 $\mathcal{S}$、动作空间 $\mathcal{A}$、奖励函数 $\mathcal{R}(\cdot)$、状态转移概率函数 $\mathcal{P}(\cdot)$、初始状态分布 $\mu(\cdot)$和折扣因子 $\gamma \in [0, 1)$ 构成。强化学习问题的目标是优化策略 π：$\mathcal{S} \times \mathcal{A} \to [0, 1]$，使生成的轨迹的期望折扣回报最大化：

$$J(\pi) = \mathbb{E}_{s_0 \sim \mu, a_t \sim \pi(s_t), s_{t+1} \sim \mathcal{P}(s_t, a_t)} \left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) \right].$$

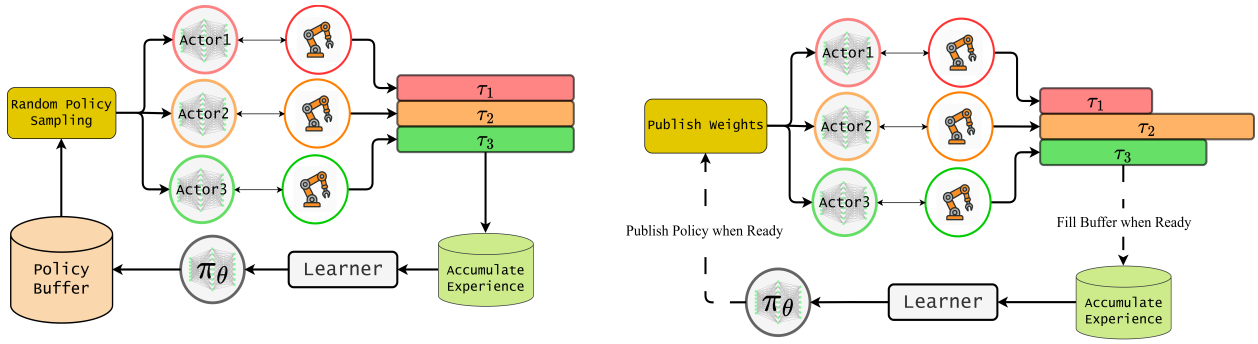
此外，状态-动作价值函数定义为从指定的状态-动作对出发所获得的期望折扣回报，

$$Q_\pi(s, a) = \mathbb{E}_{\substack{a_t \sim \pi(s_t) \\ s_{t+1} \sim \mathcal{P}(s_t, a_t)}} \left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) | s_0 = s, a_0 = a \right],$$

并可用于推导状态价值函数 $V\pi(s) = \mathbb{E}_{a \sim \pi}[Q_\pi(s, a)]$ 最后，优势函数定义为 $A_\pi(s, a) = Q_\pi(s, a) - V_\pi(s)$。从策略 π 采样时，$A_\pi(s, a)$ 的期望值为零：$\mathbb{E}_{a \sim \pi(s)}[A_\pi(s, a)] = \mathbb{E}_{a \sim \pi(s)}[A_\pi(s, a)] = \mathbb{E}_{a \sim \pi(s)}[Q_\pi(s, a)] - V_\pi(s) = 0$。

3.1. 异步同策略强化学习

在典型的异步强化学习训练设置中，会使用多个相互独立的智能体来收集轨迹。在这种设置下，智能体会在策略更新完成后，使用最新版本的策略



Mixture of Policies for Simulated Asynchronous RL　　Real-World Asynchronous RL

图 *1.* （**左**）模拟的异步强化学习设置。在每个训练阶段结束后，我们会将新策略的权重存储到策略缓冲区中。随后，我们从缓冲区中随机抽取策略并分配给执行器。接着，我们以同步方式生成轨迹。该设置旨在模拟异步强化学习训练设置，以控制反向策略滞后。（**右**）典型的异步强化学习训练设置。执行器生成轨迹的同时，策略也在接受训练。新策略一旦准备就绪，执行器便会接收该策略。因此，新数据集会包含来自较旧策略的轨迹。

。每次典型的策略训练迭代都包含两个阶段：轨迹生成和策略优化。轨迹生成阶段会持续收集数据，直到数据缓冲区填满。接下来，策略优化阶段会基于这些数据对策略进行多个轮次的更新。为此，考虑迭代 T，轨迹是使用行为策略 β_T 生成的。在异步设置下，该策略可表示为一个分段混合分布（Wiering& VanHasselt，2008）：

$$\begin{aligned}\beta_T(a|s) &= \mathbb{E}_{i\sim\mathcal{M}(0,\dots,T)}\left[\pi_i(a|s)\right]\\ \Pr\left(s|\tau\sim\beta_T(\mu)\right) &= \mathbb{E}_{s_0\sim\mu,i\sim\mathcal{M}(0,\dots,T)}[\Pr(s|\tau\sim\pi_i)]\end{aligned} \tag{1}$$

其中，混合分布 $\mathcal{M}$ 通常先验未知，并取决于具体的训练设置。由于异步设置的特性，可以观察到，即使策略由概率密度函数（例如高斯分布）参数化，混合分布也可能采用不同的参数化方式。在本文接下来的理论分析中，β_T 将用于表示迭代 T时的行为。此外，下一次迭代 $T+1$ 的策略，是通过对迭代 T的策略进行多轮训练得到的：

$$\theta_{T+1}^{(n)} \leftarrow \theta_{T+1}^{(n-1)} + \eta\nabla_\theta L(\theta_{T+1}^{(n-1)}) \tag{2}$$

其中 $\theta_{T+1}^{(n)}$ 表示第 n 轮的策略参数，$\theta_{T+1}^{(0)} = \theta_T$，$\eta$表示学习率，$L$ 表示用于优化策略的损失函数。异步强化学习的设置如图1所示。

3.2. 信赖域策略优化

信赖域策略优化（TRPO）（Schulman 等人，2015年）是最早成功开发出具有单调策略改进保证的同策略强化学习方法之一。TRPO 的策略改进保证基于性能差异引理：

引理 3.1. *(Kakade 与 Langford，2002)* 对任意两个策略 π 和 π'，以及初始状态分布 μ：

$$J(\pi') - J(\pi) = \mathbb{E}_{\tau\sim\pi'|s_0\sim\mu}\left[\sum_{t=0}^{\infty}\gamma^t A_\pi(s_t,a_t)\right] \tag{3}$$

$$= \frac{1}{1-\gamma}\mathbb{E}_{s\sim d^{\pi'},a\sim\pi'(s)}\left[A_\pi(s,a)\right] \tag{4}$$

其中 $d^\pi(s) = (1-\gamma)\sum_{t=0}^{\infty}\gamma^t\mathcal{P}(s_t=s|\mu,\ \pi)$ 表示折扣后的未来状态分布。关于性能差分引理的证明，请读者参见第 B.1 节，或参见（Kakade 与 Langford，2002；Schulman 等人，2015年；Agarwal 等人，2019年）。性能差异方程可以利用优势函数有效地关联两个策略。

然而，由于 $d^{\pi'}$ 依赖于 π'，使用该方程优化策略可能颇具挑战。为此，（Kakade 与 Langford，2002；Schulman 等人，2015年）提出了性能差异的近似版本：

$$L_\pi(\pi') = \frac{1}{1-\gamma}\mathbb{E}_{\substack{s\sim d^\pi\\ a\sim\pi(s)}}\left[\frac{\pi'(a|s)}{\pi(a|s)}A_\pi(s,a)\right]. \tag{5}$$

近似版本允许以保守步长优化策略的性能，因为它在一阶上与原始版本相匹配。随后可以使用方程 5[1]来刻画两个策略之间的性能差异：

定理 3.2。 *(Achiam 等人，2017a)* 对任意两个策略π' 和 π，定义 $\epsilon^{\pi'} = \max_{s\in\mathcal{S}}|\mathbb{E}_{a\sim\pi'}[A_\pi(s,\ a)]|$，

[1](Schulman 等人，2015年) 也提出了该定理的类似变体。不过，(Achiam 等人，2017a) 中讨论的版本使用了 D_{TV}的期望值，实际应用起来更为方便。

记作 $D_\pi^{\pm}(\pi') = {}^{L_\pi}(\frac{\pi'}{1-\gamma} \pm \frac{2\gamma\epsilon^{\pi'}()^2}{1-\gamma}\mathbb{E}_{s\sim d^\pi}[D_{TV}(\pi'\|\pi)[s]]$ 其中 $D_{TV}(\pi'\|\pi)[s] = (1/2)\sum_{a\in\mathcal{A}}|\pi'(a|s) - \pi(a|s)|$，性能差异可以界定为：

$$D_\pi^-(\pi') \le J(\pi') - J(\pi) \le D_\pi^+(\pi') \tag{6}$$

公式中的界6 是紧的，因为在 $\pi' = \pi$ 处我们得到$D_\pi^{\pm}(\pi') = 0$。因此，通过对下界的迭代优化 $D_\pi^-(\pi')$，该策略在性能上享有单调改进。

虽然公式6中性能差异的下界适合用于优化策略，但对下界进行精确优化会很困难，因为$\epsilon^{\pi'} = \max_{s\in\mathcal{S}}|\mathbb{E}_{a\sim\pi'}[A_\pi(s,a)]|$ 难以评估。信赖域策略优化的开发者将下界方程推导为 ${}^{L_{\pi(\pi')\,1-\gamma}} - {}^{4\gamma\epsilon}_{1-\gamma\,()^2}\max_{s\in\mathcal{S}} D_{KL}(\pi,\pi')$，并提出约束平均KL散度 $D_{KL}(\pi,\pi') = \mathbb{E}_{s\sim d^\pi}[D_{KL}(\pi(.|s)\ \pi'(.|s)] \le \delta$ 作为max算子的启发式近似，

$$\theta^* = \underset{\theta\in\Theta}{\text{argmax}}\ \mathbb{E}_{\substack{s\sim d^{\pi_{\theta_{old}}}\\ a\sim\pi_{\theta_{old}}(s)}}\left[\frac{\pi_\theta(a|s)}{\pi_{\theta_{old}}(a|s)}A_{\pi_{\theta_{old}}}(s,a)\right] \quad s.t.\ \bar{D}_{KL}(\pi_{\theta_{old}},\pi_\theta) \le \delta, \tag{7}$$

其中，$\pi_{\theta_{old}}$ 是用于收集轨迹的初始策略。因此，只要阈值 δ足够保守，策略就能实现回报不减。

3.3. 近端策略优化

信赖域策略优化在公式 7 中的相同思路也可以应用于公式 6 (Achiam 等人，2017a)，将约束替换为总变差距离$\mathbb{E}_{s\sim d^{\pi_{\theta_{old}}}}[D_{TV}(\pi_{\theta_{old}}\|\pi_\theta)[s]] \le \delta$。

$\mathbb{E}_{s\sim d^{\pi_{\theta_{old}}}}[D_{TV}(\pi_{\theta_{old}}\|\pi_\theta)[s]]$ 还可以通过从 $\pi_{\theta_{old}}$的滚动轨迹中采样来近似表示为：

$$\mathbb{E}_{s\sim d^{\pi_{\theta_{old}}}}[D_{TV}(\pi_{\theta_{old}}\|\pi_\theta)[s]] = \frac{1}{2}\mathbb{E}_{\substack{s\sim d^{\pi_{\theta_{old}}}\\ a\sim\pi_{\theta_{old}}(s)}}\left[\left|\frac{\pi_\theta(a|s)}{\pi_{\theta_{old}}(a|s)} - 1\right|\right] \tag{8}$$

为使用全变差散度优化策略，近端策略优化（PPO）提出了裁剪代理目标损失：

$$L_{CLIP}(\theta) = \mathbb{E}_{\substack{s\sim d^{\pi_{\theta_{old}}}\\ a\sim\pi_{\theta_{old}}(s)}}[clip(r_\theta(s,a), 1-\delta, 1+\delta)A_{\pi_{\theta_{old}}}(s,a)] \tag{9}$$

其中 $r_\theta(s,\ a) = \frac{\pi_\theta(a|s)}{\pi_{\theta_{old}}(a|s)}$。裁剪技术会移除小批量中比率超出 $(1-\delta,\ 1+\delta)$ 范围的所有梯度。该裁剪机制旨在通过计算高效的操作，对 D_{TV} 进行界定，并已在其他近端策略优化变体中得到应用（Queeney 等人，2021年）。

