arXiv:2302.03921v2 [cs.LG] 3 Jun 2023

Pre面向无监督基于模型的强化学习的可预测MDP抽象

Seohong Park [1] **Sergey Levine** [1]

摘要

基于模型的强化学习（RL）的一个关键组成部分是能够预测动作结果的动力学模型。该预测模型中的误差会降低基于模型的控制器的性能，而复杂的马尔可夫决策过程（MDP）可能带来极难解决的预测问题。为缓解这一问题，我们提出**可预测MDP抽象**（**PMA**）：我们不在原始MDP上训练预测模型，而是在经过变换的MDP上训练模型；该MDP具有学习得到的动作空间，只允许可预测、易于建模的动作，同时尽可能覆盖原始状态-动作空间。这样一来，模型学习会变得更简单、更准确，从而能够实现稳健、稳定的基于模型的规划或基于模型的强化学习。这种变换以无监督的方式学习，在用户指定任何任务之前即可完成。随后，下游任务便能以零样本方式通过基于模型的控制来解决，无需与环境进行额外交互。我们从理论上分析了PMA，并通过实验证明，在一系列基准测试环境中，PMA相较于以往的无监督基于模型的强化学习方法取得了显著提升。代码和视频见
https://seohong.me/projects/pma/

1. 引言

基于模型的强化学习（RL）算法的基本构件是预测模型$\hat{}p(\boldsymbol{s}'|\boldsymbol{s},\boldsymbol{a})$，通常用于在给定的马尔可夫决策过程（MDP）中，根据前一状态和动作预测下一状态。通过将预测模型与规划或强化学习结合使用，以往的基于模型的方法已被证明能够以较高的样本效率解决各种复杂问题，涵盖机器人领域（Wu 等人，2022年）和游戏领域（Schrit- twieser 等人，2020年）。

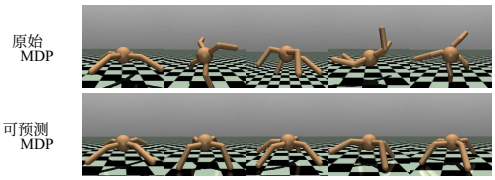


图 *1.* 在原始 Ant 环境中，某些动作会导致不可预测的行为，难以准确建模[2]，，使学习得到的动力学模型容易遭受灾难性模型利用。在转换后的可预测MDP中，每次状态转移都易于建模且可以预测，因此能够实现稳健、稳定的基于模型的学习。

然而，预测模型即使存在微小误差，也可能导致基于模型的强化学习算法表现不佳，有时甚至造成灾难性后果（Hasselt 等人，2019年；Jafferjee 等人，2020年）。这种现象称为模型利用，指控制器或策略利用这些误差，选择模型错误预测为能够带来良好结果的动作。当模型展开的时域较长时，这个问题会进一步加剧，因为预测误差会随时间累积；在复杂的马尔可夫决策过程中也是如此，因为准确地建模所有状态转移颇具挑战。以往的方法通常通过估计模型的不确定性来应对模型利用（Chua 等人，2018年），或仅将模型用于较短的轨迹展开（Buckman 等人，2018年；Janner 等人，2019年）。

我们从不同的角度看待基于模型的强化学习，以应对这一挑战：我们不在原始MDP上训练预测模型，而是在经过抽象和简化的MDP上应用基于模型的强化学习。具体而言，我们首先通过限制不可预测的动作，将MDP抽象为一个更简单的学习得到的MDP，并转换其潜在动作空间，然后在这个简化后的MDP上构建预测模型。这里，转换后的MDP旨在具备可预测性，即新MDP中的每次状态转移都易于建模，同时尽可能多地覆盖原始状态-动作空间。因此，与原始的、可能复杂的MDP相比，灾难性模型利用的空间很小，从而能够实现稳健的基于模型的规划和强化学习。图1展示了可预测MDP转换的一个示例。

我们针对基于无监督模型的

[1]加州大学伯克利分校。联系邮箱：Seo- hongPark < seohong@berkeley.edu>。

《 40th 国际机器学习会议，美国夏威夷州檀香山。机器学习研究论文集 202，2023。版权所有 © 2023 作者。

[2]尽管 Ant环境 完全确定，但由于其复杂且具有丰富接触动力学的特点，要准确建模所有可能的状态转移仍然非常困难（参见第7.3节）。

强化学习，设计了一种实用算法，用于学习可预测MDP抽象。在这种设置下，抽象会预先学习，无需用户定义任务或奖励函数。无监督训练完成后，学习得到的 MDP 抽象便可与基于模型的控制器结合，以零样本方式解决多个不同的下游任务，无需任何环境交互或额外的模型训练。

无监督可预测MDP抽象有三项要求。首先，变换后的MDP中的潜在动作应产生可预测的状态转移。其次，不同的潜在动作应产生不同的结果。第三，潜在MDP中的状态转移应尽可能覆盖原始状态-动作空间。本文将这些要求形式化为一个信息论目标，并提出一种实用方法对其进行优化。

总而言之，本文的主要贡献是提出**可预测MDP抽象**（**PMA**）作为一种无监督基于模型的强化学习方法，为基于模型的强化学习提供了一个新视角。该方法通过变换动作空间来抽象MDP，从而尽量减少模型误差。PMA可与任何现有的基于模型的规划或强化学习方法相结合，以零样本方式解决下游任务。我们从理论上分析了PMA，并讨论了与经典的基于模型的强化学习相比，本方法在哪些情况下更有优势。最后，我们通过实验证实，PMA与基于模型的强化学习相结合，能够在七种多样的机器人环境中稳健地解决各种任务，显著优于以往的无监督基于模型方法。

2. 相关工作

基于模型的强化学习。基于模型的强化学习（MBRL）通过使用预测模型来估计给定环境中动作的结果。以往的基于模型的方法利用学习得到的模型，通过规划（Hernandaz 和 Arkun，1990；Draeger 等人，1995；Deisenroth 和 Rasmussen，2011；Lenz 等人，2015；Ebert 等人，2018；Chua 等人，2018；Hafner 等人，2019；Nagabandi 等人，2019）、强化学习（Heess 等人，2015；Feinberg 等人，2018；Buckman 等，2018年；Janner 等，2019；Hafner 等人，2020；Nguyen 等人，2021）或两者结合（Argenson 和 Dulac-Arnold，2021；Sikchi 等人，2022；Hansen 等人，2022）来最大化奖励。错误的预测模型可能对策略学习产生有害影响，这种现象称为模型利用问题（Ross 和 Bagnell，2012；Janner 等，2019；Kidambi 等人，2020；Kang 等人，2022）。为避免基于错误模型做出次优决策，先前研究要么限制模型展开的时域长度（Janner 等，2019），要么采用各种不确定性估计技术，例如高斯过程（Rasmussen 和 Kuss，2003；Deisenroth 和 Rasmussen，2011）或模型集成（Rajeswaran 等人，2017；Clavera 等人，2018；Kurutach 等人，2018；Chua 等人，2018；Nagabandi 等人，2019；Yu等人，2020；Kidambi 等人，2020）。我们的工作与这些方法相互独立且互为补充基于模型的强化学习算法。我们提出一种动作表征学习方法，将马尔可夫决策过程抽象为一个更具可预测性的过程，从而简化模型学习，并且可以与多种现有的基于模型的强化学习方法结合使用。

可预测行为学习。我们的核心思路在概念上与此前有关可预测行为学习的研究相关。其中一项研究是 RPC（Eysenbach 等人，2021年），该方法通过最小化模型误差，鼓励智能体产生可预测的行为。SMiRL（Berseth 等人，2021年）和 IC2（Rhinehart 等人，2021年）则通过降低不确定性，主动寻求稳定行为。尽管这些方法鼓励智能体以可预测的方式行动或访问熟悉的状态，但它们的目标并非提供基于模型的强化学习方法，而是将可预测性奖励用于内在动机（Berseth 等人，2021年；Rhinehart 等人，2021年）或提升鲁棒性（Eysenbach 等人，2021年）。与之相反，我们表明，针对可预测性进行优化能够显著提升基于模型的强化学习性能。

马尔可夫决策过程抽象与分层强化学习。马尔可夫决策过程抽象旨在构建简化的马尔可夫决策过程，通常通过简化状态空间或动作空间，使强化学习更易处理。状态抽象（Li 等人，2006年；Watter 等人，2015年；Ha 与 Schmidhuber，2018年；Gelada 等人，2019年；Castro，2019年；Hafner 等人，2020年）着重于获得紧凑的状态表征，以促进学习。时间抽象与分层强化学习（Sutton 等人，1999年；Stolle 与 Precup，2002年；Bacon 等人，2017年；Vezhnevets 等人，2017年；Machado 等人，2017年；Nachum 等人，2018年；Eysenbach 等人，2019年；Wulfmeier 等人，2021年；Salter 等人，2022年）旨在学习在时间上延展的行为，以减少高层决策步数。与这些先前方法不同，我们探索一种有损的马尔可夫决策过程抽象方法，将动作空间变换为仅允许产生更可预测状态转移的空间，从而更有效地进行基于模型的强化学习。

无监督强化学习。无监督强化学习的目标是通过与环境进行无监督交互，获取对下游任务有用的基元、模型或其他对象。使用我们的方法学习可预测MDP抽象的过程，对应于一种无监督强化学习流程。以往的无监督强化学习方法利用内在奖励来最大化状态熵（Lee等人，2019年；Yarats等人，2021年；Liu与Abbeel，2021年）、检测新颖状态（Pathak等人，2017年；Burda等人，2019年；Pathak等人，2019年）、学习多样化的目标条件策略（Pong等人，2020年；Mendonca等人，2021年），或获取时间延展技能（Gregor等人，2016年；Eysenbach等人，2019年；Sharma 等人，2020年；Xie等人，2020年；Strouse 等人，2022年；Park等人，2022年；Laskin等人，2022年）。值得注意的是，若干无监督的基于模型方法（Shyam 等人，2019年；Sekar 等人，2020年；Rajeswar等人，2022年）已表明，通过以下方式训练的预测模型

无监督探索有助于高效解决下游任务。然而，这些方法只关注寻找新颖的状态转移（即，最大化覆盖率），却没有考虑其可预测性。最大化覆盖率而不考虑可预测性会导致模型误差，进而造成模型利用；Shyam 等人Shyam 等人(2019年)对此进行了展示，我们的实验也证实了这一点。我们的方法还与 DADS（Sharma 等人，2020年）密切相关；DADS 是一种无监督技能发现方法，使用了与我们相似的互信息（MI）目标。不过，我们工作的重点与 DADS 不同：DADS 的目标是获取一组时间延展技能，与其他技能发现研究类似；而我们的重点是将动作空间转换为一种不涉及时间抽象、只允许可预测动作的空间，同时最大限度地覆盖原始MDP中的状态转移。我们通过理论和实证均表明，这能显著提升多种基于模型的强化学习框架的性能。

3. 预备知识与问题陈述

我们考虑一个不含奖励函数的 MDP，也称为受控马尔可夫过程（CMP），$\mathcal{M} := (\mathcal{S}, \mathcal{A}, \mu, p)$，其中 $\mathcal{S}$ 表示状态空间，$\mathcal{A}$ 表示动作空间，$\mu \in \mathcal{P}(\mathcal{S})$ 表示初始状态分布，$p: \mathcal{S} \times \mathcal{A} \to \mathcal{P}(\mathcal{S})$ 表示状态转移分布。我们还考虑一组 N 下游任务 $\mathcal{T} = \{T_0, T_1, \ldots, T_{N-1}\}$，其中每个任务对应一个奖励函数 $r_i: \mathcal{S} \times \mathcal{A} \to \mathbb{R}$，适用于 $i \in [N]$。$[N]$ 表示 $\{0, 1, \ldots, N-1\}$的集合。我们将绝对奖励的上确界记为 $R = \sup_{\boldsymbol{s} \in \mathcal{S},\ \boldsymbol{a} \in \mathcal{A},\ i \in [N]} |r_i(\boldsymbol{s}, \boldsymbol{a})|$，并将折扣因子记为 γ。

问题陈述。在这项工作中，我们解决无监督的基于模型的强化学习问题，该问题包含两个阶段。在第一个无监督训练阶段，我们旨在给定一个 CMP，在不知道任务的情况下构建预测模型。在随后的测试阶段，我们在同一环境中获得多个任务奖励，并旨在仅使用学习得到的模型来解决这些任务，而无需额外训练；即，以零样本方式。因此，我们的目标是构建一个能够最佳地捕捉环境特征的模型，以便之后能够稳健地使用该模型解决各种任务。

4. 可预测MDP抽象（PMA）

基于模型的强化学习方法通常会学习一个模型 $\hat{p}(\boldsymbol{s}'|\boldsymbol{s}, \boldsymbol{a})$。然而，朴素地对所有可能的状态转移进行建模会导致误差——在复杂环境中容易出现此类问题，而后续的控制方法（规划或策略优化）可能会利用这些误差，导致基于模型的策略回报估计过于乐观，最终造成性能不佳。以往的研究通常通过限制模型的使用（Buckman 等人，2018年；Janner 等人，2019年）或更准确地估计不确定性（Chua 等人，2018年）来解决这一问题。