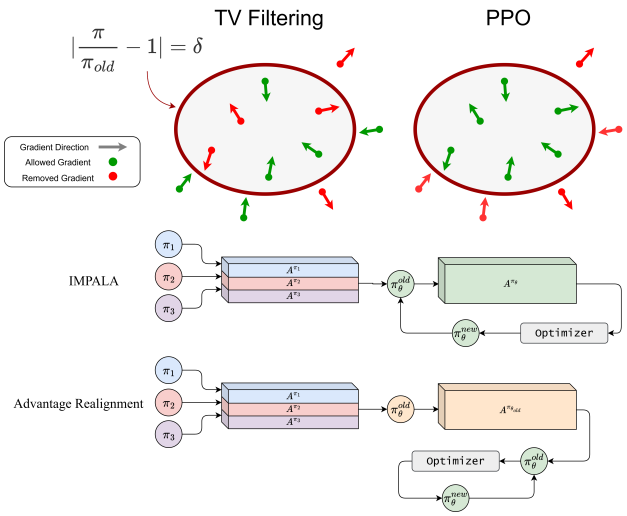


图 2. **VACO 中的算法选择。(上) VACO 与 PPO 裁剪**：为了维持固定的总变差（TV）散度，VACO（显示为“总变差过滤”）会有选择地移除会增大 TV 散度的梯度。相比之下，如果策略比率超过预定义阈值，PPO 会直接裁剪梯度。**(下) IMPALA 与优势重对齐**：IMPALA 使用异步生成的数据集估算最新策略的优势值，而 VACO 的“优势重对齐”会先将数据集与优化过程的初始策略对齐，再基于这个对齐后的数据集进行迭代优化。与 IMPALA 实时进行重对齐相比，这种方法显著降低了计算负荷。

4. 方法

4.1. 使用离策略数据的近端策略优化性能

为了理解为什么公式 5 中的性能差异近似版本会导致近端策略优化的性能下降，我们首先给出针对离策略分布的性能差分引理推广（下文使用一般记号 π 表示训练过程中可能获得的任何后续策略）：

引理 4.1. 考虑迭代 T 时的策略为 π_T，行为策略为 β_T。然后，记 $\epsilon^\pi = \max_{s\in\mathcal{S}}|\mathbb{E}_{a\sim\pi}[A_{\beta_T}(s,a)]|$，任意策略 π 与 π_T 之间的性能差异可利用 β_T 的状态分布给出一个下界：

$$J(\pi) - J(\pi_T) \ge \frac{1}{1-\gamma}\mathbb{E}_{\substack{s\sim d^{\beta_T}\\ a\sim\beta_T(s)}}\left[\frac{\pi(a|s)}{\beta_T(a|s)}A_{\beta_T}(s,a) - \frac{\pi_T(a|s)}{\beta_T(a|s)}A_{\beta_T}(s,a) - \left(\frac{2\gamma\epsilon^\pi}{1-\gamma}D_{TV}(\beta_T\|\pi)[s] + \frac{2\gamma\epsilon^{\pi_T}}{1-\gamma}D_{TV}(\beta_T\|\pi_T)[s]\right)\right] \tag{10}$$

利用方程10中的下界，可以识别出策略滞后的来源：

前向策略滞后 随着对策略进行更多梯度更新，如果不加以控制，$\frac{2\gamma\epsilon^{\pi}}{1-\gamma} D_{TV}(\beta_T\|\pi)[s]$ 下界中的惩罚项会变得更大。因此，惩罚项超过优势项的风险$\pi(a|s)\,\beta_T(a|s)\,A_{\beta_T}(s,\ a)$ 也会增加。事实上，近端策略优化中的裁剪机制或信赖域策略优化中的KL约束等各种同策略技巧，都是为了应对这一问题。

反向策略滞后 初始学习策略之间的不匹配 π_T 与行为策略 β_T 会对性能差异造成不可避免的惩罚$\left(\frac{\pi_T(a|s)\,(}{\beta_T}\right.$ $_{a|s)}\,A_{\beta_T}(s,a)\,+\,\frac{2\gamma\epsilon^{\pi_T}}{1-\gamma}\,D_{TV}(\beta_T\|\pi_T)[s]\Big)$。因此，任何初始分布不匹配都会导致性能下降。

为观察这一点，考虑当 $\pi=\pi_T$时的下界：

$$\begin{aligned} &J(\pi_T)-J(\pi_T)=0\\ &\geq \frac{1}{1-\gamma}\mathbb{E}_{\substack{s\sim d^{\beta_T}\\ a\sim\beta_T(s)}}\left[-\frac{4\gamma\epsilon^{\pi_T}}{1-\gamma}D_{TV}(\beta_T\|\pi_T)[s]\right] \end{aligned} \tag{11}$$

分布之间的任何不匹配 β_T 与 π_T 都会产生严格为负的下界。因此，要优化策略以使该项$\pi(a|s)\,\beta_T(a|s)\,A_{\beta_T}(s,a)$ 优于其他三个惩罚项并产生正的下界，可能会很困难。

4.2. 离策略性能差异

在引理4.1 中，我们证明了性能差异近似版本会带来不可避免的惩罚。不过，下界可以得到改进。

引理 4.2。考虑迭代 T 时的策略 π_T和行为策略 β_T。记 ϵ^{π} $=\max_{s\in\mathcal{S}}|\mathbb{E}_{a\sim\pi}[A_{\pi_T}(s,a)]|$，则任意策略 π 与 π_T 之间的性能差异，可以利用 β_T 的状态分布给出下界：

$$\begin{aligned} &J(\pi)-J(\pi_T)\geq\\ &\frac{1}{1-\gamma}\mathbb{E}_{\substack{s\sim d^{\beta_T}\\ a\sim\beta_T(s)}}\left[\frac{\pi(a|s)}{\beta_T(a|s)}A_{\pi_T}(s,a)-\frac{2\gamma\epsilon^{\pi}}{1-\gamma}D_{TV}(\beta_T\|\pi)[s]\right] \end{aligned} \tag{12}$$

公式 6 与公式 12 之间的主要差异可归因于优势函数。这一差异产生了重要影响。为说明这一点（如第3节所述），注意在 π $=$ π_T我们得到 $\mathbb{E}_{s\sim d^{\beta_T},a\sim\beta_T(s)}[\frac{\pi_T}{\beta_T}(a|s)\,A_{\pi_T}(s,a)]$ $=\mathbb{E}_{s\sim d^{\beta_T},a\sim\pi_T(s)}[A_{\pi_T}(s,a)]$ $=$ 0 以及 ϵ^{π_T} $=\max_{s\in\mathcal{S}}|\mathbb{E}_{a\sim\pi_T}[A_{\pi_T}(s,a)]|$ $=$ 0。因此，尽管下界中仍然存在前向策略滞后，但离策略性能差异不存在反向策略滞后。此外，我们讨论了使用公式 12 优化策略的两项主要修改。

4.2.1. 优势重对齐

如前所述，公式 12 与公式6 之间的关键差异在于使用学习策略的优势函数，A_{π_T}，而不是行为策略的优势函数，A_{β_T}。由于轨迹由行为策略 β_T生成，我们只能直接估计 A_{β_T}。这带来了离策略评估的挑战：如何使用由一个策略生成的数据来估计另一个策略的价值（Munos 等人，2016 年；Espeholt 等人，2018年）。

一种突出的方法，尤其适用于异步设置，是 IMPALA 中的 V-trace 方法（Espeholt 等人，2018年）。给定由行为策略 β 生成的轨迹，策略 π 在任意状态 s下的 V-trace目标定义如下：

$$v_\pi(s):=V_\beta(s)+\sum_{t=0}^{\infty}\gamma^t\left(\prod_{i=0}^{t-1}c_i\right)\rho_t\left(r_t+\gamma V_\beta(s_{t+1})-V_\beta(s_t)\right) \tag{13}$$

其中 $c_i=\min\left(\bar{c},\frac{\pi}{\beta}(a_t|s_t)\,(a_t|s_t)\right)$，$\rho_t=\min\left(\bar{\rho},\frac{\pi}{\beta}(a_t|s_t)\right.$ $\left.(a_t|s_t)\right)$。关于离策略性能差异，我们使用公式 13 来估计 π_T 的价值函数，使用从 β_T中采样的轨迹。此外，与近端策略优化中使用的广义优势估计（GAE）（Schulman 等人，2015b）方法类似，V-trace目标也可以在一条轨迹中通过递归高效计算。递归规则和优势函数定义如下：

$$\begin{aligned} v_\pi(s_t)=V_\beta(s_t)+\rho_t(r_t+\gamma V_\beta(s_{t+1})-V_\beta(s_t))\\ +\gamma\lambda c_t(v_{t+1}-V(s_{t+1})) \end{aligned} \tag{14}$$

$$A_{vtrace}(s_t,a_t)=r_t+\gamma v_\pi(s_{t+1})-V_\beta(s_t) \tag{15}$$

其中，$\lambda\in[0,1]$ 是类似于 GAE 的折扣参数，用于控制估计值的偏差-方差权衡。我们注意到，减去 $V_\beta(s_t)$是为了实现方差缩减。

如 Espeholt 等人，2018年 中所述，$\bar{\rho}$ 控制 V-trace 收敛到的策略不动点。因此，对于 $\bar{\rho}=\infty$，目标会得到 V_π。另一方面，$\bar{c}$通过压缩系数 η控制目标的收敛速度。

IMPALA 与我们的优势重对齐方法之间的关键区别在于优势估计的频率。IMPALA 会在每个步骤中持续重新估计当前策略的优势函数，实际上是将问题视为一系列同策略更新。相比之下，优势重对齐仅针对初始学习策略计算一次优势函数，π_T，然后迭代优化公式 6。这种避免重复计算的做法，使优势重对齐的计算效率显著高于 IMPALA。除了架构差异之外，与 IMPALA 相比，公式12还使优化过程能够更好地抵御估计中的离策略校正误差，因为目标

优势函数是固定的，而 IMPALA 中的目标则具有可变性。

4.2.2. 基于过滤的约束策略优化

除了准确估计公式 12 中的优势之外，最小化惩罚项也至关重要。然而，如第3节所述，直接控制max 算子（位于误差项 ϵ^{π} 中）颇具挑战。因此，受先前约束策略优化工作（Achiam 等人，2017a）的启发，我们重新表述该问题。我们在总变差（TV）散度约束下最小化目标，从而减轻 ϵ^{π}[2]的影响：

$$\begin{aligned}\theta^* = \text{argmax}_{\theta\in\Theta}\, \mathbb{E}_{\substack{s\sim d^{\beta_T}\\ a\sim\beta_T(s)}}\Big[\frac{\pi(a|s)}{\beta_T(a|s)}A_{\pi_T}(s,a) \\ - \frac{2\gamma\epsilon^{\pi}}{1-\gamma}D_{TV}(\beta_T\|\pi)[s]\Big] \\ \Downarrow \\ \theta^* = \text{argmax}_{\theta\in\Theta}\mathbb{E}_{s\sim d^{\beta_T}, a\sim\pi_\theta(s)}[A_{\pi_T}(s,a)] \\ s.t.\ \mathbb{E}_{s\sim d^{\beta_T}}[D_{TV}(\beta_T\|\pi_\theta)[s]] \le \delta/2\end{aligned} \tag{16}$$