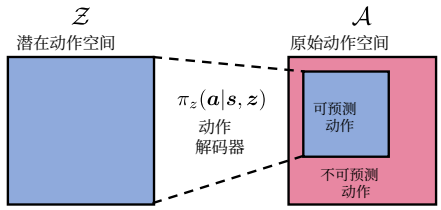


图2. 动作解码器对动作空间进行重新参数化，使其只允许可预测的状态转移。

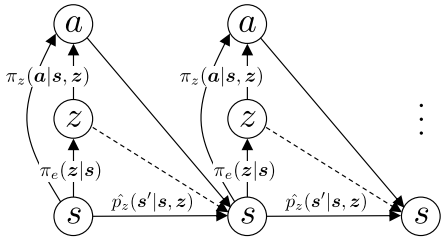


图3. PMA 在无监督训练期间的架构。探索策略 $\pi_e(\boldsymbol{z}|\boldsymbol{s})$ 选择一个潜在动作，该动作由动作解码器 $\pi_z(\boldsymbol{a}|\boldsymbol{s}, \boldsymbol{z})$ 解码到原始动作空间。潜在模型 $\hat{p}_z(\boldsymbol{s}'|\boldsymbol{s}, \boldsymbol{z})$ 预测潜在MDP中的结果。

与以往方法不同，我们在这项工作中的解决方案是将原始马尔可夫决策过程转化为一个可预测的潜在马尔可夫决策过程，其中每次状态转移都是可预测的。此处，“可预测”也意味着易于建模。形式上，我们将马尔可夫决策过程 $\mathcal{M}$ 的不可预测性定义为：对于模型类 $\mathcal{F}$、状态-动作分布 d 和差异度量 D，可能达到的最小平均模型误差 ϵ：

$$\epsilon = \inf_{f \in \mathcal{F}} \mathbb{E}_{(\boldsymbol{s}, \boldsymbol{a}) \sim d(\boldsymbol{s}, \boldsymbol{a})} [D(p(\cdot|\boldsymbol{s}, \boldsymbol{a}) \| f(\cdot|\boldsymbol{s}, \boldsymbol{a}))]. \quad (1)$$

直观而言，这衡量的是在模型类 $\mathcal{F}$ 的容量限制下，环境中不可约的模型误差（即，偶然不确定性）。值得注意的是，即使在完全确定性的环境中，如果模型类 $\mathcal{F}$ 的容量有限（例如，大小为 (512，512) 的神经网络），且环境动态复杂，也可能存在不可约的模型误差。

将原始MDP转换为简化的潜在MDP后，我们在潜在MDP之上使用基于模型的强化学习来解决下游任务。由于潜在MDP经过训练，具有尽可能强的可预测性，因此与原始环境相比，模型利用的空间很小。因此，之后我们可以稳健地使用学到的潜在预测模型，通过基于模型的控制方法解决下游任务。

4.1 架构

PMA 的核心思想是以有损方式限制原始动作空间，使其仅允许可预测的状态转移。形式上，我们将原始MDP $\mathcal{M}$ 转换为一个定义为 $\mathcal{M}_P$ 的可预测潜在MDP：$= (\mathcal{S}, \mathcal{Z}, \mu, p_z)$

其中，原始状态空间为 $\mathcal{S}$，潜在动作空间为 $\mathcal{Z}$，潜在转移动态为 p_z： $\mathcal{S} \times \mathcal{Z} \to \mathcal{P}(\mathcal{S})$。

PMA 包含三个可学习组件：动作解码器、潜在预测模型和探索策略。动作解码器策略 $\pi_z(\boldsymbol{a}|\boldsymbol{s}, \boldsymbol{z})$ 将潜在动作解码到原始动作空间，从而以有损方式重新参数化动作空间（图2）。潜在预测模型 $\hat{p}_z(\boldsymbol{s}'|\boldsymbol{s}, \boldsymbol{z})$ 预测潜在MDP中的下一个状态，并与动作解码器联合训练，使学习到的MDP尽可能具有可预测性。最后，探索策略 $\pi_e(\boldsymbol{z}|\boldsymbol{s})$ 选择 $\boldsymbol{z}$来训练无监督训练阶段的 PMA 组件，从而最大限度地覆盖原始状态空间。图3展示了 PMA 的各个组件。

在构建潜在预测模型 $\hat{p}_z(\boldsymbol{s}'|\boldsymbol{s}, \boldsymbol{z}; \boldsymbol{\theta})$ 时，我们从贝叶斯视角推导目标，将信息论表征学习目标与参数 $\boldsymbol{\theta}$ 后验分布上的信息增益相结合。这自然促成了信息寻求探索目标，以及一种针对潜在动作空间的基于互信息的表征学习方法。

无监督训练结束后，在测试阶段获得奖励函数后，我们会将探索策略替换为 任务策略 $\pi(\boldsymbol{z}|\boldsymbol{s})$，该策略旨在基于我们的动作解码器和预测模型选择潜在动作，以解决下游任务。正如我们将在第 5 节中所述，该任务策略既可以通过规划器推导得出，也可以在学习得到的模型中通过强化学习自行学习。这两种方法都以零样本方式运行，也就是说，除无监督阶段外，它们不需要与环境进行任何额外交互。

4.2. 目标

现在，我们阐述无监督可预测MDP抽象的三项期望性质：(*i*) 可预测MDP中的潜在转移$p_z(\boldsymbol{s}'|\boldsymbol{s}, \boldsymbol{z})$应尽可能可预测（即，尽量降低偶然不确定性）。(*ii*) 潜在动作的结果应彼此尽可能不同（即，最大化动作多样性，以尽可能保留原始MDP的表达能力）。(*iii*) 可预测MDP中的状态转移应尽可能多地覆盖原始状态转移（即，通过降低认知不确定性来促进探索）。这三个目标可以概括为以下简洁的信息论目标：

$$\max_{\pi_z, \pi_e} I(\boldsymbol{S}'; (\boldsymbol{Z}, \boldsymbol{\Theta})|\mathcal{D}), \tag{2}$$

其中，$\mathcal{D}$ 表示截至并包括当前状态 $\boldsymbol{S}$ 在内的整个训练数据集的随机变量（RV），$(\boldsymbol{S}, \boldsymbol{Z}, \boldsymbol{S}')$ 表示来自策略的 $(\boldsymbol{s}, \boldsymbol{z}, \boldsymbol{s}')$ 元组的随机变量，而 $\boldsymbol{\Theta}$ 表示潜在预测模型 $\hat{p}_z(\boldsymbol{s}'|\boldsymbol{s}, \boldsymbol{z}; \boldsymbol{\theta})$ 参数的随机变量。直观而言，该目标要求学习一个由 π_z 表示的潜在动作空间，以及该潜在动作空间中的探索策略 π_e，使得每次状态转移后的下一状态都易于根据潜在动作和模型参数进行预测。纳入模型参数或许看起来不太寻常，但这自然地引出了最大化状态覆盖的信息增益探索方案。

公式（2）可以分解如下，揭示了与我们的设计目标直接对应的三个项：

$$\begin{aligned} & I(\boldsymbol{S}'; (\boldsymbol{Z}, \boldsymbol{\Theta})|\mathcal{D}) && (3)\\ =& I(\boldsymbol{S}'; \boldsymbol{Z}|\mathcal{D}) + I(\boldsymbol{S}'; \boldsymbol{\Theta}|\mathcal{D}, \boldsymbol{Z}) && (4)\\ =& I(\boldsymbol{S}'; \boldsymbol{Z}|\boldsymbol{S}) + I(\boldsymbol{S}'; \boldsymbol{\Theta}|\mathcal{D}, \boldsymbol{Z}) && (5)\\ =& \underbrace{-H(\boldsymbol{S}'|\boldsymbol{S}, \boldsymbol{Z})}_{(i)\text{ predictability}} + \underbrace{H(\boldsymbol{S}'|\boldsymbol{S})}_{(ii)\text{ diversity}} \\ & + \underbrace{H(\boldsymbol{\Theta}|\mathcal{D}, \boldsymbol{Z}) - H(\boldsymbol{\Theta}|\mathcal{D}, \boldsymbol{Z}, \boldsymbol{S}')}_{(iii)\text{ information gain}}. && (6) \end{aligned}$$

公式（6）中的第一项通过降低下一状态分布的熵 p_z（$\boldsymbol{s}'|\boldsymbol{s}, \boldsymbol{z}$）来最大化可预测性，使潜在MDP尽可能可预测。第二项增加边缘化后的下一状态分布的熵，从而使不同 z所产生的状态彼此不同。第三项通过最大化信息增益来最小化认知不确定性；信息增益指的是在获知 $\boldsymbol{S}'$后，预测模型参数不确定性的降低。

将公式（2）中的目标作为内在奖励，我们就可以通过强化学习优化动作解码器策略和探索策略。因此，它们将学会生成最优的 $\boldsymbol{z}$和 $\boldsymbol{a}$，从长远来看，这些将使原始状态空间的覆盖范围最大化，同时让潜在MDP尽可能可预测。此外，我们还会同时训练潜在预测模型 $\hat{p}_z$（$\boldsymbol{s}'|\boldsymbol{s}, \boldsymbol{z}; \boldsymbol{\theta}$），以便之后用于潜在MDP中的规划。

4.3. 实际实现

现在，我们介绍一种实用方法，用于优化公式（2）中的主要目标。由于通常无法精确估计互信息或信息增益，我们采用了若干近似方法。

估计 $I(\boldsymbol{S}'; \boldsymbol{Z}|\boldsymbol{S})$。首先，对于公式（6）中的前两项，我们采用如下变分下界近似（Barber 和 Agakov，2003年）：

$$\begin{aligned} & I(\boldsymbol{S}'; \boldsymbol{Z}|\boldsymbol{S}) = -H(\boldsymbol{S}'|\boldsymbol{S}, \boldsymbol{Z}) + H(\boldsymbol{S}'|\boldsymbol{S}) && (7)\\ & \geq \mathbb{E}[\log \hat{p}_z(\boldsymbol{s}'|\boldsymbol{s}, \boldsymbol{z}; \boldsymbol{\phi}) - \log p_z(\boldsymbol{s}'|\boldsymbol{s})] && (8)\\ & \approx \underbrace{\mathbb{E}[\log \hat{p}_z(\boldsymbol{s}'|\boldsymbol{s}, \boldsymbol{z}; \boldsymbol{\phi}) - \log \frac{1}{L} \sum_{i=1}^{L} \hat{p}_z(\boldsymbol{s}'|\boldsymbol{s}, \boldsymbol{z}_i; \boldsymbol{\phi})]}_{:= r_{\text{emp}}(\boldsymbol{s}, \boldsymbol{z}, \boldsymbol{s}')}, && (9) \end{aligned}$$

其中，$\hat{p}_z(\boldsymbol{s}'|\boldsymbol{s}, \boldsymbol{z}; \boldsymbol{\phi})$ 是 $p_z(\boldsymbol{s}'|\boldsymbol{s}, \boldsymbol{z})$ 的变分下界（VLB）。此外，我们对难以处理的边缘

熵项 $H(\boldsymbol{S}'|\boldsymbol{S})$，使用 L 从 $\boldsymbol{z}$的潜在动作空间中采样得到的随机样本（Sharma 等人，2020年）。这些近似使我们能够得到可处理的内在奖励，并通过强化学习进行优化。这里需要指出的是，方程（9）中的第二项是 p_z（$\boldsymbol{s}'|\boldsymbol{s}$）的有偏估计量，因为它使用样本来估计 log 内部的期望，但我们发现这种方法在实践中仍然表现良好；事实上，先前研究也曾在强化学习中的互信息目标中采用过这种有偏估计量（Sharma 等人，2020年；Kim 等人，2021年）。探索针对这一互信息目标的无偏下界（Poole 等人，2019年）是未来研究的一个有趣方向。

估计信息增益。接下来，我们需要估计公式（6）中的信息增益项。我们可以直接采用先前提出通过信息增益进行探索的方法来近似这一项，例如使用变分近似（Houthooft 等人，2016）。不过，在我们的实现中，我们采用了一种基于集成分歧的启发式近似方法；受先前研究的启发（Shyam 等人，2019；Ball 等人，2020；Sekar 等人，2020；Strouse 等人，2022），我们发现这种方法实现起来更简单。具体来说，我们首先用一个由 E 个预测模型组成的集成来近似模型后验，$\{\hat{}p_z(\boldsymbol{s}'|\boldsymbol{s},\boldsymbol{z};\boldsymbol{\theta}_i)\}_{i\in[E]}$，其中 $p(\boldsymbol{\theta}|\mathcal{D}) = \frac{1}{E}\sum_i \delta(\boldsymbol{\theta}-\boldsymbol{\theta}_i)$。每个分量都将状态转移建模为条件高斯分布，其均值由神经网络给出，且具有单位对角协方差，$\boldsymbol{s}' \sim \mathcal{N}(\boldsymbol{\mu}(\boldsymbol{s},\boldsymbol{z};\boldsymbol{\theta}_i)\ I)$。随后，我们使用集成均值的方差，并乘以系数 β，

$$\mathbb{E}[\underbrace{\beta \cdot \mathrm{Tr}[\mathbb{V}_i[\boldsymbol{\mu}(\boldsymbol{s},\boldsymbol{z};\boldsymbol{\theta}_i)]]}_{:=r_{\text{dis}}(\boldsymbol{s},\boldsymbol{z},\boldsymbol{s}')}], \tag{10}$$