约束优化的一种常见方法是使用拉格朗日方法将问题转化为无约束问题，该方法能够渐近地保证约束满足（Altman，2021年）。然而在实践中，由于乘子的初始选择以及每次更新的数据有限，在每次迭代都满足约束颇具挑战。为了在每一步更好地满足约束，我们的方法直接分析损失梯度与约束梯度之间的关系。设 θ 为策略参数；我们考虑损失函数的梯度以及 D_{TV} 关于 θ 的梯度：

$$\begin{aligned}&\nabla_\theta \mathbb{E}_{\substack{s\sim d^{\beta_T}\\ a\sim\beta_T(s)}}\left[\frac{\pi_\theta(a|s)}{\beta_T(a|s)}A_{\pi_T}(s,a)\right] \\ &= \mathbb{E}_{\substack{s\sim d^{\beta_T}\\ a\sim\beta_T(s)}}\left[\frac{\pi_\theta(a|s)}{\beta_T(a|s)}\nabla\log\pi_\theta(a|s)A_{\pi_T}(s,a)\right]\end{aligned} \tag{17}$$

$$\begin{aligned}&\nabla_\theta \mathbb{E}_{\substack{s\sim d^{\beta_T}\\ a\sim\beta_T(s)}}\left[\left|\frac{\pi_\theta(a|s)}{\beta_T(a|s)}-1\right|\right] \\ &= \mathbb{E}_{\substack{s\sim d^{\beta_T}\\ a\sim\beta_T(s)}}\left[\frac{\pi_\theta(a|s)}{\beta_T(a|s)}\nabla\log\pi_\theta(a|s)sgn(\pi_\theta(a|s)-\beta_T(a|s))\right]\end{aligned} \tag{18}$$

从该比较中可以观察到，某个特定数据点是否会导致 D_{TV} 的增加或减少，可以由该比较来确定

[2]阈值指定为 $\delta/2$，以便与 $1/2$ 中用于定义 D_{TV} 的阈值保持一致，并使 δ 反映裁剪机制旨在处理的同一阈值。

介于 $A_{\pi_T}(s,a)$的符号与 $sign(\pi_\theta(a|s)-\beta_T(a|s))$的符号之间。利用这一性质，如果 $\mathbb{E}[D_{TV}]$ 在该小批量上超过阈值，我们建议分离该小批量中会导致全变差散度增加的数据点的梯度：

$$\begin{aligned}&if\ \mathbb{E}_{s\sim d^{\beta_T}}[D_{TV}(\pi_\theta\|\beta_T)[s]] > \delta \\ &\Rightarrow \text{remove } i : A_{\pi_T}(s_i,a_i)sgn(\pi_\theta(a_i|s_i)-\beta_T(a_i|s_i)) > 0\end{aligned} \tag{19}$$

我们提出的过滤方法可以修改损失函数，使其在满足约束的同时无需额外控制超参数。此外，这种修改也兼容最大熵目标：通过从 β_T 中采样动作，并使用重要性采样比：

$$\mathcal{H}(\pi_\theta) = -\mathbb{E}_{\substack{s\sim d^{\beta_T}\\ a\sim\beta_T(s)}}\left[\frac{\pi_\theta(a|s)}{\beta_T(a|s)}\log\pi_\theta(a|s)\right] \tag{20}$$

$$\begin{aligned}&\nabla_\theta L_{\beta_T}(\theta) \\ &= \mathbb{E}_{\substack{s\sim d^{\beta_T}\\ a\sim\beta_T(s)}}\left[\frac{\pi_\theta(a|s)}{\beta_T(a|s)}\nabla_\theta\log\pi_\theta(a|s)(A_{\pi_T}(s,a)-c_{\mathcal{H}}\log\pi_\theta(a|s))\right]\end{aligned} \tag{21}$$

其中，$c_{\mathcal{H}}$ 是最大熵系数。该过滤技术能够保守地提升策略的性能，同时最大化熵，而不会有违反全变差约束的风险。

直观而言，TV过滤方法的作用类似于控制器。当主策略与行为策略之间的全变差散度低于阈值时，该方法允许使用所有数据点。反之，当散度值高于阈值时，它只使用那些能够降低散度值的数据点。

总体而言，我们将优势重新对齐与基于 TV 的过滤相结合的理念称为 VACO。重新对齐与 IMPALA 的区别，以及基于 TV 的过滤与近端策略优化的区别，如图 2 所示；该算法的伪代码见算法1。

5. 实验

我们通过考察策略滞后对标准强化学习算法性能的影响，在实验中验证了我们的方法。随后，我们实现 VACO，以解决以下问题：1. 异步 MuJoCo 设置导致的反向策略滞后；2. 用于大语言模型推理的强化学习设置导致的前向策略滞后。

5.1. MuJoCo 中的反向策略滞后

如上一节所述，策略滞后的主要来源可归因于当前策略与行为策略之间的不匹配。这种不匹配最初

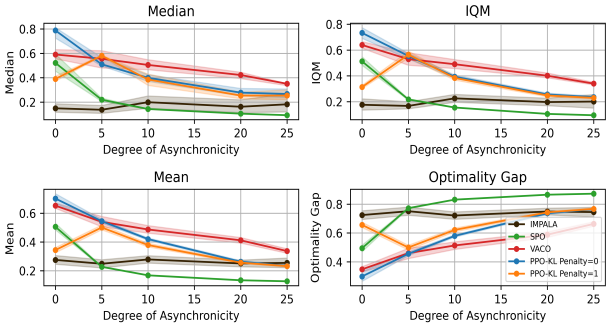


图 *3.* 随着反向策略滞后增加（异步程度提高），VACO 在多个 MuJoCo 任务的综合指标上取得了更好的表现。中位数、四分位均值和均值越高，最优性差距越低，表明性能越好。分数基于 10 个独立随机种子、每个种子运行 100M 步计算得出，阴影表示各项指标的 95%置信区间。

可表现为状态分布中行为策略的差异，并在连续执行损失函数的梯度下降过程中进一步放大。前者发生在异步／分布式采样设置中，后者则发生在每次梯度下降中。尽管过去已有各种异步强化学习设置被提出，但由于底层硬件层面的非确定性，获得可复现的结果仍是一项挑战（Huang 等人，2023）。为此，为了在受控环境中研究策略滞后，我们提出使用模拟异步设置（Howes 等人，2025年）。模拟平台与真实环境设置之间的区别如图1所示。如第 3.1 节所述，异步设置可描述为一种策略混合设置，其混合规则未知。因此，我们旨在通过可控设置复现这一过程。在模拟设置中，我们使用固定容量的策略缓冲区来保存过去的策略。训练阶段结束后，我们将缓冲区中策略的权重保存下来，再从缓冲区中随机采样并将其分配给执行器。因此，我们同步采样数据，以生成轨迹数量相等的数据集。该设置使我们能够通过策略混合设置研究行为策略与当前策略之间的不匹配。训练阶段结束后，我们在环境中运行最终策略多个回合，并通过计算各条轨迹回报的平均值来得出该策略的回报。

我们在 CleanRL 代码库中实现了 VACO（Huang 等人，2022），并以 PPO 为基础。根据 Pinsker 不等式，我们有 $D_{TV}(p\|q)^2 \leq D_{KL}(p\|q)$。因此，由于使用这一方法已成为 PPO 减少策略滞后的标准技巧（Schulman 等人，2017；Ziegler 等人，2019；Stiennon，2020），我们采用不同系数值的 KL 散度惩罚来运行 PPO，并报告回报表现最佳的运行结果。我们将“PPO-KL Penalty=0”称为 PPO 的标准版本（称为 PPO-Clip），将“PPO-KL Penalty=1”称为采用 KL 正则化且系数为 1 的版本。此外，我们使用 Simple Policy Optimization（SPO）（Xie 等人，2025年），该方法将全变差散度 $\mathbb{E}[(\frac{\pi}{\beta}-1)^2]$ 的平方用作惩罚项，并移除裁剪机制。此外，我们基于 CleanRL 基线实现 IM-PALA（Espeholt 等人，2018年），以评估这种持续重新估计新策略优势函数的方法。关于超参数的选择，我们采用 CleanRL 代码库提供的标准默认超参数，并将 $\delta = 0.2$ 用作 VACO 的约束界限。最后，为了评估策略滞后对算法性能的影响，我们使用支持高度并行化的 MuJoCo 环境。正如（Xie 等人，2025年）中所述，由于 MuJoCo 运动任务没有人工评分，我们使用所有算法获得的最高分和最低分，并将分数归一化为 $\frac{\text{回报-最低分}}{\text{最高分-最低分}}$。

此外，按照（Agarwal 等人，2021年）的提议，我们使用分层自助法置信区间，并评估算法的置信区间以获得各项指标。我们针对每种算法以及具有固定异步行为程度的运行（图 1中的策略缓冲区容量）计算综合指标。实验结果如图 3所示。尽管异步程度提高会影响所有对比算法，但可以观察到 VACO 对离策略数据具有更强的鲁棒性。

此外，为了研究 VACO 与其他算法相比的样本效率，图 4展示了训练过程中五种环境下各算法汇总后的 IQM 值。除了图 3所示的最终性能外，在整个训练阶段，VACO 对异步数据也表现出更好的鲁棒性。此外，归一化回报图中曲线下面积的 IQM 值表明，在不同程度的异步性下，VACO 通常都具有更高的样本效率。

5.2. RLVR 中的前向策略滞后

最后，我们将 VACO 应用于一种现代设置：通过 RL 微调大语言模型，以借助可验证奖励实现推理（RLVR）（Lambert 等人，2025）。一个主要挑战是，异步 RLVR（Noukhovitch 等人，2025）已被证明效率更高，但代价是策略滞后加剧，可能导致性能下降。遵循最先进的 RLVR 方法（Sheng 等人，2025），我们使用 LLM 推理引擎 vLLM（Kwon 等人，2023）通过同一策略生成 N 小批量数据 β，为每个答案标注奖励：正确为 1，错误为 0，然后使用训练库 transformers（Wolf 等人，2020）训练N 步。我们的策略 π 因此在最初的第一个小批量中是同策略的，但

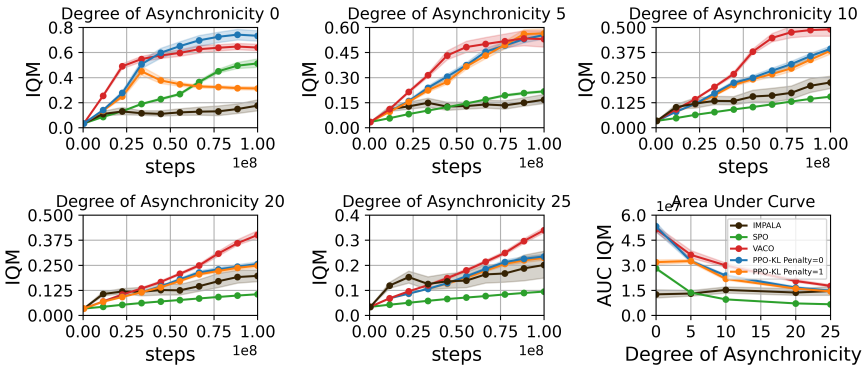


图 *4*。 在模拟异步设置的训练过程中，对比算法的四分位均值。分数根据 10 个独立随机种子下的 1 亿步数据计算，阴影区域表示 95%置信区间。（**右下**）归一化回报图曲线下面积的四分位均值。数值越高，表示训练过程中的样本效率越高。

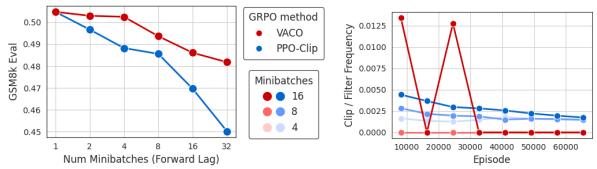


图 *5*。 在 GSM8k 上训练大语言模型进行推理时，VACO 的表现优于 PPO裁剪。（**上**）前向策略滞后可以提高训练效率，但代价是评估性能下降。随着滞后增加，VACO 仍能保持较高性能。（**下**）PPO-Clip 始终会进行裁剪，裁剪幅度与前向滞后成正比。VACO 的过滤频率较低，因此能够从策略滞后较大的样本中学习；同时，在启用过滤时会过滤批次中更大比例的样本，以保持稳定性。

最终会出现前向策略滞后： $N-1$ 到最后一个小批量时，策略领先于数据生成策略 β 步。我们的设置大体遵循 Noukhovitch 等人（2025）的做法，并在小学数学（GSM8k；Cobbe 等人，2021年）上训练 Qwen 2.5 0.5B 基础模型（Qwen 等人，2025）。我们通过将该方法应用于该领域的标准算法 GRPO（Shao 等人，2024年），展示了该方法的灵活性。GRPO 可视为近端策略优化的一种变体，它使用蒙特卡洛采样而非价值函数来估计优势。所有实验细节见附录C.2。