作为信息增益的一种简单（尽管粗略）估计量$I(\boldsymbol{S}';\boldsymbol{\Theta}|\boldsymbol{\mathcal{D}}\ \boldsymbol{Z})$。我们在附录 C 中给出了详细的论证。直观而言，公式（10）鼓励智能体探索此前未访问过、且集成模型的预测彼此不一致的状态。

训练 PMA。采用这些近似后，我们将$r_{\text{emp}}(\boldsymbol{s},\boldsymbol{z},\boldsymbol{s}')$ $+ r_{\text{dis}}(\boldsymbol{s},\boldsymbol{z},\boldsymbol{s}')$用作动作解码器 $\pi_z(\boldsymbol{a}|\boldsymbol{s},\boldsymbol{z})$的内在奖励。对于探索策略$\pi_e(\boldsymbol{z}|\boldsymbol{s})$，若假设动作解码器是最优的，则可以将 $H(\boldsymbol{Z}|\boldsymbol{S})\ + r_{\text{dis}}(\boldsymbol{s},\boldsymbol{z},\boldsymbol{s}')$用作内在奖励。我们可以使用任意最大熵强化学习方法来优化它。不过在实践中，我们发现，在实验中简单地采用最大熵策略 $\pi_e(\cdot|\boldsymbol{s})\ = \mathrm{Unif}(\mathcal{Z})$就已足够，因为我们的动作解码器也会最大化$r_{\text{dis}}(\boldsymbol{s},\boldsymbol{z},\boldsymbol{s}')$。最后，我们使用从策略中采样得到的（$\boldsymbol{s},\boldsymbol{z},\boldsymbol{s}'$）元组，拟合 VLB预测模型$\hat{}p_z(\boldsymbol{s}'|\boldsymbol{s},\boldsymbol{z};\boldsymbol{\phi})$和集成模型 $\{\hat{}p_z(\boldsymbol{s}'|\boldsymbol{s},\boldsymbol{z};\ \boldsymbol{\theta}_i)\}$。PMA 的完整训练流程见附录 F 和算法 1。

4.4. 与先前工作的联系

PMA 的目标与无监督强化学习中的若干先前研究有关。例如，若我们令 $\beta\ =\ 0$ 和$\pi_e(\boldsymbol{z}_t|\boldsymbol{s}_t)\ = \boldsymbol{z}_{t-1}$ 取 $t \geq 1$，便可得到 DADS（Sharma 等人，2020年），这是一种先前提出的无监督技能发现方法。此外，若我们令 $\pi_z(\boldsymbol{a}|\boldsymbol{s},\ \boldsymbol{z})\ = \boldsymbol{z}$取，PMA 就与采用基于分歧的内在奖励的先前无监督的基于模型方法相似（Shyam 等人，2019年；Sekar 等人，2020年）。然而，这些方法要么不以覆盖状态-动作空间为目标，要么不考虑可预测性，因此效果欠佳或不稳定。在第7节中，我们通过实证比较 PMA 与这些先前方法，并证明我们的完整目标函数带来了显著提升。此外，我们在附录E中从理论上比较了 PMA 与 DADS。

5. 基于 PMA 的模型学习

完成 PMA 的无监督训练后，我们可以利用学得的潜在预测模型，通过基于模型的规划或强化学习来优化奖励函数。本节将介绍几种利用可预测MDP以零样本方式解决下游任务的方法。

5.1. 基于 PMA 的模型学习

训练模型后，PMA 可以与任何现有的基于模型的规划或强化学习方法结合。具体而言，我们可以在潜在动作空间 $\mathcal{Z}$ 和学得的潜在动力学模型 $\hat{}p_z(\boldsymbol{s}'|\boldsymbol{s},\ \boldsymbol{z})$ 上应用任意现成的基于模型的强化学习方法，以最大化下游任务奖励；其中，我们将集成模型输出的均值作为潜在动力学模型。通过在潜在动作空间上进行规划，我们可以有效避免模型利用，因为难以预测的动作会被过滤掉。

在我们的实验中，我们研究了基于模型的学习的两种可能实现：一种基于模型预测控制，另一种基于近似动态规划（即., actor-critic强化学习），其中学得的模型被用作“模拟器”，无需额外采集环境样本。请注意，这两种变体都能以“零样本”的方式解决新任务，也就是说，它们不需要在环境中额外采集任何真实状态转移。

模型预测路径积分（MPPI）。MPPI（Williams 等人，2016年）是一种基于采样的零阶规划算法，它会在每个时间步优化一段短时域的（潜在）动作序列，执行序列中的第一个动作，然后重新规划。完整的训练流程请参阅附录F.2 和算法2 。

基于模型的策略优化（MBPO）。MBPO（Janner 等人，2019年）是一种 Dyna风格的基于模型的强化学习算法，它从环境的中间状态开始进行截断的基于模型的展开，并在此基础上训练无模型强化学习方法。在我们的零样本设置中，我们训练任务策

略仅使用纯粹的基于模型的展开，其起始状态从无监督训练中恢复的回放缓冲区中采样。完整的训练流程请参阅附录F.3 和算法3。

5.2. 应对分布偏移

使用固定的预训练模型进行基于模型的控制，本质上容易受到分布偏移的影响，因为测试时控制器可能会找到一些"对抗性" $\boldsymbol{z}$ 值，使智能体状态偏离分布。即使在我们的零样本设置中，这个问题同样存在，尽管潜在MDP中的每次状态转移都经过训练，具有可预测性。因此，我们明确惩罚智能体访问分布外状态，这遵循了先前的离线基于模型的强化学习方法（Yu等人，2020年；Kidambi等人，2020年），这些方法也处理了同样的问题。由于我们已经拥有一个潜在预测模型集成，因此我们使用集成模型之间的以下最大分歧（Kidambi等人，2020年）作为惩罚项，其系数为 λ:

$$u(\boldsymbol{s},\boldsymbol{z}) = -\lambda \cdot \max_{i,j\in[E]} \|\mu(\boldsymbol{s},\boldsymbol{z};\boldsymbol{\theta}_i) - \mu(\boldsymbol{s},\boldsymbol{z};\boldsymbol{\theta}_j)\|^2. \quad (11)$$

在特定任务规划或强化学习期间，我们将此惩罚项加到任务奖励上，方式与MOPO（Yu等人，2020年）类似。

6. 理论结果

可预测MDP抽象是一种有损过程。本节从理论上分析这种损失对策略在抽象MDP中的性能所产生的影响程度，并就相比经典基于模型的强化学习，PMA 何时能够发挥作用提供实践层面的见解。所有形式化定义和证明均可在附录D中找到。

6.1. PMA 性能界

我们首先给出 PMA 的性能界。形式化地，对于原始MDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mu, p, r)$，我们将具有学习得到的动力学模型 $\hat{p}$ 的 MDP 定义为 $\hat{\mathcal{M}} = (\mathcal{S}, \mathcal{A}, \mu, \hat{p}, r)$，其中假设 $\hat{p}$ 是在由 $\pi_{\mathcal{D}}$ 收集的数据集 $\mathcal{D}$ 上训练得到的。对于我们的可预测潜在MDP $\mathcal{M}_P = \hat{} (\mathcal{S}, \mathcal{Z}, \mu, p_z, r)$，我们类似地定义 $\mathcal{M}_P = (\mathcal{S}, \mathcal{Z}, \mu, \hat{p}_z, r)$、$\mathcal{D}_P$ 和 $\pi_{\mathcal{D}_P}$。对于原始MDP中的策略 $\pi(\boldsymbol{a}|\boldsymbol{s})$，我们将其对应的、最能模仿原始策略的潜在策略定义为 $\pi_z^{\phi^*}(\boldsymbol{z}|\boldsymbol{s})$，并将其诱导的下一状态分布定义为 $p_z^{\phi^*}(\boldsymbol{s}'|\boldsymbol{s},\boldsymbol{a})$（形式化定义请参见附录D）。现在，我们给出具有学习得到的潜在动力学模型的 PMA 性能界，如下所示：

定理6.1。 如果抽象损失、模型误差和策略差异满足以下界限：

$$\mathbb{E}_{(\boldsymbol{s},\boldsymbol{a})\sim d^{\pi}(\boldsymbol{s},\boldsymbol{a})}[D_{\mathrm{TV}}(p(\cdot|\boldsymbol{s},\boldsymbol{a})\|p_z^{\phi^*}(\cdot|\boldsymbol{s},\boldsymbol{a}))] \le \epsilon_a,$$
$$\mathbb{E}_{(\boldsymbol{s},\boldsymbol{z})\sim d^{\pi_{\mathcal{D}_P}}(\boldsymbol{s},\boldsymbol{z})}[D_{\mathrm{TV}}(p_z(\cdot|\boldsymbol{s},\boldsymbol{z})\|\hat{p}_z(\cdot|\boldsymbol{s},\boldsymbol{z}))] \le \epsilon'_m,$$
$$\mathbb{E}_{\boldsymbol{s}\sim d^{\pi_{\mathcal{D}_P}}(\boldsymbol{s})}[D_{\mathrm{TV}}(\pi_z^{\phi^*}(\cdot|\boldsymbol{s})\|\pi_{\mathcal{D}_P}(\cdot|\boldsymbol{s}))] \le \epsilon'_\pi,$$

π 原始*MDP* 与可预测的潜变量模型马尔可夫决策过程之间的性能差异可以

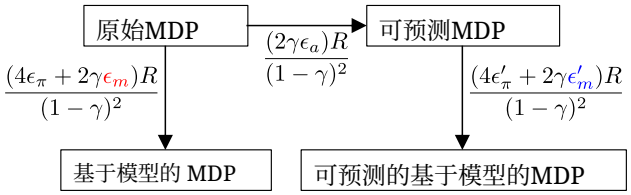


图 4 $\epsilon_m \gg \epsilon_a + \epsilon'_m$。四种 MDP 的性能界。当，PMA 提供的界比经典基于模型的强化学习更紧。

界定为：

$$|J_{\mathcal{M}}(\pi) - J_{\hat{\mathcal{M}}_P}(\pi_z^{\phi^*})| \le \frac{R}{(1-\gamma)^2}(2\gamma\epsilon_a + 4\epsilon'_\pi + 2\gamma\epsilon'_m). \quad (12)$$

直观而言，PMA 的性能界由以下三个因素构成：(*i*) 有损动作解码器造成的性能损失程度 (ϵ_a)、(*ii*) 潜在预测模型中的模型误差 (ϵ'_m)，以及 (*iii*) 数据收集策略与任务策略之间的分布偏移 (ϵ'_π)。因此，如果状态-动作覆盖范围更广、模型误差更低，性能界就会更紧；这正是 PMA 力求实现的目标（公式 (6)）。

6.2. 何时应优先选择 PMA，而非经典基于模型的强化学习?

为了深入了解 PMA 的实际效果，我们从理论上比较 PMA 与经典基于模型的强化学习。我们首先给出经典基于模型的强化学习的性能界 (Janner 等人，2019年)：

定理 6.2。 若模型误差和策略差异满足以下有界条件：

$$\mathbb{E}_{(\boldsymbol{s},\boldsymbol{a})\sim d^{\pi_{\mathcal{D}}}(\boldsymbol{s},\boldsymbol{a})}[D_{\mathrm{TV}}(p(\cdot|\boldsymbol{s},\boldsymbol{a})\|\hat{p}(\cdot|\boldsymbol{s},\boldsymbol{a}))] \le \epsilon_m,$$
$$\mathbb{E}_{\boldsymbol{s}\sim d^{\pi_{\mathcal{D}}}(\boldsymbol{s})}[D_{\mathrm{TV}}(\pi(\cdot|\boldsymbol{s})\|\pi_{\mathcal{D}}(\cdot|\boldsymbol{s}))] \le \epsilon_\pi,$$

$\hat{}\,\pi$ 在 $\mathcal{M}$ 与 $\hat{\mathcal{M}}$ 之间的性能差异可界定为：

$$|J_{\mathcal{M}}(\pi) - J_{\hat{\mathcal{M}}}(\pi)| \le \frac{R}{(1-\gamma)^2}(4\epsilon_\pi + 2\gamma\epsilon_m). \quad (13)$$

比较公式（12）和公式（13）可知，当 $4\epsilon_\pi + 2\gamma\epsilon_m > 2\gamma\epsilon_a + 4\epsilon'_\pi + 2\gamma\epsilon'_m$（图 4）时，PMA 能够得到更紧的界。直观而言，如果我们假设两种情况下数据收集策略与目标策略 π的偏离程度相近，那么这一条件对应于 $\epsilon_m \gg \epsilon_a + \epsilon'_m$。这表明，当可预测转移带来的模型误差降低幅度超过抽象损失时，PMA 可能比经典基于模型的强化学习更有益。

PMA 何时能在实践中发挥作用? PMA 在任务的最优策略主要由可预测的状态转移构成时很有用，因此我们可以降低模型误差

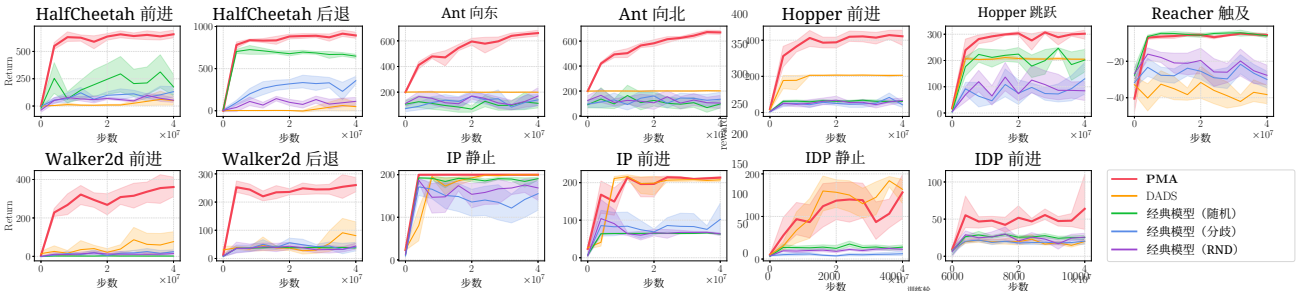


图5。不同无监督基于模型的强化学习方法在周期性零样本规划（MPPI 与 MOPO 惩罚项结合）中的性能比较。PMA 在大多数任务中表现最佳，尤其是在 Ant 和 Walker2d 上。

ϵ'_m 同时保持较小的 ϵ_a （即我们关注的最优策略在转移分布上的平均抽象损失）。例如，在现实世界的驾驶场景中，我们无需对多种不可预测的行为进行建模（甚至可以主动避免这些行为），就能实现大多数驾驶目的，例如以各种方式损坏汽车，因此 PMA 很有益处。正如我们在第7节通过实证展示的那样，类似的论点也适用于许多机器人环境。另一方面，我们可以设想这样的马尔可夫决策过程：最优行为要求有意进入状态空间中不可预测的区域，在这种情况下，PMA 可能不是最优的。

7. 实验

在实验中，我们研究 PMA 作为无监督模型学习算法，用于下游零样本 基于模型的强化学习时的性能，并分析 PMA 学习更具可预测性的模型、从而支持更长时域模拟轨迹展开的程度。具体而言，我们旨在回答以下问题：(*i*) PMA 能否在各种任务中带来更好的零样本任务表现？(*ii*) PMA 能否实现稳健的长时域规划，同时避免模型利用？(*iii*) PMA 能否学习到更具可预测性的模型？

实验设置。由于 PMA 在模型训练过程中不需要访问任务奖励，我们将比较重点放在其他采用类似假设的无监督基于模型的强化学习方法上：在预训练阶段可以与马尔可夫决策过程交互，但不能访问其奖励函数；随后进入零样本评估阶段。以往的无监督基于模型方法（Shyam等人，2019；Pathak等人，2019；Sekar 等，2020；Rajeswar等人，2022）通常会利用探索策略收集的数据，预训练形式为 $\hat{}p(\boldsymbol{s}'|\boldsymbol{s},\boldsymbol{a})$ 的经典动力学模型。我们选取其中三种作为基线方法：经典模型（CM），即通过以下方式训练的 $\hat{}p(\boldsymbol{s}'|\boldsymbol{s},\boldsymbol{a})$：(i) 随机动作（“Random”）；(ii) 基于集成分歧的探索（“Disagreement”）（Pathak等人，2019），此前有多项研究将其作为模型学习的无监督数据收集方案（Shyam等人，2019；Sekar 等，2020；Rajeswar等人，2022）；以及 (iii) 随机网络蒸馏（“RND”）（Burda等人，2019），这是Rajeswar等人（2022）考察的另一种数据收集方法。我们还将 PMA 与 DADS（Sharma 等人，2020）进行比较。DADS 是一种早期的无监督技能发现方法，它也会学习潜在动作动力学模型 $\hat{}p_z(\boldsymbol{s}'|\boldsymbol{s},\boldsymbol{z})$，但其目标是发现紧凑且在时间上延展的行为，而不是将原始MDP转化为更易预测的形式。在基准测试中，我们在七个 MuJoCo 机器人环境（Todorov等人，2012；Brockman 等，2016）中的 13 个不同任务上测试 PMA 和四种以往的方法。值得注意的是，在实验中，我们始终为每种方法、任务和控制器分别调整集成分歧惩罚项（MOPO 惩罚项，第5节 5.2节），以确保比较公平。每项实验均使用 8 个随机种子运行，并在图中展示 95% 置信区间。

7.1. 使用 PMA 进行基于模型的规划

为了考察 PMA 是否能够带来更好的规划性能，我们首先使用零样本规划来完成下游任务，评估由 PMA 和各类先前方法学习得到的模型。PMA 和 DADS 使用潜在模型 $\hat{}p_z(\boldsymbol{s}'|\boldsymbol{s},\boldsymbol{z})$，而其他方法均使用形式为 $\hat{}p(\boldsymbol{s}'|\boldsymbol{s},\boldsymbol{a})$ 的“经典”模型，彼此的差异仅在于无监督数据收集策略。我们在七个 MuJoCo 环境（HalfCheetah、Ant、Hopper、Walker2d、InvertedPendulum（“IP”）、InvertedDoublePendulum（“IDP”）和 Reacher）中的 13 个任务上进行比较。在这些方法的无监督训练过程中，我们定期在下游任务上结合 MOPO 惩罚项运行 MPPI 规划（第 5.1 节）（这些试验不用于模型训练；模型训练过程完全不涉及任务奖励），并在图 5 中报告结果。结果表明，PMA 在大多数任务中表现最佳。尤其是，在 Ant 环境中，PMA 是唯一成功的无监督模型式方法。由于 Ant 的动力学复杂且涉及丰富接触，经典模型难以成功，因为模型预测一旦出错，智能体往往会翻倒。PMA 则成功解决了这些任务，因为我们提出的可预测抽象能够有效避免此类错误。

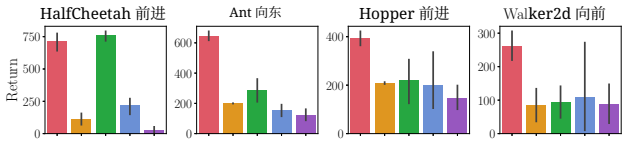


图 6. 使用结合 MOPO 惩罚项的 MBPO 对无监督基于模型的强化学习方法进行比较。PMA 大多优于以往方法，且通常具有较大优势。

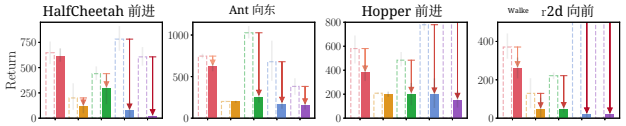
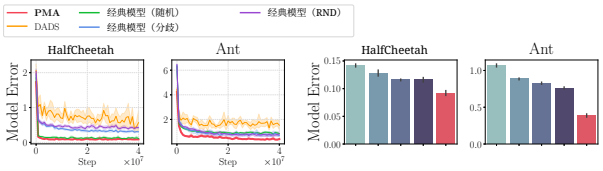


图 7. SAC 与 MOPO 惩罚项结合后，在纯基于模型的全长度轨迹展开上训练所得的性能。虚线框表示预测回报，实线框表示真实回报。在这种长时域设置下，经典模型会遭受模型利用问题，表现为预测回报与实际回报之间的差距；而 PMA 的预测回报降幅较小，性能显著更好。



(a) 不同方法的模型误差 (b) 不同网络规模的模型误差

图 8。 (a) 由于采用了我们的可预测性目标，PMA 的模型误差最低（视频）。(b) 即使在确定性环境中，也很难彻底降低经典模型中的误差
即使使用更大的神经网络，模型也是如此。

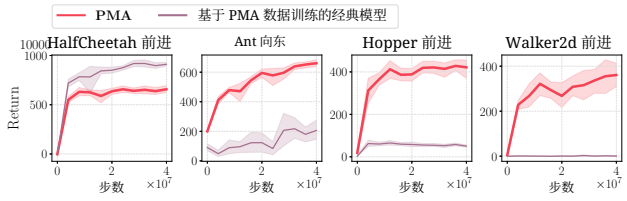


图 9。 MPPI 对比 PMA 与使用 PMA 收集的数据训练的经典模型的性能。尽管 HalfCheetah 无法从动作重参数化中受益，但在其他环境中，拥有独立的潜在动作空间至关重要。

7.2. 使用 PMA 的基于模型的强化学习

上一节采用的规划方法使用学习得到的模型进行短时域模型预测控制（MPC）。而通过将这些模型用于长时域无模型强化学习流程（本质上是把模型当作“模拟器”），我们可以更好地评估 PMA 模型和基线模型进行准确长时域预测的能力。本节研究两种方法。第一种基于 MBPO（Janner 等人，2019年），具体介绍见第 5.1 节。第二种是在全长度模型式轨迹展开之上使用 SAC（Haarnoja 等人，2018a），从某种意义上说，这是将学习得到的模型用作真实环境“模拟器”最直接的方式。第二种方法需要显著更长的模型式轨迹展开（最长可达 200 个时间步）。尽管它在实践中的表现较差，却能更严格地检验模型进行长时域预测而不产生过度误差累积的能力。在这两种评估方案中，我们都使用 MOPO 惩罚项来防止分布偏移。

图 6 和图 7分别展示了 MBPO 和 SAC 的结果。在这两种设置中，PMA 的表现大多优于以往基于模型的方法，这表明 PMA 能够进行可靠的长时域预测。此外，通过比较图 5 和图 6中 Hopper 和 Walker2d 的表现，我们发现，经典模型无法与 MPPI 配合使用，必须借助 MBPO 这类复杂控制器才能取得成功；而 PMA 凭借其可预测的动力学，使用简单的规划器便能达到与 MBPO 相近的结果。在图7的完整长度 SAC 曲线中，我们比较了模型的预测回报与实际回报。我们发现，PMA 的预测回报与实际回报之间的下降通常并不明显，这表明我们的可预测抽象能够有效防止模型利用。DADS 同样表现出较小的性能差异，因为它也限制了动作；但由于其对状态-动作空间的覆盖范围有限，DADS 的绝对性能明显落后于 PMA。另一方面，经典模型往往会因复杂的动力学而对回报产生错误的乐观估计，从而遭受模型利用。

7.3. 分析

能否仅通过增大模型规模来减少模型误差? 我们首先比较了五种方法在 HalfCheetah 和 Ant 中对状态（归一化后）的预测均方误差，并在图8a 中报告了结果。在这两种环境中，PMA 的模型误差最低，因为它的训练目标是尽可能提高可预测性。为考察仅通过增大经典模型$\hat{}p(\boldsymbol{s}'|\boldsymbol{s},\boldsymbol{a})$的规模能否达到如此低的误差，我们使用随机动作训练了四种不同规模的经典模型，规模从两个256单元的隐藏层到两个2048单元的隐藏层不等。图8b 展示了结果，这表明即使在确定性环境中，由于其动力学复杂且涉及大量接触，模型误差实际上也无法完全消除。另一方面，PMA 并不试图对整个马尔可夫决策过程建模，而只对动作空间中“可预测”的部分建模，从而减少模型误差，使基于模型的学习得以应用。

数据限制与动作重参数化。PMA 同时实现了（*i*）通过不选择不可预测的动作来限制数据，以及（*ii*）通过设置独立的潜在动作空间来进行动作重参数化。为剖析这两种作用，我们还考虑了一个使用与训练 PMA 相同数据进行训练的经典模型$\hat{}p(\boldsymbol{s}'|\boldsymbol{s},\boldsymbol{a})$。我们在图9中比较了该设置与 PMA 的周期性 MPPI 性能。结果表明，尽管在 HalfCheetah 中，数据限制结合 MOPO 惩罚项已足够，但在其他具有提前终止或不可逆状态的“不稳定”环境中（例如，Ant 翻倒），设置独立的潜在动作空间至关重要。这是因为，测试时的 MOPO 惩罚项只能防止短期偏离数据分布，而通过强化学习 训练的 PMA 会考虑长期可预测性，从而有效避免选择那些看似无害、最终却可能导致智能体失去平衡的动作（这对应于不可预测的行为）。

我们请参阅我们的项目页面以及附录A，了解定性结果；另请参阅附录B，了解消融研究。

8. 结论

我们提出了可预测MDP抽象（PMA），这是一种无监督的基于模型的方法，通过重新参数化动作空间，仅允许可预测动作，从而构建潜在MDP。我们运用信息论对其目标进行了形式化，并从理论上分析了有损训练方案导致的次优性。通过实验，我们证实PMA能够实现稳健的基于模型的学习，其性能显著优于先前的方法。

局限性。 PMA 的一个局限是它是一种有损过程。尽管我们通过实验表明，在我们的实验中，PMA 提高的可预测性超过了限制动作空间所带来的局限，但对于那些需要不可预测或高度复杂行为的任务（例如第6.2 节讨论的任务），PMA 可能不是最优选择。一般来说，也很难保证 PMA 学得的抽象对每个下游任务都适用（不过，任何方法都可能难以提供此类保证）。此外，PMA 需要调整系数 β，以平衡可预测性与状态-动作空间覆盖率。尽管如此，我们相信，专门针对马尔可夫决策过程中的可预测部分进行建模的方法，对未来基于模型的强化学习方法大有裨益。未来的工作可以探索将这类技术与无模型方法结合，以处理“不可预测”部分；进一步研究有效的数据收集策略，或能够利用先前收集的离线数据的方法；并探讨如何将这类方法扩展至更复杂、维度更高的观测空间。在此类空间中，直接针对可预测性进行训练或许能带来更显著的性能提升。