我们采用当前文献中表现最佳的 PPO-Clip 设置运行 GRPO 基线（Yu 等人，2025年）。对于每种小批量大小，我们运行 3 个随机种子： $N \in 1,\ 2,\ 4,\ 8,\ 16,\ 32$ 因为大规模运行通常使用 $N = 16$ 个或更多小批量（Yu 等人，2025年）。在图 5中，我们证实了此前的发现（Noukhovitch 等人，2025年：GSM8k 的评估性能会随着前向策略滞后的增加，从完全同策略状态开始下降 $N = 1$ 。将 PPO裁剪 替换为 VACO 后，我们发现其对高程度前向滞后的鲁棒性显著提高。为展示差异，我们在 $N \in \{4, 8, 16\}$ 的图 5右侧绘制了每个批次中 PPO裁剪 和 VACO 过滤的频率。PPO裁剪 的频率保持不变，并随前向滞后的增加而成比例上升。相比之下，VACO 在前向滞后较低时不会进行任何过滤（$N = 4,\ 8$）。滞后较高时，VACO 会更有选择性地过滤，但过滤的批次比例更大。众所周知，裁剪是 RLVR 的关键组成部分（Yu 等人，2025年），但这些结果表明，裁剪过于激进，许多被裁剪的比率实际上有助于学习。VACO 仍通过过滤维持训练稳定性，同时允许更多批次充分参与学习。

6. 结论

同策略强化学习是许多现代强化学习任务中的主流范式。随着计算预算增加，利用异步训练和离策略数据的高效方法已成为标准做法。但这也带来了策略滞后现象，即学习策略与行为策略之间的不匹配。我们提出了一种全新的理论视角，将这一现象分为反向策略滞后和前向策略滞后。反向策略滞后源于学习策略与行为策略之间最初的不匹配。前向策略滞后则是由于学习策略在使用同一批数据的同时进行梯度更新而导致的。异步强化学习天生容易受到这两类问题的影响，而且已有研究表明，它们会降低性能，限制异步训练带来的效率提升。本文提出了一种新算法 VACO，基于两项核心思路提升对策略滞后的鲁棒性：使优势函数与初始学习策略重新对齐，以及基于全变差散度进行数据过滤。我们的实证结果表明，在两个重要的强化学习领域——机器人学和大语言模型——精心设计的实验设置中，VACO 对策略滞后表现出更强的鲁棒性。我们希望这项工作能助力未来研究进一步突破异步高效训练的局限，同时保持性能。

7. 影响声明

本文旨在推动机器学习领域的发展。我们的工作可能带来许多潜在的社会影响，但我们认为没有哪一项需要在此特别指出。

参考文献

Achiam, J., Held, D., Tamar, A., and Abbeel, P. 约束策略优化。载于国际学习机器会议，第 22–31 页。PMLR, 2017a。

Achiam, J., Held, D., Tamar, A., and Abbeel, P. 约束策略优化。载于国际学习机器会议，第 22–31 页。PMLR, 2017b。

Agarwal, A., Jiang, N., Kakade, S. M., and Sun, W. 强化学习：理论与算法。华盛顿大学西雅图分校计算机科学系，西雅图，华盛顿州，美国，技术报告，32:96，2019。Agarwal, R., Schwarzer, M., Castro, P. S., Courville, A. C., and Bellemare, M. 统计悬崖边缘的深度强化学习。神经信息处理系统进展，34，2021。Altman, E. 受约束马尔可夫决策过程。Routledge，2021。Andrychowicz, M., Raichuk, A., Sta´nczyk, P., Orsini, M., Girgin, S., Marinier, R., Hussenot, L., Geist, M., Pietquin, O., Michalski, M., 等。对于同策略深度演员-评论家方法，哪些因素最重要？一项大规模研究。载于国际学习表征会议。Babaeizadeh, M., Frosio, I., Tyree, S., Clemons, J., and Kautz, J. GA3C：用于深度强化学习的基于GPU的A3C。2016年。Cobbe, K., Kosaraju, V., Bavarian, M., Chen, M., Jun, H., Kaiser, L., Plappert, M., Tworek, J., Hilton, J., Nakano, R., 等。训练验证器以解答数学应用题。*arXiv*预印本 *arXiv:2110.14168*，2021年。Czech, J. 强化学习分布式方法综述。载于强化学习算法：分析与应用，第151–161 页。施普林格，2021年。Espeholt, L., Soyer, H., Munos, R., Simonyan, K., Mnih, V., Ward, T., Doron, Y., Firoiu, V., Harley, T., Dunning, I., 等。IMPALA：采用重要性加权演员-学习器架构的可扩展分布式深度强化学习。载于国际学习机器会议，第 1407–1416 页。PMLR，2018年。

Fakoor, R., Chaudhari, P. 和 Smola, A. J. P3o：在策略-离策略策略优化。载于人工智能中的不确定性，第 1017–1027 页。PMLR，2020。

Gu, S., Holly, E., Lillicrap, T. 和 Levine, S. 使用异步离策略更新进行机器人操作的深度强化学习。载于 *2017 IEEE* 国际机器人与自动化会议（*ICRA*），第 3389–3396 页。IEEE，2017。

Howes, S.、Beton, M.、Baioumy, M.、Cheema, A. 与 Reed, M. Exo Gym：一种用于低带宽训练的仿真环境。载于*ICML25* 倡导机器学习开源开发研讨会，2025年。

Huang, S.、Dossa, R. F. J.、Ye, C.、Braga, J.、Chakraborty, ˜D.、Mehta, K. 与 AraA ˇsjo, J. G. CleanRL：深度强化学习算法的高质量单文件实现。机器学习研究期刊，23(274):1–18，2022。

Huang, S., Weng, J., Charakorn, R., Lin, M., Xu, Z. 和 Onta˜n´on, S. Cleanba：一个可复现且高效的分布式强化学习平台。*arXiv*预印本 *arXiv:2310.00036*，2023。Kakade, S. 和 Langford, J. 近似最优的近似强化学习。载于第十九届国际机器学习大会论文集，第267–274页，2002。Kwon, W., Li, Z., Zhuang, S., Sheng, Y., Zheng, L., Yu, C. H., Gonzalez, J., Zhang, H. 和 Stoica, I. 使用 PagedAttention 高效管理大语言模型服务的内存。载于第二十九届操作系统原理研讨会论文集，第611–626页，2023。Kyzy, A. N., Ji, C.-H., Choi, Y., Seok, Y.-J., Ullah, I. 和 Han, Y.-H. 通过异步近端策略优化加速分布式强化学习的收敛。载于*2025*年国际人工智能与信息通信会议（*ICAIIC*），第1059–1064页。IEEE，2025。Lambert, N., Morrison, J., Pyatkin, V., Huang, S., Ivison, H., Brahman, F., Miranda, L. J. V., Liu, A., Dziri, N., Lyu, S., Gu, Y., Malik, S., Graf, V., Hwang, J. D., Yang, J., Bras, R. L., Tafjord, O., Wilhelm, C., Soldaini, L., Smith, N. A., Wang, Y., Dasigi, P. 和 Hajishirzi, H. Tulu 3：推动开放语言模型后训练的前沿，2025。Langer, M., He, Z., Rahayu, W. 和 Xue, Y. 深度学习模型的分布式训练：分类学视角。*IEEE*并行与分布式系统汇刊，31(12):2802–2818，2020。Liu, Z., Xu, X., Qiao, P. 和 Li, D. 利用并行与分布式计算加速深度强化学习：综述。*ACM*计算综述，57(4):1–35，2024年。

Meng, W., Zheng, Q., Pan, G. 和 Yin, Y. 离策略近端策略优化。载于美国人工智能协会会议论文集，第37卷，第9162–9170页，2023。

Mnih, V., Badia, A. P., Mirza, M., Graves, A., Lillicrap, T., Harley, T., Silver, D. 和 Kavukcuoglu, K. 深度强化学习的异步方法。发表于国际机器学习大会，第 1928–1937 页。PMLR，2016 年。

Munos, R., Stepleton, T., Harutyunyan, A. 和 Bellemare, M. 安全高效的离策略强化学习。发表于神经信息处理系统进展，第 29 卷，2016 年。

Nair, A., Srinivasan, P., Blackwell, S., Alcicek, C., Fearon, R., De Maria, A., Panneershelvam, V., Suleyman, M., Beattie, C., Petersen, S., 等。用于深度强化学习的大规模并行方法。*arXiv* 预印本 *arXiv:1507.04296*，2015年。Nikishin, E., Schwarzer, M., D'Oro, P., Bacon, P.-L., 和 Courville, A. 深度强化学习中的首因偏差。载于国际机器学习会议，第 16828–16847 页。PMLR，2022 年。Noukhovitch, M., Huang, S., Xhonneux, S., Hosseini, A., Agarwal, R., 和 Courville, A. 异步人类反馈强化学习：面向语言模型、更快速高效的离策略强化学习。载于第十三届国际学习表征会议，2025年。Petrenko, A., Huang, Z., Kumar, T., Sukhatme, G., 和 Koltun, V. Sample Factory：借助异步强化学习，以 100000 fps 的速度从像素实现自我中心三维控制。载于国际机器学习会议，第 7652–7662 页。PMLR，2020年。Puterman, M. L.马尔可夫决策过程：离散随机动态规划。John Wiley & Sons，2014 年。Queeney, J., Paschalidis, Y., 和 Cassandras, C. G. 结合样本复用的广义近端策略优化。神经信息处理系统进展，34：11909–11919，2021年。Qwen，：，Yang, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Li, C., Liu, D., Huang, F., Wei, H., Lin, H., Yang, J., Tu, J., Zhang, J., Yang, J., Yang, J., Zhou, J., Lin, J., Dang, K., Lu, K., Bao, K., Yang, K., Yu, L., Li, M., Xue, M., Zhang, P., Zhu, Q., Men, R., Lin, R., Li, T., Tang, T., Xia, T., Ren, X., Ren, X., Fan, Y., Su, Y., Zhang, Y., Wan, Y., Liu, Y., Cui, Z., Zhang, Z., 和 Qiu, Z. Qwen2.5 技术报告，2025年。URL https://arxiv.org/abs/2412.15115。

Radac, M.-B. 和 Chirla, D.-P. 采用同步或异步更新的近实时在线强化学习。《科学报告》，15(1):17158，2025年。

Rudin, N., Hoeller, D., Reist, P. 和 Hutter, M. 使用大规模并行深度强化学习，在几分钟内学会行走。发表于机器人学习会议，第 91–100 页。PMLR，2022。

Schulman, J., Levine, S., Abbeel, P., Jordan, M. 和 Moritz, P. 信赖域策略优化。发表于国际机器学习大会，第 1889–1897 页。PMLR，2015年。

Schulman, J.、Moritz, P.、Levine, S.、Jordan, M. 和 Abbeel, P. 使用广义优势估计进行高维连续控制。*arXiv* 预印本 *arXiv:1506.02438*，2015b。

Schulman, J.、Wolski, F.、Dhariwal, P.、Radford, A. 和 Klimov, O. 近端策略优化算法。*arXiv* 预印本 *arXiv:1707.06347*，2017。Shao, Z.、Wang, P.、Zhu, Q.、Xu, R.、Song, J.、Bi, X.、Zhang, H.、Zhang, M.、Li, Y.、Wu, Y. 等。DeepSeekMath：推动开放语言模型数学推理的极限。*arXiv* 预印本 *arXiv:2402.03300*，2024。Sheng, G.、Zhang, C.、Ye, Z.、Wu, X.、Zhang, W.、Zhang, R.、Peng, Y.、Lin, H. 和 Wu, C. HybridFlow：灵活高效的基于人类反馈的强化学习框架。载于第二十届欧洲计算机系统会议论文集，第 1279–1297 页，2025。Singla, J.、Agarwal, A. 和 Pathak, D. SAPG：拆分与聚合策略梯度。载于第 *41* 届机器学习国际会议论文集，第 45759–45772 页，2024。Stiennon, N.、Ouyang, L.、Wu, J.、Ziegler, D.、Lowe, R.、Voss, C.、Radford, A.、Amodei, D. 和 Christiano, P. F. 通过人类反馈学习摘要。神经信息处理系统进展，33:3008–3021，2020。Sutton, R. S.、Barto, A. G. 等。强化学习：导论，第 1 卷。麻省理工学院出版社，剑桥，1998年。Todorov, E.、Erez, T. 和 Tassa, Y. MuJoCo：用于基于模型的控制的物理引擎。载于 *2012 IEEE/RSJ* 国际智能机器人与系统会议，第 5026–5033 页。IEEE，2012。Vuong, Q.、Zhang, Y. 和 Ross, K. W. 深度强化学习的监督式策略更新。载于国际学习表征会议，2019。