致谢

我们感谢 Benjamin Eysenbach 就最初的想法进行了富有启发性的讨论，感谢 Mitsuhiko Nakamoto、Jaekyeom Kim 和 YoungwoonLee 就实现与展示进行了讨论，也感谢 RAIL 成员和匿名审稿人提出的宝贵意见。Seohong Park 获得了韩国高等研究基金会（KFAS）的部分资助。本研究使用了加州大学伯克利分校伯克利研究计算项目提供的 Savio 计算集群资源，并获得美国空军科学研究办公室 AFOSR FA9550-22-1-0273 和美国海军研究办公室的部分资助。

参考文献

Ajay, A., Kumar, A., Agrawal, P., Levine, S., and Nachum, O. Opal：用于加速离线强化学习的离线原语发现。载于国际学习表征会议（*ICLR*），2021年。Argenson, A. and Dulac-Arnold, G. 基于模型的离线规划。载于 国际学习表征会议（*ICLR*），2021年。Bacon, P.-L., Harb, J., and Precup, D. Option-Critic 架构。载于 *AAAI*人工智能会议（*AAAI*），2017年。Ball, P. J., Parker-Holder, J., Pacchiano, A., Choromanski, K., and Roberts, S. J. Ready policy one：通过主动学习构建世界。载于 国际机器学习会议（*ICML*），2020年。Barber, D. and Agakov, F. IM算法：一种用于信息最大化的变分方法。载于 神经信息处理系统大会（*NeurIPS*），2003年。Berseth, G., Geng, D., Devin, C., Rhinehart, N., Finn, C., Jayaraman, D., and Levine, S. SMiRL：在不稳定环境中最小化意外的强化学习。载于国际学习表征会议（*ICLR*），2021年。Brockman, G., Cheung, V., Pettersson, L., Schneider, J., Schulman, J., Tang, J., and Zaremba, W. OpenAI Gym。*arXiv*, abs/1606.01540，2016年。Buckman, J., Hafner, D., Tucker, G., Brevdo, E., and Lee, H. 通过随机集成价值扩展实现样本高效的强化学习。载于 神经信息处理系统大会（*NeurIPS*），2018年。

Burda, Y.、Edwards, H.、Storkey, A. J. 和 Klimov, O. 基于随机网络蒸馏的探索。载于国际学习表征会议（*ICLR*），2019年。

Campos Cam´u˜nez, V.、Trott, A.、Xiong, C.、Socher, R.、Gir´o Nieto, X. 和 Torres Vi˜nals, J. 探索、发现与学习：无监督发现状态覆盖技能。载于国际机器学习会议（*ICML*），2020年。

Castro, P. S. 在确定性马尔可夫决策过程中计算状态相似度的可扩展方法。载于*AAAI*人工智能会议（*AAAI*），2019年。

Chua, K., Calandra, R., McAllister, R., and Levine, S. 使用概率动力学模型，通过少量试验实现深度强化学习。载于神经信息处理系统大会（*NeurIPS*），2018年。

Clavera, I., Rothfuss, J., Schulman, J., Fujita, Y., Asfour, T., and Abbeel, P. 通过元策略优化实现基于模型的强化学习。载于机器人学习会议（*CoRL*），2018年。Deisenroth, M. P. and Rasmussen, C. E. Pilco：一种基于模型且数据高效的策略搜索方法。载于国际机器学习会议（*ICML*），2011年。Draeger, A., Engell, S., and Ranke, H. D. 使用神经网络的模型预测控制。*IEEE*控制系统杂志，15:61–66，1995年。Ebert, F., Finn, C., Dasari, S., Xie, A., Lee, A. X., and Levine, S. 视觉前瞻：用于基于视觉的机器人控制的基于模型的深度强化学习。*arXiv*，abs/1812.00568，2018年。Eysenbach, B., Gupta, A., Ibarz, J., and Levine, S. 多样性就是你所需要的一切：无需奖励函数的技能学习。载于国际学习表征会议（*ICLR*），2019年。Eysenbach, B., Salakhutdinov, R., and Levine, S. 鲁棒可预测控制。载于神经信息处理系统大会（*NeurIPS*），2021年。Feinberg, V., Wan, A., Stoica, I., Jordan, M. I., Gonzalez, J. E., and Levine, S. 基于模型的价值扩展：实现高效无模型强化学习。载于国际机器学习会议（*ICML*），2018年。Gelada, C., Kumar, S., Buckman, J., Nachum, O., and Belle- mare, M. G. DeepMDP：学习用于表征学习的连续潜在空间模型。载于国际机器学习会议（*ICML*），2019年。Gregor, K., Rezende, D. J., and Wierstra, D. 变分内在控制。*arXiv*，abs/1611.07507，2016年。Ha, D. and Schmidhuber, J. 循环世界模型助力策略演化。载于神经信息处理系统大会（*NeurIPS*），2018年。Haarnoja, T., Zhou, A., Abbeel, P., and Levine, S. 软演员-评论家：利用随机演员进行离策略最大熵深度强化学习。发表于国际机器学习会议（*ICML*），2018a。Haarnoja, T., Zhou, A., Hartikainen, K., Tucker, G., Ha, S., Tan, J., Kumar, V., Zhu, H., Gupta, A., Abbeel, P., and Levine, S. 软演员-评论家算法及其应用。*arXiv*，abs/1812.05905，2018b。Hafner, D., Lillicrap, T. P., Fischer, I. S., Villegas, R., Ha, D. R., Lee, H., and Davidson, J. 学习潜在动力学，以便基于像素进行规划。发表于国际机器学习会议（*ICML*），2019。Hafner, D., Lillicrap, T. P., Ba, J., and Norouzi, M. 梦境控制：通过潜在想象学习行为。发表于国际学习表征会议（*ICLR*），2020。Hansen, N., Wang, X., and Su, H. 用于模型预测控制的时序差分学习。发表于国际机器学习会议（*ICML*），2022。Hasselt, H. V., Hessel, M., and Aslanides, J. 强化学习中何时使用参数化模型？发表于神经信息处理系统大会（*NeurIPS*），2019。Heess, N. M. O., Wayne, G., Silver, D., Lillicrap, T. P., Erez, T., a nd Tassa, Y. 通过随机价值梯度学习连续控制策略。发表于神经信息处理系统大会（*NeurIPS*），2015。Hernandaz, E. P. S. and Arkun, Y. 用于控制非线性系统的神经网络建模与扩展DMC算法。发表于美国控制会议，1990。Houthooft, R., Chen, X., Duan, Y., Schulman, J., Turck, F. D., and Abbeel, P. VIME：变分信息最大化探索。发表于神经信息处理系统大会（*NeurIPS*），2016。Jafferjee, T., Imani, E., Talvitie, E. J., White, M., and Bowling, M. 幻觉价值：使用不完善环境模型进行Dyna式规划的一个陷阱。*arXiv*，abs/2006.04363，2020。Janner, M., Fu, J., Zhang, M., and Levine, S. 何时信任你的模型：基于模型的策略优化。发表于神经信息处理系统大会（*NeurIPS*），2019。Kang, K., Gradu, P., Choi, J. J., Janner, M., Tomlin, C. J., and Levine, S. 李雅普诺夫密度模型：约束基于学习的控制中的分布偏移。发表于国际机器学习会议（*ICML*），2022。Kidambi, R., Rajeswaran, A., Netrapalli, P., and Joachims, T. MOREL：基于模型的离线强化学习。发表于神经信息处理系统大会（*NeurIPS*），2020。

Kim, J., Park, S., and Kim, G. 通过瓶颈选项学习进行无监督技能发现。发表于国际机器学习会议（*ICML*），2021。

Kingma, D. P. 和 Ba, J. Adam：一种随机优化方法。发表于国际学习表征会议（*ICLR*），2015年。

Kurutach, T.、Clavera, I.、Duan, Y.、Tamar, A. 和 Abbeel, P. 模型集成信赖域策略优化。发表于国际学习表征会议（*ICLR*），2018年。

Laskin, M.、Liu, H.、Peng, X. B.、Yarats, D.、Rajeswaran, A. 和 Abbeel, P. 结合对比内在控制的无监督强化学习。发表于神经信息处理系统大会（*NeurIPS*），2022年。

Lee, L.、Eysenbach, B.、Parisotto, E.、Xing, E. P.、Levine, S. 和 Salakhutdinov, R. 通过状态边缘分布匹配实现高效探索。*arXiv*，abs/1906.05274，2019年。

Lenz, I.、Knepper, R. A. 和 Saxena, A. DeepMPC：学习用于模型预测控制的深层潜在特征。载于机器人：科学与系统（*RSS*），2006年。

Li, L.、Walsh, T. J. 和 Littman, M. L. 面向马尔可夫决策过程状态抽象的统一理论。载于国际人工智能与数学研讨会，2006年。Liu, H. 和 Abbeel, P. 虚无中的行为：无监督主动预训练。载于 神经信息处理系统（*NeurIPS*），2021年。Lu, K.、Grover, A.、Abbeel, P. 和 Mordatch, I. 通过技能空间规划实现无重置终身学习。载于 国际学习表征会议（*ICLR*），2021年。

Machado, M. C.、Bellemare, M. G. 和 Bowling, M. 用于强化学习中选项发现的拉普拉斯框架。载于国际机器学习会议（*ICML*），2017年。

Mendonca, R., Rybkin, O., Daniilidis, K., Hafner, D., 和 Pathak, D. 通过世界模型发现并实现目标。载于神经信息处理系统大会 *(NeurIPS)*，2021年。

Nachum, O., Gu, S. S., Lee, H., 和 Levine, S. 数据高效的分层强化学习。载于神经信息处理系统大会 *(NeurIPS)*，2018年。

Nachum, O., Gu, S. S., Lee, H. 和 Levine, S. 层次化强化学习的近似最优表征学习。载于 国际学习表征会议（*ICLR*），2019年。

Nagabandi, A., Konolige, K., Levine, S. 和 Kumar, V. 用于灵巧操作学习的深度动力学模型。载于机器人学习会议（*CoRL*），2019年。Nguyen, T. D., Shu, R., Pham, T., Bui, H. H. 和 Ermon, S. 潜在空间中用于基于模型的规划的时间预测编码。载于 国际机器学习会议（*ICML*），2021年。Park, S., Choi, J., Kim, J., Lee, H. 和 Kim, G. 受 Lipschitz 约束的无监督技能发现。载于国际学习表征会议（*ICLR*），2022年。

Pathak, D., Agrawal, P., Efros, A. A., 和 Darrell, T. 通过自监督预测进行好奇心驱动的探索。载于国际机器学习会议 *(ICML)*，2017年。

Pathak, D., Gandhi, D., 和 Gupta, A. K. 通过分歧进行自监督探索。载于国际机器学习会议 *(ICML)*，2019年。

Pong, V. H., Dalal, M., Lin, S., Nair, A., Bahl, S., 和 Levine, S. Skew-Fit：覆盖状态的自监督强化学习。载于国际机器学习会议 *(ICML)*，2020年。

Poole, B.、Ozair, S.、van den Oord, A.、Alemi, A. A. 和 Tucker, G. 关于互信息的变分界。载于 国际机器学习会议（*ICML*），2019年。

Rajeswar, S.、Mazzaglia, P.、Verbelen, T.、Pich'e, A.、Dhoedt, B.、Courville, A.C. 和 Lacoste, A. 基于无监督模型的预训练：利用像素实现数据高效控制。*arXiv*，abs/2209.12016，2022年。

Rajeswaran, A., Ghotra, S., Levine, S., 和 Ravindran, B. Epopt：利用模型集成学习鲁棒神经网络策略。载于国际表征学习会议（*ICLR*），2017年。Rasmussen, C. E. 和 Kuss, M. 强化学习中的高斯过程。载于神经信息处理系统大会（*NeurIPS*），2003年。Rhinehart, N., Wang, J., Berseth, G., Co-Reyes, J. D., Hafner, D., Finn, C., 和 Levine, S. 信息即力量：通过信息捕获实现内在控制。载于神经信息处理系统大会（*NeurIPS*），2021年。Ross, S. 和 Bagnell, J. A. 基于模型的强化学习中的无先验系统辨识。载于国际机器学习会议（*ICML*），2012年。Salter, S., Wulfmeier, M., Tirumala, D., Heess, N. M. O., Riedmiller, M. A., Hadsell, R., 和 Rao, D. Mo2：基于模型的离线选项。载于终身学习智能体会议（*CoLLAs*），2022年。

Schrittwieser, J., Antonoglou, I., Hubert, T., Simonyan, K., Sifre, L., Schmitt, S., Guez, A., Lockhart, E., Hassabis, D., Graepel, T., Lillicrap, T., 和 Silver, D. 通过使用学习得到的模型进行规划，精通雅达利、围棋、国际象棋和将棋。《自然》，588 7839:604–609，2020年。