Wang, Y.、He, H. 和 Tan, X. 真正的近端策略优化。载于人工智能中的不确定性，第 113–122 页。PMLR，2020。

Wiering, M. A. 和 Van Hasselt, H. 强化学习中的集成算法。*IEEE*《系统、人类与控制论汇刊*B*辑（控制论）》，38(4):930–936，2008。Wolf, T.、Debut, L.、Sanh, V.、Chaumond, J.、Delangue, C.、Moi, A.、Cistac, P.、Rault, T.、Louf, R.、Funtowicz, M. 等。Transformers：自然语言处理领域的先进技术。载于*2020*年自然语言处理实证方法会议论文集：系统演示，第38–45页，2020。Xie, Z.、Zhang, Q.、Yang, F.、Hutter, M. 和 Xu, R. 简单策略优化。载于第四十二届国际机器学习大会，2025年。Yao, F.、Liu, L.、Zhang, D.、Dong, C.、Shang, J. 和 Gao, J. 你的高效 RL 框架暗中让你进行异策略强化学习训练，2025年8月。网址：h
ttps://fengyao.notion.site/off-policy-rl。Yu, Q.、Zhang, Z.、Zhu, R.、Yuan, Y.、Zuo, X.、Yue, Y.、Dai, W.、Fan, T.、Liu, G.、Liu, L. 等。DAPO：大规模开源大语言模型强化学习系统。*arXiv*预印本 *arXiv:2503.14476*，2025年。Yuan, Y. 和 Mahmood, A. R. 用于实体机器人实时控制的异步强化学习。载于*2022*年国际机器人与自动化会议（*ICRA*），第5546–5552页。IEEE，2022。Ziegler, D. M.、Stiennon, N.、Wu, J.、Brown, T. B.、Radford, A.、Amodei, D.、Christiano, P. 和 Irving, G. 基于人类偏好微调语言模型。*arXiv*预印本 *arXiv:1909.08593*，2019年。

A. 伪代码

算法1VACO**初始化**初始化价值网络和策略网络，V_ϕ，π_θ^1，以及全变差阈值 δ、价值损失、策略损失和熵系数

ts

$c_v, c_\pi, c_\mathcal{H}$对于 $T = 1$，...，T_{max} 执行 # 使用行为策略进行数据收集 收集轨迹 $\mathcal{D} = \{\tau = (s_t, a_t, r_t)_{t=0}^{T_{\max}} \sim \beta_T\}$# 使用优势重对齐估计优势 计算比率 $\pi_\theta^T(a_t|s_t)\ \beta_T(a_t|s_t)$ 使用公式 14 和 15 计算 V-trace目标 $v(s_t, a_t)$ 和优势 $A_{vtrace}(s_t, a_t)$ 对于每个训练轮次执行 # 计算期望总变差散度$\bar{D}_{TV}(\pi_\theta \| \beta_T) = \frac{1}{2N}\sum_{t=1}^{N} |\ ^{\pi_\theta}(a_t|s_t)\ \beta_T(a_t|s_t) - 1|$如果 $D_{TV} > \delta/2$ 则 对于 ~~$t = 1,...,$~~ N执行 如果 $(A_{vtrace}(s_t, a_t) - c_\mathcal{H}) \times \text{sgn}(\pi_\theta(a_t|s_t) - \beta_T(a_t|s_t)) > 0$则 分离梯度 $\pi_\theta(a_t|s_t)$ 结束如果 结束对于 结束如果$L_{\pi_\theta} \leftarrow -\frac{1}{N}\sum_{t=0}^{N}[\ ^{\pi_\theta}(a_t|s_t)\ \beta_T(a_t|s_t)\ (A_{vtrace}(s_t, a_t) - c_\mathcal{H}$ 记录 $\pi_\theta(a_t|s_t))$# 计算价值损失 $L_v \leftarrow \frac{1}{2N}\sum_{t=1}^{N}[V_\phi(s_t) - v(s_t, a_t)]L \leftarrow c_\pi L_\pi + c_v L_v\theta \leftarrow \theta - \eta_\theta \nabla_\theta L,\ \phi \leftarrow \phi - \eta_\phi \nabla_\phi L$ 结束对于 结束对于

B. 理论细节

B.1. 引理证明3.1

我们使用（Agarwal 等人，2019年）来给出证明：

$$J(\pi') - J(\pi) = \mathbb{E}_{\tau\sim\pi'|s_0\sim\mu}[\sum_{t=0}^{\infty}\gamma^t r(s_t, a_t)] - J(\pi) \tag{22}$$

$$= \mathbb{E}_{\tau\sim\pi'|s_0\sim\mu}[\sum_{t=0}^{\infty}\gamma^t (r(s_t, a_t) + V_\pi(s_t) - V_\pi(s_t)] - J(\pi) \tag{23}$$

$$= \mathbb{E}_{\tau\sim\pi'|s_0\sim\mu}[\sum_{t=0}^{\infty}\gamma^t (r(s_t, a_t) + \gamma V_\pi(s_{t+1}) - V_\pi(s_t)] \tag{24}$$

$$= \mathbb{E}_{\tau\sim\pi'|s_0\sim\mu}[\sum_{t=0}^{\infty}\gamma^t (r(s_t, a_t) + \gamma\mathbb{E}[V_\pi(s_{t+1})|s_t, a_t] - V_\pi(s_t)] \tag{25}$$

$$= \mathbb{E}_{\tau\sim\pi'|s_0\sim\mu}[\sum_{t=0}^{\infty}\gamma^t (Q_\pi(s_t, a_t) - V_\pi(s_t)] \tag{26}$$

$$= \mathbb{E}_{\tau\sim\pi'|s_0\sim\mu}[\sum_{t=0}^{\infty}\gamma^t A_\pi(s_t, a_t)] \tag{27}$$

$$= \frac{1}{1-\gamma}\mathbb{E}_{s\sim d^{\pi'}, a\sim\pi'(s)}[A_\pi(s, a)] \tag{28}$$

B.2. 定理3.2的证明

首先，我们评估公式 3.1 中状态分布的变化：

$$J(\pi') - J(\pi) = \frac{1}{1-\gamma}\mathbb{E}_{s\sim d^{\pi'}, a\sim\pi'(s)}[A_\pi(s,a)] \tag{29}$$

$$= \frac{1}{1-\gamma}(\mathbb{E}_{s\sim d^{\pi}, a\sim\pi(s)}[\frac{\pi'(a|s)}{\pi(a|s)}A_\pi(s,a)] + \int_{s\in\mathcal{S}}(d^{\pi'}(s) - d^{\pi}(s))\mathbb{E}_{a\sim\pi'(s)}[A_\pi(s,a)]) \tag{30}$$

此外，对惩罚项应用 H¨older 不等式，对于任意 p， $q \in [1,\infty]$： $1/p + 1/q = 1$ ，有：

$$-\|d^{\pi'} - d^{\pi}\|_p \|\mathbb{E}_{a\sim\pi'(s)}[A_\pi(s,a)]\|_q \leq \int_{s\in\mathcal{S}}(d^{\pi'}(s) - d^{\pi}(s))\mathbb{E}_{a\sim\pi'(s)}[A_\pi(s,a)] \leq \|d^{\pi'} - d^{\pi}\|_p \|\mathbb{E}_{a\sim\pi'(s)}[A_\pi(s,a)]\|_q \tag{31}$$

Fu此外，（Achiam 等人，2017a）中给出的状态分布上界可用于：

引理 B.1。*(Achiam 等人，2017a)* 两个折扣未来状态分布之间的全变差散度，受其各自策略下期望散度的界约束：

$$\|d^{\pi'} - d^{\pi}\|_1 \leq \frac{2\gamma}{1-\gamma}\mathbb{E}_{s\sim d^{\pi}}[D_{TV}(\pi'\|\pi)[s]] \tag{32}$$

关于散度引理的详细证明，请读者参阅（Achiam 等人，2017a）。

选择 $p = 1$， $q = \infty$，将引理B.1 应用于公式31，并通过界定公式29，即可得到该引理的结论。

B.3. 引理4.1的证明

U使用定理3.2 来获得 π_T 与 β 之间以及 π 与 β之间的性能差异 $_T$:

$$J(\pi) - J(\beta_T) \geq \frac{1}{1-\gamma}\mathbb{E}_{s\sim d^{\beta_T}, a\sim\beta_T(s)}[\frac{\pi(a|s)}{\beta_T(a|s)}A_{\beta_T}(s,a) - \frac{2\gamma\epsilon^{\pi}}{1-\gamma}D_{TV}(\beta_T\|\pi)[s]] \tag{33}$$

$$J(\pi_T) - J(\beta_T) \leq \frac{1}{1-\gamma}\mathbb{E}_{s\sim d^{\beta_T}, a\sim\beta_T(s)}[\frac{\pi_T(a|s)}{\beta_T(a|s)}A_{\beta_T}(s,a) + \frac{2\gamma\epsilon^{\pi_T}}{1-\gamma}D_{TV}(\beta_T\|\pi_T)[s]] \tag{34}$$

通过取逆转置第二个不等式并将两个不等式相加，即可得到该界 .

B.4. 引理4.2的证明

证明主要沿用定理3.2的相同步骤。从公式4出发，将折扣未来状态分布从 π 改为 β_T:

$$J(\pi) - J(\pi_T) = \frac{1}{1-\gamma}\left(\mathbb{E}_{s\sim d^{\beta_T}, a\sim\beta_T(s)}[\frac{\pi(a|s)}{\beta_T(a|s)}A_{\pi_T}(s,a)] + \int_{s\in\mathcal{S}}(d^{\beta_T}(s) - d^{\pi}(s))\mathbb{E}_{a\sim\pi(s)}[A_{\pi_T}(s,a)]ds\right) \tag{35}$$

对惩罚项应用 H¨older 不等式，对于任意 p， $q \in [1,\infty]$： $1/p + 1/q = 1$ 我们有：

$$\int_{s\in\mathcal{S}}(d^{\beta_T}(s) - d^{\pi}(s))\mathbb{E}_{a\sim\pi(s)}[A_{\pi_T}(s,a)]ds \leq \|d^{\beta_T} - d^{\pi}\|_p \|\mathbb{E}_{a\sim\pi(s)}[A_{\pi_T}(s,a)]\|_q \tag{36}$$

选取 $p = 1$、 $q = \infty$ 并应用引理B.1 即可得到该结果。

B.5. 优势重新对齐的理论分析

为分析 V-trace 算子的近似精度，（Espeholt 等人，2018年）给出了以下定理:

定理 B.2. *(Espeholt 等人，2018年）定义 V-trace 算子如下:*

$$\mathcal{R}V(s) = V_\beta(s) + \mathbb{E}_{\tau\sim\beta}[\sum_{t=0}^{\infty}\gamma^t(c_0\dots c_{t-1})\rho_t(r_t+\gamma V_\beta(s_{t+1})-V_\beta(s_t))|s_0=s] \tag{37}$$

假设存在 $\alpha\in(0,1]$，使得 $\mathbb{E}_{a\sim\beta(s)}[\rho_0]\geq\alpha$，并令 $\bar{\rho}\geq\bar{c}$，策略 $\pi_{\bar{\rho}}$ 如下:

$$\pi_{\bar{\rho}}(a|s) \coloneqq \frac{\min(\bar{\rho}\beta(a|s),\pi(a|s))}{\sum_{b\in\mathcal{A}}\min(\bar{\rho}\beta(b|s),\pi(b|s))} \tag{38}$$