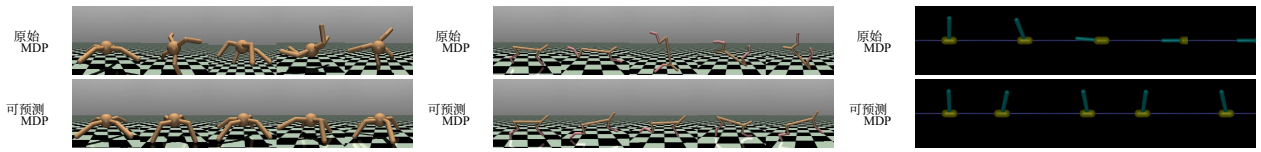
Sekar, R., Rybkin, O., Daniilidis, K., Abbeel, P., Hafner, D., and Pathak, D. 通过自监督世界模型进行探索规划。载于国际机器学习会议 *(ICML)*，2020年。

Sharma, A., Gu, S., Levine, S., Kumar, V., 和 Hausman, K. 动力学感知的无监督技能发现。发表于国际学习表征会议 *(ICLR)*，2020年。Shyam, P., Ja´ skowski, W., 和 Gomez, F. J. 基于模型的主动探索。发表于国际机器学习会议 *(ICML)*，2019年。Sikchi, H. S., Zhou, W., 和 Held, D. 通过在线规划学习离策略策略。发表于机器人学习会议 *(CoRL)*，2022年。Stolle, M. 和 Precup, D. 在强化学习中学习选项。发表于抽象、重构与近似研讨会，2002年。Strouse, D., Baumli, K., Warde-Farley, D., Mnih, V., 和 Hansen, S. S. 通过乐观探索学习更多技能。发表于国际学习表征会议 *(ICLR)*，2022年。Sutton, R. S. Dyna：用于学习、规划和反应的集成架构。*ACM Sigart Bulletin*，2(4):160–163，1991年。Sutton, R. S., Precup, D., 和 Singh, S. 从马尔可夫决策过程到半马尔可夫决策过程：强化学习中时间抽象的框架。人工智能，112(1-2):181–211，1999年。Testud, J., Richalet, J., Rault, A., 和 Papon, J. 模型预测启发式控制：在工业过程中的应用。《自动化》，14(5):413–428，1978年。Todorov, E., Erez, T., 和 Tassa, Y. MuJoCo：用于基于模型的控制的物理引擎。发表于*IEEE/RSJ*智能机器人与系统国际会议 *(IROS)*，2012年。Vezhnevets, A. S., Osindero, S., Schaul, T., Heess, N. M. O., Jaderberg, M., Silver, D., 和 Kavukcuoglu, K. 用于分层强化学习的封建网络。发表于国际机器学习会议 *(ICML)*，2017年。

Watter, M., Springenberg, J. T., Boedecker, J., 和 Riedmiller, M. A. 嵌入以实现控制：一种用于从原始图像进行控制的局部线性潜在动力学模型。发表于神经信息处理系统大会 *(NeurIPS)*，2015年。

Williams, G., Drews, P., Goldfain, B., Rehg, J. M., and Theodorou, E. A. 使用基于模型的预测路径积分控制实现激进驾驶。载于*IEEE* 国际机器人与自动化会议 *(ICRA)*，2016年。

Wu, P., Escontrela, A., Hafner, D., Goldberg, K., and Abbeel, P. Daydreamer：用于实体机器人学习的世界模型。载于机器人学习会议（*CoRL*），2022年。Wulfmeier, M., Rao, D., Hafner, R., Lampe, T., Abdol- maleki, A., Hertweck, T., Neunert, M., Tirumala, D., Siegel, N., Heess, N. M. O., and Riedmiller, M. A. 高效利用事后信息进行离策略选项学习。载于国际机器学习会议（*ICML*），2021年。Xie, K., Bharadhwaj, H., Hafner, D., Garg, A., and Shkurti, F. 用于探索与迁移的潜在技能规划。载于国际学习表征会议（*ICLR*），2020年。Yarats, D., Fergus, R., Lazaric, A., and Pinto, L. 基于原型表示的强化学习。载于国际机器学习会议（*ICML*），2021年。Yu, T., Thomas, G., Yu, L., Ermon, S., Zou, J. Y., Levine, S., Finn, C., and Ma, T. MOPO：基于模型的离线策略优化。载于神经信息处理系统大会（*NeurIPS*），2020年。



(a) Ant (b) HalfCheetah (c) 倒立摆

图 *10.* PMA 示例。PMA 可防止不可预测的混乱动作，使潜在MDP中的每次状态转移都具有最大的可预测性。视频请访问 https://seohong.me/projects/pma/。

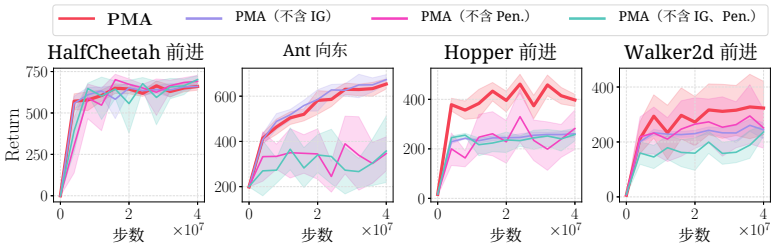


图 *11.*消融研究：无监督训练期间的分歧奖励（“IG”）和规划期间的分歧惩罚（“Pen.”）。分歧惩罚通常能稳定训练，而分歧奖励则能提升性能。

A. PMA 示例

为说明原始马尔可夫决策过程与相应可预测马尔可夫决策过程之间的区别，我们在图 10中展示了 PMA 的定性示例。在 Ant 和 HalfCheetah 中，我们的可预测马尔可夫决策过程只允许采用易于建模、同时又足够多样化以解决下游任务的动作，从而避免混乱翻转等不可预测行为。在倒立摆任务中，我们发现，即使没有提前终止，大多数学到的潜在动作也会在保持平衡的同时让智能体朝不同方向移动，以使状态转移尽可能可预测。视频请见https://seohong.me/projects/pma/。

B. 消融研究

为评估 PMA 各组成部分的相对重要性，我们在无监督训练期间消融信息增益（分歧奖励）项，并在定期进行 MPPI 规划时消融分歧惩罚项；实验表现如图所示11。虽然 HalfCheetah 中不同设置的性能差异较小，但这些组成部分通过鼓励探索和防止分布偏移，提升并稳定了其他更复杂环境中的表现。

C. 用集成分歧近似信息增益

本节将说明为何使用集成分歧来近似信息增益项 $I(\boldsymbol{S}';\ \boldsymbol{\Theta}|\boldsymbol{\mathcal{D}}\ \boldsymbol{Z})$，该项见公式（6）。首先，令随机变量 $\hat{}\boldsymbol{S}' \sim \hat{}p_z(\cdot|\boldsymbol{S},\ \boldsymbol{Z};\ \boldsymbol{\Theta})$ 表示参数为 $\boldsymbol{\Theta}$的模型所预测的状态。由于 $\boldsymbol{S}' \to \boldsymbol{\Theta} \to \hat{}\boldsymbol{S}'$ 在给定$\boldsymbol{\mathcal{D}}$ 和 $\boldsymbol{Z}$ 的贝叶斯设定下构成马尔可夫链，根据数据处理不等式可得以下下界：

$$I(\boldsymbol{S}';\boldsymbol{\Theta}|\boldsymbol{\mathcal{D}},\boldsymbol{Z}) \geq I(\boldsymbol{S}';\hat{\boldsymbol{S}}'|\boldsymbol{\mathcal{D}},\boldsymbol{Z}) \tag{14}$$

$$= H(\hat{\boldsymbol{S}}'|\boldsymbol{\mathcal{D}},\boldsymbol{Z}) - H(\hat{\boldsymbol{S}}'|\boldsymbol{\mathcal{D}},\boldsymbol{Z},\boldsymbol{S}'). \tag{15}$$

现在，我们用一个由 E 个预测模型组成的集成来近似模型后验，$\{\hat{}p_z(\boldsymbol{s}'|\boldsymbol{s},\boldsymbol{z};\boldsymbol{\theta}_i)\}_{i\in[E]}$ with $p(\boldsymbol{\theta}|\mathcal{D}) = \frac{1}{E}\sum_i \delta(\boldsymbol{\theta}-\boldsymbol{\theta}_i)$，其中每个模型都表示为条件高斯分布，其均值由神经网络给出，协方差为单位对角协方差，$\boldsymbol{s}' \sim \mathcal{N}(\boldsymbol{\mu}(\boldsymbol{s},\boldsymbol{z};\boldsymbol{\theta}_i)\ \boldsymbol{I})$。公式（15）中的各项分别衡量观测结果 $\boldsymbol{S}'$ 之前和之后对下一状态预测的不确定性。然而，这仍然难以处理，因为高斯混合分布的微分熵没有封闭形式。因此，我们进一步对这些项作如下简化。首先，我们假设公式（15）中的第二项对目标的影响可以忽略不计，这大致相当于假设训练误差较低（即，如果我们知道 $\boldsymbol{S}'$ 的取值并更新模型，它们应当对预测结果达成一致

于 $\hat{\boldsymbol{S}}'$，或者至少具有相似的误差）。这一假设在异方差环境中可能不成立，但除此之外非常方便。接下来，我们根据二者都与 $\boldsymbol{S}'$ 中的不确定性相关这一事实，采用系数为 $\beta, \mathbb{E}[\beta \cdot \mathrm{Tr}[\mathbb{V}_i[\boldsymbol{\mu}(\boldsymbol{s}, \boldsymbol{z}; \boldsymbol{\theta}_i)]]]$, 的集成均值方差，对公式（15）中的第一项进行经验性替代（Sekar等人，2020）：如果集成模型的预测彼此差异很大（即方差较大），那么 $\boldsymbol{S}'$ 的边缘熵也会较大，反之亦然。由此，我们得到公式（10）中的近似。关于集成分歧与信息增益之间的类似联系，也可参见先前研究（Shyam等人，2019；Sekar等人，2020；Ball等人，2020；Strouse等人，2022）。

D. 理论结果

D.1. 技术引理

对于一个马尔可夫决策过程 $\mathcal{M} := (\mathcal{S}, \mathcal{A}, \mu, p, r)$[3] 和一个策略 π，我们首先定义折扣状态分布和折扣状态-动作分布，并陈述贝尔曼流约束引理。

定义 D.1.（折扣状态分布） $d^\pi(\boldsymbol{s}) := (1-\gamma) \sum_{t=0}^{\infty} \gamma^t P(\boldsymbol{s}_t = \boldsymbol{s} | \mu, p, \pi)$.

定义 D.2.（折扣状态-动作分布） $d^\pi(\boldsymbol{s}, \boldsymbol{a}) := \pi(\boldsymbol{a}|\boldsymbol{s}) d^\pi(\boldsymbol{s})$。

引理 D.3.（贝尔曼流约束）

$$d^\pi(\boldsymbol{s}) = (1-\gamma)\mu(\boldsymbol{s}) + \gamma \sum_{\boldsymbol{s}^- \in \mathcal{S}, \boldsymbol{a}^- \in \mathcal{A}} p(\boldsymbol{s}|\boldsymbol{s}^-, \boldsymbol{a}^-) d^\pi(\boldsymbol{s}^-, \boldsymbol{a}^-). \tag{16}$$

证明。

$$\begin{aligned}
d^\pi(\boldsymbol{s}) &= (1-\gamma)(P(\boldsymbol{s}_0 = \boldsymbol{s}) + \gamma P(\boldsymbol{s}_1 = \boldsymbol{s}) + \gamma^2 P(\boldsymbol{s}_2 = \boldsymbol{s}) + \cdots) && (17)\\
&= (1-\gamma)\mu(\boldsymbol{s}) + \gamma(1-\gamma)(P(\boldsymbol{s}_1 = \boldsymbol{s}) + \gamma P(\boldsymbol{s}_2 = \boldsymbol{s}) + \cdots) && (18)\\
&= (1-\gamma)\mu(\boldsymbol{s}) + \gamma(1-\gamma) \sum_{\boldsymbol{s}^-, \boldsymbol{a}^-} p(\boldsymbol{s}|\boldsymbol{s}^-, \boldsymbol{a}^-)(P(\boldsymbol{s}_0 = \boldsymbol{s}^-, \boldsymbol{a}_0 = \boldsymbol{a}^-) + \gamma P(\boldsymbol{s}_1 = \boldsymbol{s}^-, \boldsymbol{a}_1 = \boldsymbol{a}^-) + \cdots) && (19)\\
&= (1-\gamma)\mu(\boldsymbol{s}) + \gamma \sum_{\boldsymbol{s}^-, \boldsymbol{a}^-} p(\boldsymbol{s}|\boldsymbol{s}^-, \boldsymbol{a}^-) d^\pi(\boldsymbol{s}^-, \boldsymbol{a}^-)。 && (20)
\end{aligned}$$

□

现在，我们考虑两个具有不同转移动态的马尔可夫决策过程，$\mathcal{M}_1 := (\mathcal{S}, \mathcal{A}, r, \mu, p_1)$ 和 $\mathcal{M}_2 := (\mathcal{S}, \mathcal{A}, r, \mu, p_2)$，以及两个策略 π_1 和 π_2。我们将 π 的期望回报记为 $J_{\mathcal{M}}(\pi) := \frac{1}{1-\gamma} \mathbb{E}_{(\boldsymbol{s}, \boldsymbol{a}) \sim d^\pi(\boldsymbol{s}, \boldsymbol{a})}[r(\boldsymbol{s}, \boldsymbol{a})]$，并将最大奖励记为 $R := \max_{\boldsymbol{s} \in \mathcal{S}, \boldsymbol{a} \in \mathcal{A}} r(\boldsymbol{s}, \boldsymbol{a})$。它们的性能差异可以界定如下。

引理 D.4. 若动力学和策略的全变差距离满足以下界

$$\mathbb{E}_{(\boldsymbol{s}, \boldsymbol{a}) \sim d_1^{\pi_1}(\boldsymbol{s}, \boldsymbol{a})}[D_{\mathrm{TV}}(p_1(\cdot|\boldsymbol{s}, \boldsymbol{a}) \| p_2(\cdot|\boldsymbol{s}, \boldsymbol{a}))] \leq \epsilon_m, \tag{21}$$

$$\mathbb{E}_{\boldsymbol{s} \sim d_1^{\pi_1}(\boldsymbol{s})}[D_{\mathrm{TV}}(\pi_1(\cdot|\boldsymbol{s}) \| \pi_2(\cdot|\boldsymbol{s}))] \leq \epsilon_\pi, \tag{22}$$

则它们的性能差异满足以下不等式：

$$|J_{\mathcal{M}_1}(\pi_1) - J_{\mathcal{M}_2}(\pi_2)| \leq \frac{R}{(1-\gamma)^2}(2\gamma\epsilon_m + 2\epsilon_\pi). \tag{23}$$

[3]为简化分析，我们在理论分析中假设状态空间和动作空间均为有限集。

证明。 我们首先对它们的折扣状态-动作分布之间的差异进行界定。

$$\sum_{\boldsymbol{s}\in\mathcal{S},\boldsymbol{a}\in\mathcal{A}} |d_1^{\pi_1}(\boldsymbol{s},\boldsymbol{a}) - d_2^{\pi_2}(\boldsymbol{s},\boldsymbol{a})| \tag{24}$$

$$= \sum_{\boldsymbol{s}\in\mathcal{S},\boldsymbol{a}\in\mathcal{A}} |\pi_1(\boldsymbol{a}|\boldsymbol{s})d_1^{\pi_1}(\boldsymbol{s}) - \pi_2(\boldsymbol{a}|\boldsymbol{s})d_2^{\pi_2}(\boldsymbol{s})| \tag{25}$$

$$\leq \sum_{\boldsymbol{s}\in\mathcal{S},\boldsymbol{a}\in\mathcal{A}} |\pi_1(\boldsymbol{a}|\boldsymbol{s})d_1^{\pi_1}(\boldsymbol{s}) - \pi_2(\boldsymbol{a}|\boldsymbol{s})d_1^{\pi_1}(\boldsymbol{s})| + \sum_{\boldsymbol{s}\in\mathcal{S},\boldsymbol{a}\in\mathcal{A}} |\pi_2(\boldsymbol{a}|\boldsymbol{s})d_1^{\pi_1}(\boldsymbol{s}) - \pi_2(\boldsymbol{a}|\boldsymbol{s})d_2^{\pi_2}(\boldsymbol{s})| \tag{26}$$

$$= \sum_{\boldsymbol{s}\in\mathcal{S}} d_1^{\pi_1}(\boldsymbol{s}) \sum_{\boldsymbol{a}\in\mathcal{A}} |\pi_1(\boldsymbol{a}|\boldsymbol{s}) - \pi_2(\boldsymbol{a}|\boldsymbol{s})| + \sum_{\boldsymbol{s}\in\mathcal{S}} |d_1^{\pi_1}(\boldsymbol{s}) - d_2^{\pi_2}(\boldsymbol{s})| \tag{27}$$

$$\leq \sum_{\boldsymbol{s}\in\mathcal{S}} |d_1^{\pi_1}(\boldsymbol{s}) - d_2^{\pi_2}(\boldsymbol{s})| + 2\epsilon_\pi \tag{28}$$

$$= \gamma \sum_{\boldsymbol{s}\in\mathcal{S}} \left| \sum_{\boldsymbol{s}^-\in\mathcal{S},\boldsymbol{a}^-\in\mathcal{A}} \left(p_1(\boldsymbol{s}|\boldsymbol{s}^-,\boldsymbol{a}^-)d_1^{\pi_1}(\boldsymbol{s}^-,\boldsymbol{a}^-) - p_2(\boldsymbol{s}|\boldsymbol{s}^-,\boldsymbol{a}^-)d_2^{\pi_2}(\boldsymbol{s}^-,\boldsymbol{a}^-)\right) \right| + 2\epsilon_\pi \tag{29}$$

$$\leq \gamma \sum_{\boldsymbol{s}^-\in\mathcal{S},\boldsymbol{a}^-\in\mathcal{A},\boldsymbol{s}\in\mathcal{S}} \left|p_1(\boldsymbol{s}|\boldsymbol{s}^-,\boldsymbol{a}^-)d_1^{\pi_1}(\boldsymbol{s}^-,\boldsymbol{a}^-) - p_2(\boldsymbol{s}|\boldsymbol{s}^-,\boldsymbol{a}^-)d_2^{\pi_2}(\boldsymbol{s}^-,\boldsymbol{a}^-)\right| + 2\epsilon_\pi \tag{30}$$

$$\leq \gamma \sum_{\boldsymbol{s}^-\in\mathcal{S},\boldsymbol{a}^-\in\mathcal{A},\boldsymbol{s}\in\mathcal{S}} \left|p_1(\boldsymbol{s}|\boldsymbol{s}^-,\boldsymbol{a}^-)d_1^{\pi_1}(\boldsymbol{s}^-,\boldsymbol{a}^-) - p_2(\boldsymbol{s}|\boldsymbol{s}^-,\boldsymbol{a}^-)d_1^{\pi_1}(\boldsymbol{s}^-,\boldsymbol{a}^-)\right|$$
$$+ \gamma \sum_{\boldsymbol{s}^-\in\mathcal{S},\boldsymbol{a}^-\in\mathcal{A},\boldsymbol{s}\in\mathcal{S}} \left|p_2(\boldsymbol{s}|\boldsymbol{s}^-,\boldsymbol{a}^-)d_1^{\pi_1}(\boldsymbol{s}^-,\boldsymbol{a}^-) - p_2(\boldsymbol{s}|\boldsymbol{s}^-,\boldsymbol{a}^-)d_2^{\pi_2}(\boldsymbol{s}^-,\boldsymbol{a}^-)\right| + 2\epsilon_\pi \tag{31}$$

$$= \gamma \sum_{\boldsymbol{s}^-\in\mathcal{S},\boldsymbol{a}^-\in\mathcal{A}} d_1^{\pi_1}(\boldsymbol{s}^-,\boldsymbol{a}^-) \sum_{\boldsymbol{s}\in\mathcal{S}} \left|p_1(\boldsymbol{s}|\boldsymbol{s}^-,\boldsymbol{a}^-) - p_2(\boldsymbol{s}|\boldsymbol{s}^-,\boldsymbol{a}^-)\right|$$
$$+ \gamma \sum_{\boldsymbol{s}^-\in\mathcal{S},\boldsymbol{a}^-\in\mathcal{A}} \left|d_1^{\pi_1}(\boldsymbol{s}^-,\boldsymbol{a}^-) - d_2^{\pi_2}(\boldsymbol{s}^-,\boldsymbol{a}^-)\right| + 2\epsilon_\pi \tag{32}$$

$$\leq 2\gamma\epsilon_m + 2\epsilon_\pi + \gamma \sum_{\boldsymbol{s}^-\in\mathcal{S},\boldsymbol{a}^-\in\mathcal{A}} \left|d_1^{\pi_1}(\boldsymbol{s}^-,\boldsymbol{a}^-) - d_2^{\pi_2}(\boldsymbol{s}^-,\boldsymbol{a}^-)\right| \tag{33}$$

$$= 2\gamma\epsilon_m + 2\epsilon_\pi + \gamma \sum_{\boldsymbol{s}\in\mathcal{S},\boldsymbol{a}\in\mathcal{A}} |d_1^{\pi_1}(\boldsymbol{s},\boldsymbol{a}) - d_2^{\pi_2}(\boldsymbol{s},\boldsymbol{a})|\,, \tag{34}$$

这意味着

$$\sum_{\boldsymbol{s}\in\mathcal{S},\boldsymbol{a}\in\mathcal{A}} |d_1^{\pi_1}(\boldsymbol{s},\boldsymbol{a}) - d_2^{\pi_2}(\boldsymbol{s},\boldsymbol{a})| \leq \frac{1}{1-\gamma}(2\gamma\epsilon_m + 2\epsilon_\pi), \tag{35}$$

其中，我们在公式（29）中使用引理 D.3。因此，我们得到

$$|J_{\mathcal{M}_1}(\pi_1) - J_{\mathcal{M}_2}(\pi_2)| = \frac{1}{1-\gamma} \left| \sum_{\boldsymbol{s}\in\mathcal{S},\boldsymbol{a}\in\mathcal{A}} (d_1^{\pi_1}(\boldsymbol{s},\boldsymbol{a}) - d_2^{\pi_2}(\boldsymbol{s},\boldsymbol{a}))r(\boldsymbol{s},\boldsymbol{a}) \right| \tag{36}$$

$$\leq \frac{R}{1-\gamma} \sum_{\boldsymbol{s}\in\mathcal{S},\boldsymbol{a}\in\mathcal{A}} |d_1^{\pi_1}(\boldsymbol{s},\boldsymbol{a}) - d_2^{\pi_2}(\boldsymbol{s},\boldsymbol{a})| \tag{37}$$

$$\leq \frac{R}{(1-\gamma)^2}(2\gamma\epsilon_m + 2\epsilon_\pi). \tag{38}$$

□

公式（23）与 Janner 等人（2019年）中的引理 B.3 给出的界相同，但我们在公式（22）中采用了较宽松的假设，该假设仅要求策略间全变差距离的期望（而非最大值）有界。

D.2. MBRL 性能界

我们首先给出原始MDP $\mathcal{M}=(\mathcal{S},\mathcal{A},\mu,p,r)$ 中策略 π 的性能界，以及其基于模型的 MDP（Janner 等，2019）的性能界。我们将基于学习得到的预测模型 $\hat{}p$ 的基于模型的 MDP 记为 $\hat{}\mathcal{M}=(\mathcal{S},\mathcal{A},\mu,\hat{}p,r)$。我们假设模型 $\hat{}p$ 是在数据集 $\mathcal{D}$ 上训练的，该数据集由数据收集策略 $\pi_{\mathcal{D}}$ 收集。

定理 D.5。 对于任意策略 π，如果以下全变差距离有界：*(i)* 真实动力学 p 与学习得到的模型 $\hat{}p$之间，以及 *(*ii*)* 策略 π 与数据收集策略 $\pi_{\mathcal{D}}$ 之间，则

$$\mathbb{E}_{(\boldsymbol{s},\boldsymbol{a})\sim d^{\pi_{\mathcal{D}}}(\boldsymbol{s},\boldsymbol{a})}[D_{\mathrm{TV}}(p(\cdot|\boldsymbol{s},\boldsymbol{a})\|\hat{p}(\cdot|\boldsymbol{s},\boldsymbol{a}))]\leq\epsilon_m, \tag{39}$$

$$\mathbb{E}_{\boldsymbol{s}\sim d^{\pi_{\mathcal{D}}}(\boldsymbol{s})}[D_{\mathrm{TV}}(\pi(\cdot|\boldsymbol{s})\|\pi_{\mathcal{D}}(\cdot|\boldsymbol{s}))]\leq\epsilon_\pi, \tag{40}$$

$\hat{}\pi$ 在 $\mathcal{M}$与 $\mathcal{M}$ 之间的性能差异满足以下不等式：

$$|J_{\mathcal{M}}(\pi)-J_{\hat{\mathcal{M}}}(\pi)|\leq\frac{R}{(1-\gamma)^2}(4\epsilon_\pi+2\gamma\epsilon_m). \tag{41}$$

证明。由引理D.4，我们得到

$$|J_{\mathcal{M}}(\pi)-J_{\hat{\mathcal{M}}}(\pi)|\leq|J_{\mathcal{M}}(\pi)-J_{\mathcal{M}}(\pi_{\mathcal{D}})|+|J_{\mathcal{M}}(\pi_{\mathcal{D}})+J_{\hat{\mathcal{M}}}(\pi)| \tag{42}$$

$$\leq\frac{R}{(1-\gamma)^2}(2\epsilon_\pi)+\frac{R}{(1-\gamma)^2}(2\epsilon_\pi+2\gamma\epsilon_m) \tag{43}$$

$$=\frac{R}{(1-\gamma)^2}(4\epsilon_\pi+2\gamma\epsilon_m). \tag{44}$$

□

D.3. PMA 性能界

我们给出了可预测MDP抽象的性能界。将可预测潜在MDP记为$\mathcal{M}_P:=(\mathcal{S},\ \mathcal{Z},\ \mu,\ p_z,\ r)$，其潜在动作空间为 $\mathcal{Z}$，奖励函数为 $r(\boldsymbol{s},\ \boldsymbol{z})=\sum_a\pi_z(\boldsymbol{a}|\boldsymbol{s},\ \boldsymbol{z})r(\boldsymbol{s},\ \boldsymbol{a})$；其基于模型的 MDP 记为$\hat{}\mathcal{M}_P:=(\mathcal{S},\ \mathcal{Z},\ \mu,\ \hat{}p_z,\ r)$。我们假设模型$\hat{}p_z$是在数据集 $\mathcal{D}_P$上训练的，该数据集由潜在MDP中的数据收集策略$\pi_{\mathcal{D}_P}$收集得到。