那么，$\mathcal{R}$ 在 $V_{\pi_{\bar{\rho}}}$ 下有唯一的不动点，并且是由 a 定义的上确界范数下的 η-压缩映射 *s:*

$$\eta = \gamma^{-1}-(\gamma^{-1}-1)\mathbb{E}_{\tau\sim\beta}[\sum_{t\geq0}\gamma^t(\prod_{i=0}^{t-1}c_i)\rho_{t-1}]\leq 1-(1-\gamma)\alpha<1 \tag{39}$$

为完整起见，我们给出该定理的证明。

注意到 $\mathcal{R}$可以改写为:

$$\mathcal{R}V(s) = (1-\mathbb{E}_{a_0\sim\beta(s)}[\rho_0])V_\beta(s) + \mathbb{E}_{\tau\sim\beta}\left[\sum_{t\geq0}\gamma^t\Big(\prod_{i=0}^{t-1}c_i\Big)\Big(\rho_t r_t+\gamma[\rho_t-c_t\rho_{t+1}]V_\beta(s_{t+1})\Big)\right]. \tag{40}$$

因此，我们有:

$$\begin{aligned}\mathcal{R}V_1(s)-\mathcal{R}V_2(s) =&(1-\mathbb{E}_{a_0\sim\beta(s)}[\rho_0])\big[V_\beta^1(s)-V_\beta^2(s)\big]\\ &+\mathbb{E}_{\tau\sim\beta}\left[\sum_{t\geq0}\gamma^{t+1}\Big(\prod_{i=0}^{t-1}c_i\Big)[\rho_t-c_t\rho_{t+1}]\big[V_\beta^1(s_{t+1})-V_\beta^2(s_{t+1})\big]\right] &(41)\\ =&\mathbb{E}_{\tau\sim\beta}\left[\sum_{t\geq0}\gamma^t\Big(\prod_{i=0}^{t-2}c_i\Big)[\rho_{t-1}-c_{t-1}\rho_t]\big[V_\beta^1(s_t)-V_\beta^2(s_t)\big]\right] &(42)\end{aligned}$$

作者将 $c_{-1}=\rho_{-1}=1$ 和 $\prod_{s=0}^{t-2}c_s=1$ 分别记作 $t=0$ 和 1。

此外，由于假设 了 $\rho\geq c$和 $\mathbb{E}_{\tau\sim\beta}[\rho_t]\leq\mathbb{E}_{\tau\sim\beta}[^{\pi(a_t|s_t)}\ _{\beta\ (a_t|s_t)}$，记作 $\alpha_t=\rho_{t-1}-c_{t-1}\rho_t$，我们有:

$$\mathbb{E}_{\tau\sim\beta}\alpha_t = \mathbb{E}\big[\rho_{t-1}-c_{t-1}\rho_t\big]\geq\mathbb{E}_{\tau\sim\beta}\big[c_{t-1}(1-\rho_t)\big]\geq0 \tag{43}$$

因此，价值函数 $\mathcal{R}V_1(s)-\mathcal{R}V_2(s)$ 之间的差异是由价值函数构成的加权线性组合

，这些价值函数位于轨迹 $V_{\beta}^{1}(s_t) - V_{\beta}^{2}(s_t)$ 中，且系数非负，其和为：

$$\sum_{t\geq 0}\gamma^t\mathbb{E}_{\tau\sim\beta}\left[\sum_{t\geq 0}\Big(\prod_{i=0}^{t-2}c_i\Big)[\rho_{t-1}-c_{t-1}\rho_t]\right] \tag{44}$$

$$=\sum_{t\geq 0}\gamma^t\mathbb{E}_{\tau\sim\beta}\left[\Big(\prod_{i=0}^{t-2}c_i\Big)\rho_{t-1}\right]-\sum_{t\geq 0}\gamma^t\mathbb{E}_{\tau\sim\beta}\left[\Big(\prod_{i=0}^{t-2}c_i\Big)\rho_t\right] \tag{45}$$

$$=\sum_{t\geq 0}\gamma^t\mathbb{E}_{\tau\sim\beta}\left[\Big(\prod_{i=0}^{t-2}c_i\Big)\rho_{t-1}\right]-\frac{1}{\gamma}\Big(\sum_{t\geq 0}\gamma^t\mathbb{E}_{\tau\sim\beta}\left[\Big(\prod_{i=0}^{t-2}c_i\Big)\rho_{t-1}\right]-1\Big) \tag{46}$$

$$=\gamma^{-1}-(\gamma^{-1}-1)\sum_{t\geq 0}\gamma^t\mathbb{E}_{\tau\sim\beta}\left[\Big(\prod_{i=0}^{t-2}c_i\Big)\rho_{t-1}\right] \tag{47}$$

$$\leq 1-(1-\gamma)\mathbb{E}_{a\sim\beta(s)}\rho_0 \tag{48}$$

$$\leq 1-(1-\gamma)\alpha \tag{49}$$

$$<1 \tag{50}$$

因此，可以得出 $\|\mathcal{R}V_1(s) - \mathcal{R}V_1(s)\| \leq \eta\|V_{\beta}^{1}(s) - V_{\beta}^{2}(s)\|_{\infty}$，且 $1-(1-\gamma)\alpha \leq 1$，这使得 $\mathcal{R}$ 成为压缩映射。

最后，为了证明 $\mathcal{R}$ 是 $\pi_{\bar{\rho}}$ 的不动点，我们有：

$$\mathbb{E}_{\tau\sim\beta}[\rho_t(r(s_t,a_t)+\gamma V_{\pi_{\bar{\rho}}}(s_{t+1})-V_{\pi_{\bar{\rho}}}(s_t)|s_t] \tag{51}$$

$$=\sum_{a\in\mathcal{A}}\beta(a|s_t)\min(\bar{\rho},\frac{\pi(a|s_t)}{\beta(a|s_t)})[r(s_t,a)+\gamma\sum_{s'\in\mathcal{S}}\mathcal{P}(s'|s_t,a)V_{\pi_{\bar{\rho}}}(s')-V_{\pi_{\bar{\rho}}}(s_t)] \tag{52}$$

$$=\sum_{a\in\mathcal{A}}\pi_{\bar{\rho}}(a|s_t)\left[r(s_t,a)+\gamma\sum_{s'\in\mathcal{S}}\mathcal{P}(s'|s_t,a)V_{\pi_{\bar{\rho}}}(s')-V_{\pi_{\bar{\rho}}}(s_t)\right]\sum_{b\in\mathcal{A}}\min(\bar{\rho}\beta(b|s_t),\pi(b|s_t)) \tag{53}$$

$$=0 \tag{54}$$

零 Bellman 误差意味着 $V_{\pi_{\bar{\rho}}}$ 是 V-trace 算子的唯一不动点。

B.6. 使用全变差散度衡量策略滞后与使用 KL散度的比较

Kullback–Leibler（KL）散度是衡量分布之间统计距离的最常用指标之一。本节将讨论，与 KL散度相比，使用全变差散度作为策略滞后的度量所带来的优势。如第 4.2 节所述，在第 T 次迭代中，当使用行为策略 β_T 生成轨迹时，两个策略 π 和 π_T 之间的性能差异可以利用公式 12 得到下界。

此外，正如（Achiam 等人，2017a）所指出的，利用 Pinsker 不等式（$D_{TV}(p\|q) \leq \sqrt{D_{KL}(p\|q)/2}$）和 Jensen 不等式，可以将期望总变差散度表示为以下上界：

$$\mathbb{E}_{s\sim d^{\beta_T}}[D_{TV}(\beta_T\|\pi)[s]] \leq \sqrt{\mathbb{E}_{s\sim d^{\beta_T}}[D_{KL}(\beta_T\|\pi)[s]]/2} \tag{55}$$

结合等式 12 和等式，我们得到（也可以从等式 10 推导出类似的下界）：

$$\begin{aligned} J(\pi)-J(\pi_T) &\geq \frac{1}{1-\gamma}\mathbb{E}_{s\sim d^{\beta_T},a\sim\beta_T(s)}\left[\frac{\pi(a|s)}{\beta_T(a|s)}A_{\pi_T}(s,a)-\frac{2\gamma\epsilon^{\pi}}{1-\gamma}D_{TV}(\beta_T\|\pi)[s]\right] \\ &\geq \frac{1}{1-\gamma}\mathbb{E}_{s\sim d^{\beta_T},a\sim\beta_T(s)}\left[\frac{\pi(a|s)}{\beta_T(a|s)}A_{\pi_T}(s,a)\right]-\frac{2\gamma\epsilon^{\pi}}{(1-\gamma)^2}\sqrt{\frac{1}{2}\mathbb{E}_{s\sim d^{\beta_T}}\left[D_{KL}(\beta_T\|\pi)[s]\right]} \end{aligned} \tag{56}$$

表 *1.* PPO、SPO、IMPALA 和 VACO 的重要超参数

超参数	值
裁剪比率/全变差阈值	0.2
学习率	3e-4
学习率退火	True
环境数量	500
步数	1000
折扣因子	0.99
小批次数量	32
训练轮数	10
最大梯度范数	0.5
$\bar{\rho}$	1
$\bar{c}$	1

因此，使用 KL 正则化目标函数来优化策略如今已成为一种标准做法（Schulman 等人，2017；2015年）。不过，我们首先指出，当 $\sqrt{\mathbb{E}[D_{KL}(\beta_T\|\pi)[s]]} \geq 1$ 时，KL 正则化在理论上是有效的，并能保证策略改进，而我们通常选择用于约束 KL散度的阈值往往小于 1。

更重要的是，KL散度可能带来更严峻的优化挑战。请看（Wang 等人，2020）中的定理 3：

定理 B.3。（*Wang 等人，2020*）假设对于具有 $|\mathcal{A}| > 3$ 的离散动作空间任务，策略参数化为$\pi_\theta(s_t) = p_t \in \mathbb{R}^{+|\mathcal{A}|}$，且满足 $\sum_{i=1}^{|\mathcal{A}|} p_t^{(i)} = 1$；对于连续动作空间任务，策略参数化为$\pi_\theta(a|s_t) = \mathcal{N}(a|\mu_t, \sigma_t)$。定义可行参数空间 $\Theta = \{\theta | 1-\delta \leq \frac{\pi_\theta(s,a)}{\beta_T(s,a)} \leq 1+\delta \; \forall s, a \in \mathcal{S} \times \mathcal{A}\}$，则对于离散和连续动作空间任务，均有最大值$\theta \in \Theta \; D_{KL}(\pi_\theta\|\beta_T)[s_t] = +\infty$。

因此，可行参数空间内可能存在导致 KL散度无穷大的策略。因此，若使用 $D_{KL} \leq \delta$ 来构建约束策略优化，许多可行策略都会被排除。相较之下，很容易发现，同一参数空间 Θ 内的所有策略均满足 $\max_{\theta \in \Theta} D_{TV}(\pi_\theta\|\beta_T)[s_t] \leq \delta/2$。因此，全变差散度为策略优化提供了更紧的下界，并能在优化过程中涵盖更广泛的策略。

C. 实验设置详情

C.1. 模拟异步设置算法超参数

我们使用（Huang 等人，2022）提出的默认超参数，在模拟异步设置中训练强化学习算法。表 1讨论了一些重要的超参数。对于计算优势重对齐所用的超参数值，我们遵循（Espeholt 等人，2018年）的超参数选择。

C.2. GSM8k RLVR 设置与超参数

我们主要遵循 Noukhovitch 等人（2025年）的设置，在 GSM8k 上进行训练，但将模型替换为现代的 Qwen 2.5 0.5B 基础模型（Qwen 等，2025年）。我们使用 GSM8k 的 ≈8000 个样本进行训练（Cobbe等人，2021年），并使用评估集中的 ≈1300 个样本进行评估。我们的超参数主要遵循 Yao 等人（2025年），总体上复现了他们的$N = 1$ 小批量结果。所有超参数均列于 表 2。为与尽可能强的基线进行比较，我们遵循 Yu 等人（2025年），在 GRPO 中使用他们分别设定的低裁剪值和高裁剪值。

由于不存在反向滞后，我们将优势重新对齐比率设为 1。我们注意到，另一种方法即使在完全同策略强化学习可验证奖励中仍会使用优势重新对齐项。正如 Yao 等人 （2025年）所指出的，数据在生成引擎 vLLM 下的对数概率与其在训练库 transformers 下的对数概率不同。因此，我们仍可以使用 $\beta = \pi_{vllm}$ 来实现优势重对齐，其中它可被视为一种用于解决

与截断重要性采样相同的问题 Yao 等人 （TIS；2025年）。但对于我们训练与生成均采用 BF16 的特定场景，Yao 等人 （2025年）指出，重新对齐项将始终非常接近 1，并且不会带来实证差异。

表 2. GRPO、PPO-Clip 和 VACO 在 GSM8k 上的超参数

超参数	值
PPO-Clip 下限比率	0.2
PPO-Clip 上限比率	0.272
全变差阈值	0.05
学习率	1e-6
每个小批次的提示数	32
每个提示的补全数	8
总回合数	65536
步数	256
PPO 轮数	1
提示长度	512
补全长度	512
温度（训练）	1.0
Top-p（训练）	1.0
温度（评估）	0.
Top-p（评估）0.95	

D. 实验详情

D.1. 算法样本效率比较

除了图 4 展示了不同异步程度下算法在整个训练阶段的总体性能外，图 6 还展示了算法在所有环境和不同异步程度下的未归一化回报。

D.2. 对 δ 阈值取值的消融实验

我们对阈值 δ 对 VACO 性能的影响进行了消融分析。为此，图 7 展示了 VACO 在训练过程中的多种取值下的性能。结果表明，VACO 对较为激进的 δ取值具有较强的鲁棒性。这一点在图 8中可以更清楚地观察到。因此，采用约束策略优化的 VACO，即使在反向策略滞后程度较高且阈值取值较为激进的情况下，通常也能有效抵御策略崩溃。

D.3. $\bar{\rho}$阈值取值的消融实验

我们对 $\bar{\rho}$ 对 VACO 性能的影响进行了实验。正如第4.2 节和第B.5节所讨论的， $\bar{\rho}$控制着 V-trace目标的不动点。因此，该取值可能会对所提出的算法产生重要影响。如图9 和图10所示，我们的结果证实了原始 IMPALA 论文（Espeholt 等人，2018年）中的结论：与较高取值相比，IMPALA 在 $\bar{\rho} = 1$ 时的整体性能更佳。

D.4. 基于总变差散度的过滤效果

图 11 比较了 VACO 和近端策略优化在有无 KL惩罚时，每次训练所得最终策略的平均全变差散度。我们观察到，尽管近端策略优化的全变差散度值较低，但要根据裁剪比率预测其数值会很困难。另一方面，VACO 的结果表明，所提出的算法能够维持相同水平的全变差散度。因此，这种特性使实践者能够可靠地调整约束值，在追求最优性的同时，将全变差散度维持在合理且保守的水平。

D.5. 优势重对齐的效果

为了研究优势重对齐对 VACO 的影响，我们还分别运行了启用和未启用重对齐过程的算法。结果如图 12所示。我们在训练过程中观察到一个重要现象：随着训练过程中加入更多异步数据，如果使用价值函数的最新版本计算 v-trace值，算法的表现会更好。事实上，在某些情况下，如果不使用最新版本，算法的表现并不理想。

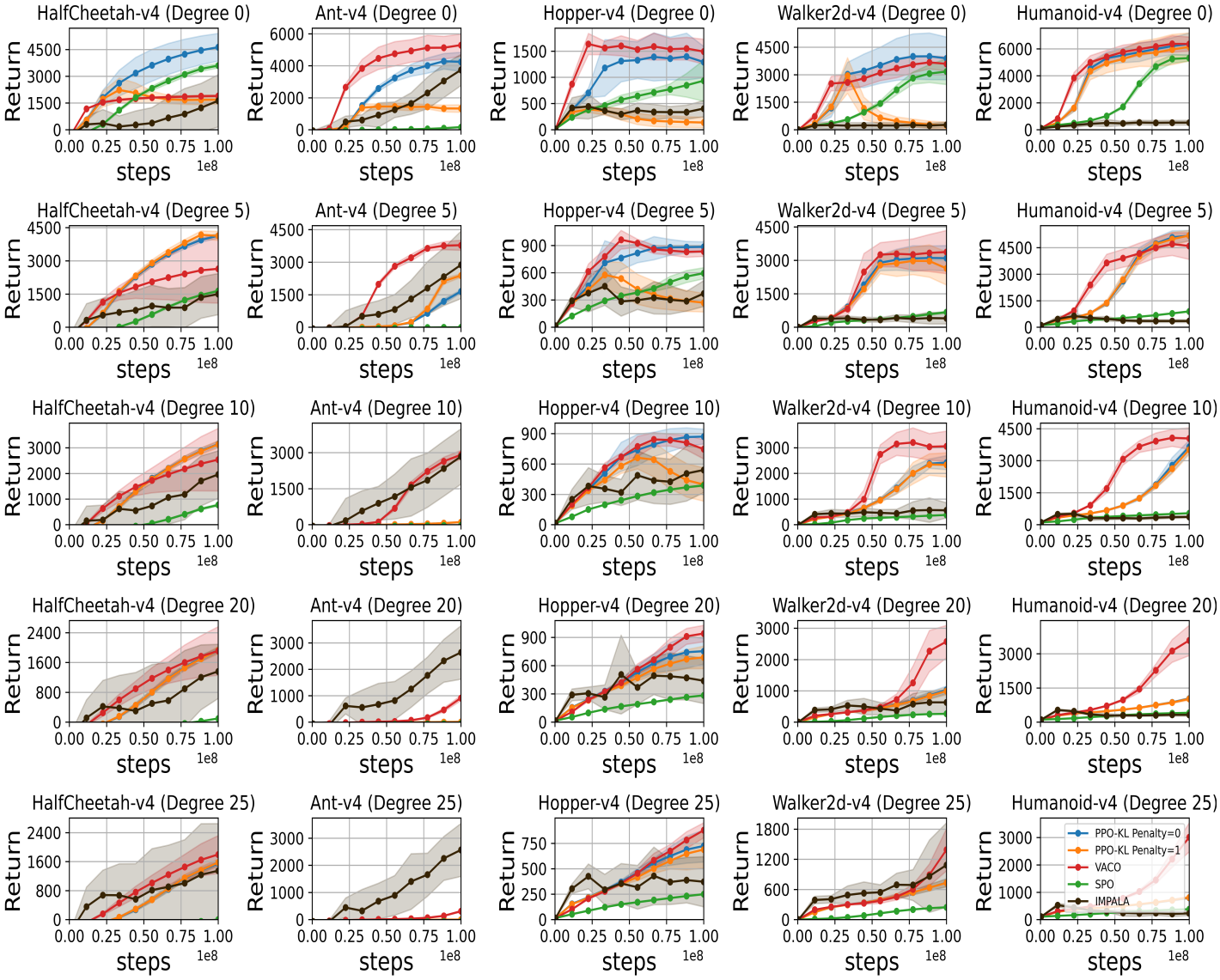


图 *6.* 五种环境下，对比算法在五种异步程度中的回报图。该图展示了策略在 10 个随机种子上的平均回报，阴影部分表示回报的标准差。

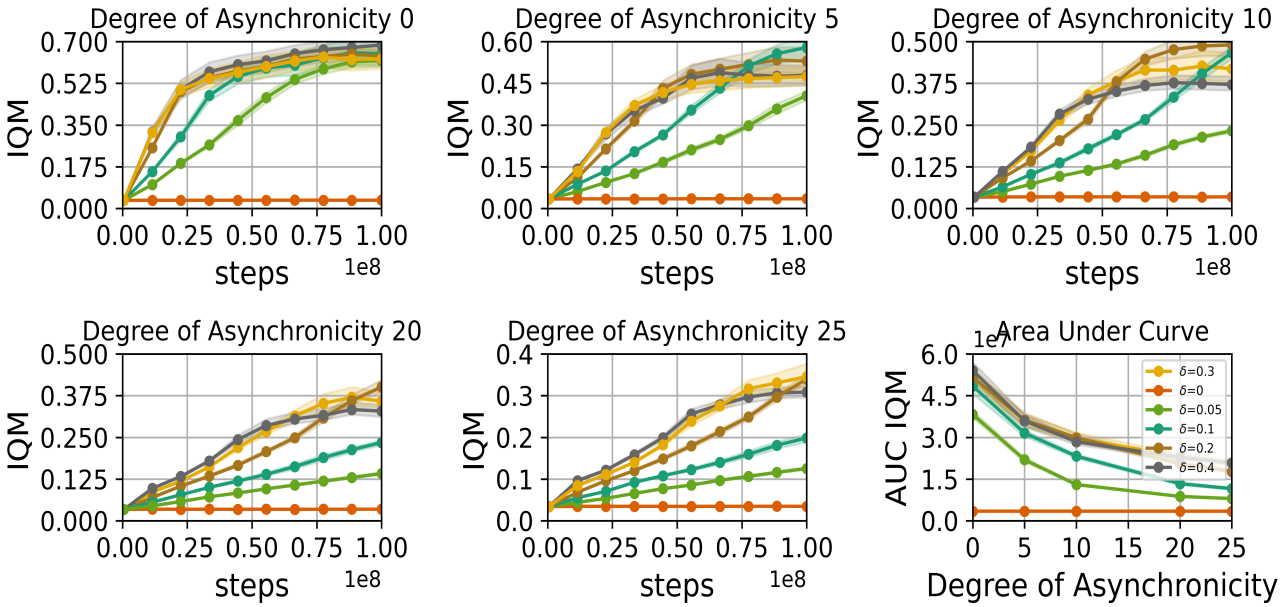


图 7. 模拟异步设置下训练过程中，VACO 在不同 δ 全变差散度阈值下的 IQM 值。得分是在 10 个独立随机种子下，基于 100M 个步骤计算得出的，阴影区域表示 95%置信区间。**(右下)** 归一化回报图曲线下面积的 IQM 值。数值越高，表示训练过程中的样本效率越高。

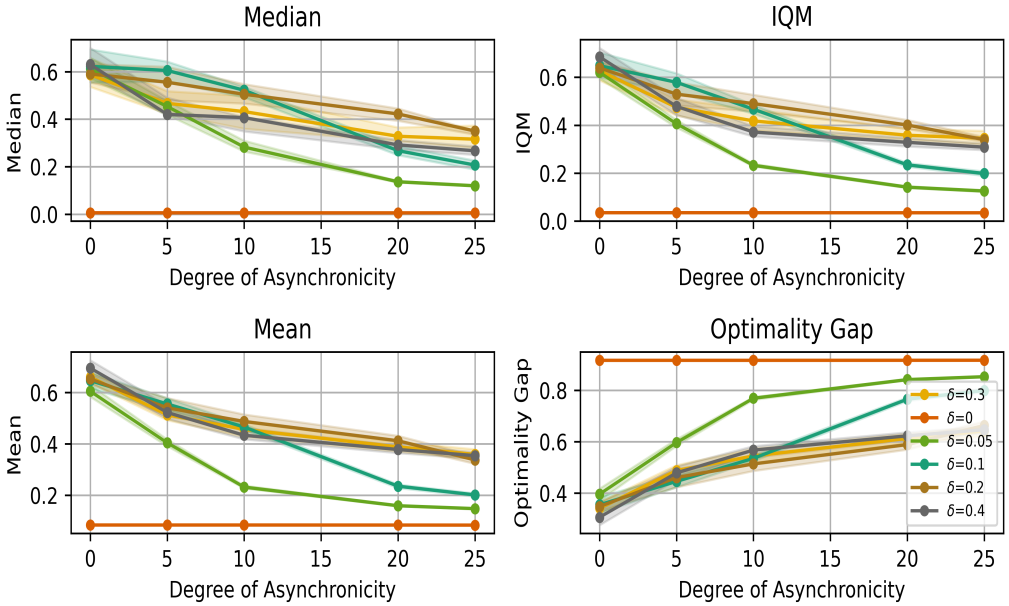


图 8. 对 VACO 在不同全变差散度阈值 δ下的性能进行消融分析。中位数、IQM 和均值越高，最优性差距越低，表示性能越好。得分是在 10 个独立随机种子下，基于 100M 个步骤计算得出的，阴影区域表示各项指标的 95%置信区间。

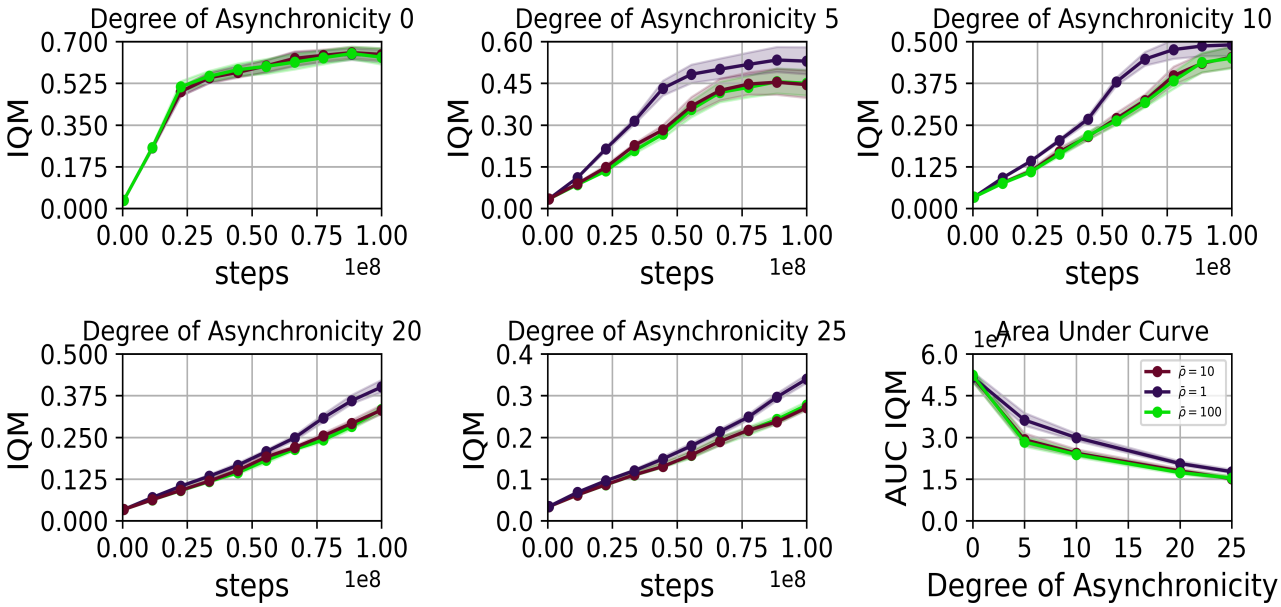


图 *9.* 在训练过程中，模拟异步设置下用于优势重对齐的 ρ 取不同值时 VACO 的 IQM 值。分数基于 10 个独立随机种子在 1 亿步训练中的结果计算，阴影部分表示 95%置信区间。（**右下**）归一化回报图曲线下面积的 IQM 值。数值越高，表示训练过程中的样本效率越高。

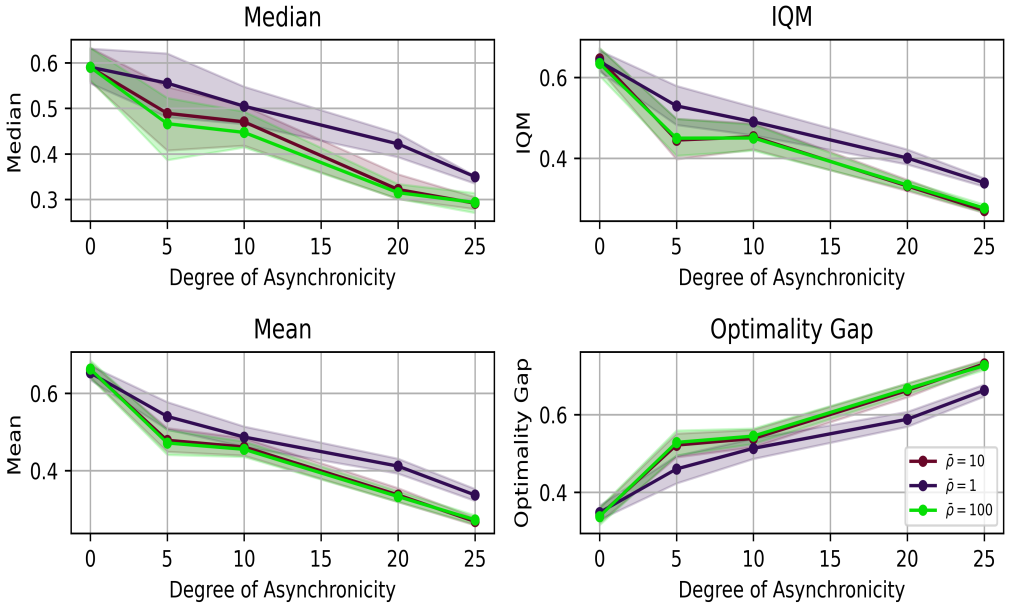


图 *10.* 对用于优势重对齐的 ρ 取不同值时 VACO 性能的消融分析。中位数、IQM 和均值越高，最优性差距越低，表示性能越好。分数基于 10 个独立随机种子在 1 亿步训练中的结果计算，阴影部分表示各项指标的 95%置信区间。

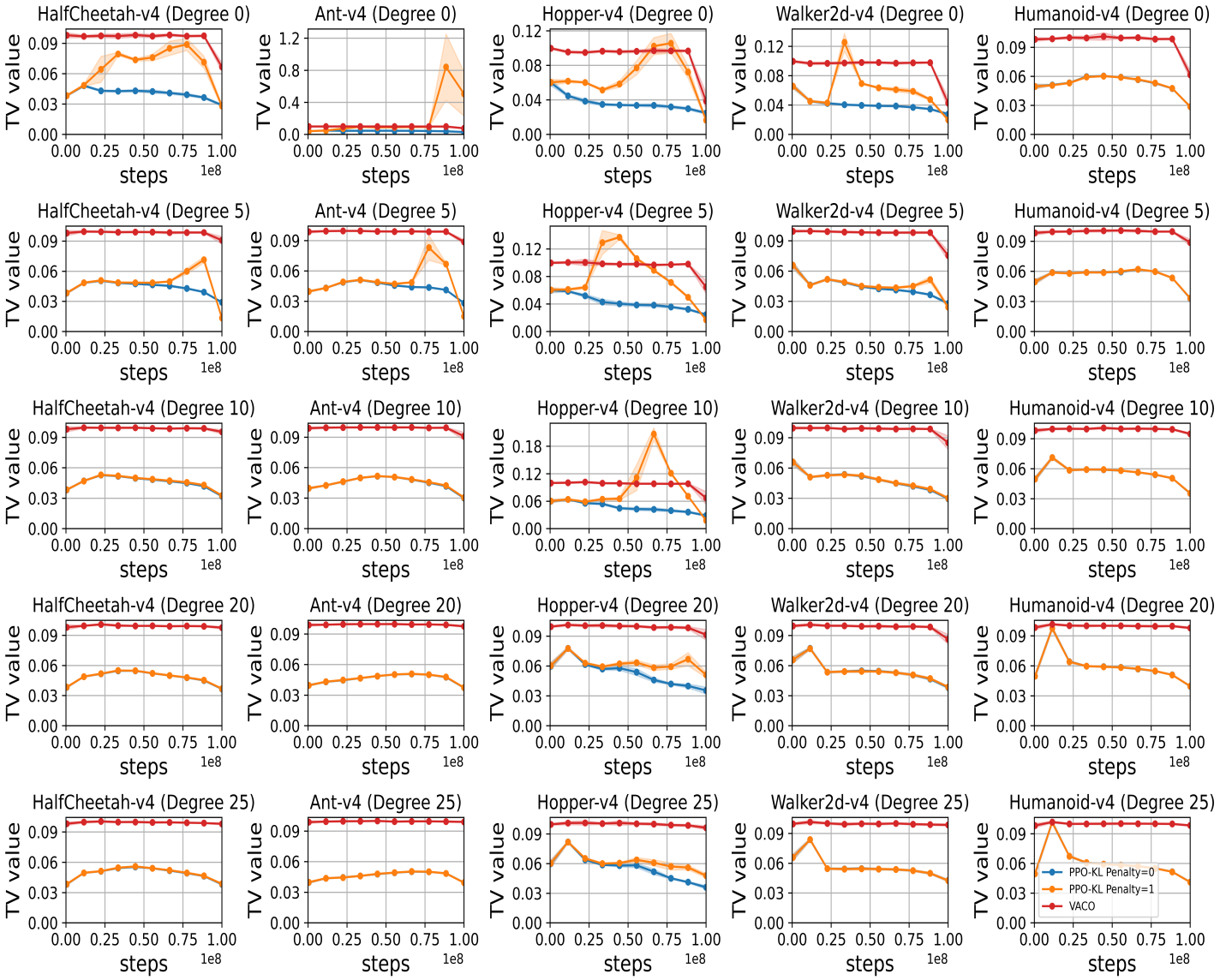


图 *11.* 在模拟的异步强化学习训练设置中，VACO 和近端策略优化的训练阶段最终策略与行为策略分布之间的平均全变差散度。可以看到，在满足约束 $1/2\mathbb{E}[|\pi/\beta - 1|] \leq \delta/2$ where $\delta = 0.2$ 的情况下，VACO 在所有环境和不同异步程度下始终保持相同水平的全变差散度。该图展示了基于 10 个随机种子的全变差散度平均估计值，阴影部分表示标准差。

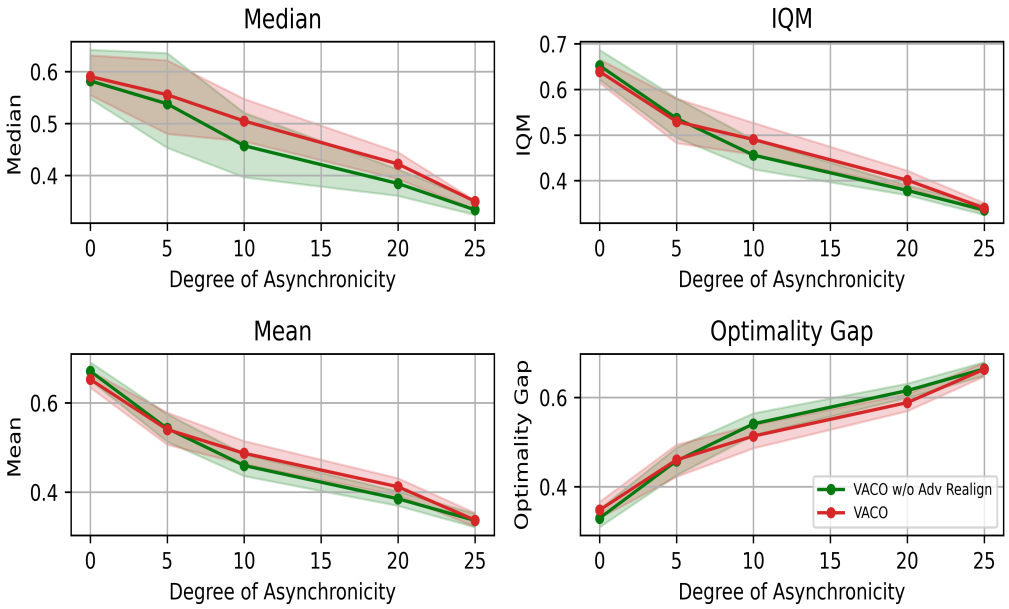


图*12*。VACO 在采用和未采用优势重对齐过程时的性能。可以看到，平均而言，回报重对齐能更好地提升对离策略数据的鲁棒性。分数是在 10 个独立随机种子下，经过 100M 步计算得出的，阴影区域表示 95%置信区间。