对于原始MDP与可预测的基于模型的MDP之间策略 $\pi(\boldsymbol{a}|\boldsymbol{s})$ 的性能界，我们分别处理由（i）MDP 抽象和（ii）模型学习所导致的性能损失。对于第一部分，我们采用与Nachum 等人(2019年)类似的方法；Ajay 等人(2021年)。对于任意 $\boldsymbol{s}\in\mathcal{S},\ \boldsymbol{a}\in\mathcal{A}$，以及概率分布$\phi:\ \mathcal{S}\times\mathcal{A}\to\mathcal{P}(\mathcal{Z})$，我们定义 $\mathcal{S}\times\mathcal{A}\to\mathcal{P}(\mathcal{S})$中的如下状态分布：

$$p_z^\phi(\cdot|\boldsymbol{s},\boldsymbol{a}):=\sum_{\boldsymbol{z}}\phi(\boldsymbol{z}|\boldsymbol{s},\boldsymbol{a})p_z(\cdot|\boldsymbol{s},\boldsymbol{z}). \tag{45}$$

此外，我们定义最优 $\boldsymbol{z}$ 分布，使其尽可能地模仿原始转移分布 ：

$$\phi^*(\cdot|\boldsymbol{s},\boldsymbol{a}):=\operatorname*{arg\,min}_{\phi\in\mathcal{S}\times\mathcal{A}\to\mathcal{P}(\mathcal{Z})}D_{\mathrm{TV}}(p(\cdot|\boldsymbol{s},\boldsymbol{a})\|p_z^\phi(\cdot|\boldsymbol{s},\boldsymbol{a})). \tag{46}$$

对于原始马尔可夫决策过程中的策略 π，我们将其对应的最优潜在策略定义如下：

$$\pi_z^{\phi^*}(\cdot|\boldsymbol{s}):=\sum_{\boldsymbol{a}\in\mathcal{A}}\pi(\boldsymbol{a}|\boldsymbol{s})\phi^*(\cdot|\boldsymbol{s},\boldsymbol{a}). \tag{47}$$

直观而言，该策略生成的潜在动作分布会尽可能贴近 π 的下一状态分布。现在，我们给出原始马尔可夫决策过程与可预测潜在MDP之间的 π 性能界。

引理 D.6. 对于任意策略 π，若原始转移动力学与最优潜在动力学之间的全变差距离有界，如下所示：

$$\mathbb{E}_{(\boldsymbol{s},\boldsymbol{a})\sim d^\pi(\boldsymbol{s},\boldsymbol{a})}[D_{\mathrm{TV}}(p(\cdot|\boldsymbol{s},\boldsymbol{a})\|p_z^{\phi^*}(\cdot|\boldsymbol{s},\boldsymbol{a}))]\leq\epsilon_a, \tag{48}$$

原始*MDP*与可预测潜在*MDP*之间的性能差异满足以下不等式：

$$|J_{\mathcal{M}}(\pi)-J_{\mathcal{M}_P}(\pi_z^{\phi^*})|\leq\frac{2R\gamma\epsilon_a}{(1-\gamma)^2}. \tag{49}$$

Prof. 潜在MDP中，$\pi_z^{\phi^*}$ 在状态 $\boldsymbol{s} \in \mathcal{S}$ 下的下一状态分布可写为如下形式 ：

$$p_z(\cdot|\boldsymbol{s}) = \sum_{\boldsymbol{z}\in\mathcal{Z}} \pi_z^{\phi^*}(\boldsymbol{z}|\boldsymbol{s}) p_z(\cdot|\boldsymbol{s},\boldsymbol{z}) \tag{50}$$

$$= \sum_{\boldsymbol{z}\in\mathcal{Z}} \sum_{\boldsymbol{a}\in\mathcal{A}} \pi(\boldsymbol{a}|\boldsymbol{s}) \phi^*(\boldsymbol{z}|\boldsymbol{s},\boldsymbol{a}) p_z(\cdot|\boldsymbol{s},\boldsymbol{z}) \tag{51}$$

$$= \sum_{\boldsymbol{a}\in\mathcal{A}} \pi(\boldsymbol{a}|\boldsymbol{s}) \sum_{\boldsymbol{z}\in\mathcal{Z}} \phi^*(\boldsymbol{z}|\boldsymbol{s},\boldsymbol{a}) p_z(\cdot|\boldsymbol{s},\boldsymbol{z}) \tag{52}$$

$$= \sum_{\boldsymbol{a}\in\mathcal{A}} \pi(\boldsymbol{a}|\boldsymbol{s}) p_z^{\phi^*}(\cdot|\boldsymbol{s},\boldsymbol{a}). \tag{53}$$

因此， $J_{\mathcal{M}_P}$（$\pi_z^{\phi^*}$）等于$J_{\mathcal{M}^{\phi*}}$（π），其中 $\mathcal{M}^{\phi^*}$ 定义为（$\mathcal{S}$， $\mathcal{A}$， μ， $p_z^{\phi^*}$， r）。然后，式（49）由引理D.4得出。

□

接下来, 我们界定 潜在MDP 与潜在基于模型的马尔可夫决策过程之间的$\pi_z^{\phi^*}$ 性能差异 .

引理 D.7。如果（i）真实动力学 p_z 与学习得到的模型 $\hat{}p_z$ 之间，以及（ii）策略$\pi_z^{\phi^*}$ 与数据收集策略 $\pi_{\mathcal{D}_P}$ 之间的全变差距离有界，满足

$$\mathbb{E}_{(\boldsymbol{s},\boldsymbol{z})\sim d^{\pi_{\mathcal{D}_P}}(\boldsymbol{s},\boldsymbol{z})}[D_{\mathrm{TV}}(p_z(\cdot|\boldsymbol{s},\boldsymbol{z})\|\hat{p}_z(\cdot|\boldsymbol{s},\boldsymbol{z}))] \le \epsilon'_m, \tag{54}$$

$$\mathbb{E}_{\boldsymbol{s}\sim d^{\pi_{\mathcal{D}_P}}(\boldsymbol{s})}[D_{\mathrm{TV}}(\pi_z^{\phi^*}(\cdot|\boldsymbol{s})\|\pi_{\mathcal{D}_P}(\cdot|\boldsymbol{s}))] \le \epsilon'_\pi, \tag{55}$$

$\pi_z^{\phi^*}$ 在 $\mathcal{M}_P$ 与$\hat{}\,\mathcal{M}_P$ 之间的性能差异满足以下不等式：

$$|J_{\mathcal{M}_P}(\pi_z^{\phi^*}) - J_{\hat{\mathcal{M}}_P}(\pi_z^{\phi^*})| \le \frac{R}{(1-\gamma)^2}(4\epsilon'_\pi + 2\gamma\epsilon'_m). \tag{56}$$

证明。 证明与定理D.5相同。 □

现在，我们得到 PMA 的如下性能界：

定理 D.8。 （*PMA* 性能界）如果抽象损失、模型误差和策略差异满足以下界限：

$$\mathbb{E}_{(\boldsymbol{s},\boldsymbol{a})\sim d^{\pi}(\boldsymbol{s},\boldsymbol{a})}[D_{\mathrm{TV}}(p(\cdot|\boldsymbol{s},\boldsymbol{a})\|p_z^{\phi^*}(\cdot|\boldsymbol{s},\boldsymbol{a}))] \le \epsilon_a, \tag{57}$$

$$\mathbb{E}_{(\boldsymbol{s},\boldsymbol{z})\sim d^{\pi_{\mathcal{D}_P}}(\boldsymbol{s},\boldsymbol{z})}[D_{\mathrm{TV}}(p_z(\cdot|\boldsymbol{s},\boldsymbol{z})\|\hat{p}_z(\cdot|\boldsymbol{s},\boldsymbol{z}))] \le \epsilon'_m, \tag{58}$$

$$\mathbb{E}_{\boldsymbol{s}\sim d^{\pi_{\mathcal{D}_P}}(\boldsymbol{s})}[D_{\mathrm{TV}}(\pi_z^{\phi^*}(\cdot|\boldsymbol{s})\|\pi_{\mathcal{D}_P}(\cdot|\boldsymbol{s}))] \le \epsilon'_\pi, \tag{59}$$

原始 *MDP* 与可预测的潜变量模型马尔可夫决策过程之间的 π 性能差异有如下界限：

$$|J_{\mathcal{M}}(\pi) - J_{\hat{\mathcal{M}}_P}(\pi_z^{\phi^*})| \le \frac{R}{(1-\gamma)^2}(2\gamma\epsilon_a + 4\epsilon'_\pi + 2\gamma\epsilon'_m). \tag{60}$$

证明。 由引理D.6和引理 D.7可得

$$|J_{\mathcal{M}}(\pi) - J_{\hat{\mathcal{M}}_P}(\pi_z^{\phi^*})| \le |J_{\mathcal{M}}(\pi) - J_{\mathcal{M}_P}(\pi_z^{\phi^*})| + |J_{\mathcal{M}_P}(\pi_z^{\phi^*}) - J_{\hat{\mathcal{M}}_P}(\pi_z^{\phi^*})| \tag{61}$$

$$\le \frac{R}{(1-\gamma)^2}(2\gamma\epsilon_a) + \frac{R}{(1-\gamma)^2}(4\epsilon'_\pi + 2\gamma\epsilon'_m) \tag{62}$$

$$= \frac{R}{(1-\gamma)^2}(2\gamma\epsilon_a + 4\epsilon'_\pi + 2\gamma\epsilon'_m). \tag{63}$$

□

E. PMA 与 DADS 的理论比较

在本节中，我们从互信息最大化、熵近似和状态覆盖率三个方面，对 PMA 和 DADS 的目标进行理论比较。

互信息最大化。我们首先比较 PMA 和 DADS 目标中的互信息（MI）项 $I(\boldsymbol{S}';\boldsymbol{Z}|\boldsymbol{S})$。这里，我们忽略 PMA 的信息增益项 即， $\beta = 0$，稍后将对此进行讨论。此外，为了简化对信息论目标的分析，我们假设潜在动作空间 $\mathcal{Z}$ 是离散的，即 $\mathcal{Z} = [Z]$。在这些假设下，PMA 与 DADS 的差异主要体现在探索策略 $\pi_e(\boldsymbol{z}|\boldsymbol{s})$ 上。DADS 会在每次轨迹展开开始时首先采样一个潜在动作 $\boldsymbol{z}$ （$\pi_e(\cdot|\boldsymbol{s}_0) = \text{Unif}(\mathcal{Z})$），并在整个回合期间保持该动作不变（$\pi_e(\boldsymbol{z}_t = \boldsymbol{z}_{t-1}|\boldsymbol{s}_t) = 1$）；而 PMA 始终使用均匀随机策略（$\pi_e(\cdot|\boldsymbol{s}) = \text{Unif}(\mathcal{Z})$），因为我们已作出假设 $\beta = 0$。

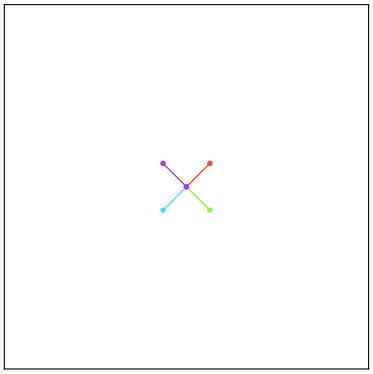


图 12. DADS 策略 ($Z = 4$) 的轨迹示例。

我们首先考虑互信息目标 $I(\boldsymbol{S}';\boldsymbol{Z}|\boldsymbol{S})$ 的以下上界，

$$\max_{\pi_e,\pi_z} I(\boldsymbol{S}';\boldsymbol{Z}|\boldsymbol{S}) = \max_{\pi_e,\pi_z} H(\boldsymbol{Z}|\boldsymbol{S}) - H(\boldsymbol{Z}|\boldsymbol{S},\boldsymbol{S}') \tag{64}$$

$$\leq \max_{\pi_e,\pi_z} H(\boldsymbol{Z}|\boldsymbol{S}). \tag{65}$$

公式（65）在 $p^{\pi_e,\pi_z}(\boldsymbol{z}|\boldsymbol{s})$ 为均匀随机分布时达到其最大值 $\log Z$，其中 $p^{\pi_e,\pi_z}(\boldsymbol{s},\boldsymbol{z})$ 是来自
这些策略的状态-潜在动作分布。PMA 使用 $\pi_e(\cdot|\boldsymbol{s}) = \text{Unif}(\mathcal{Z})$，这恰好对应于这一最优条件。另一方面，DADS 的分布可以改写为

$$p^{\pi_e,\pi_z}(\boldsymbol{z}|\boldsymbol{s}) = \frac{p^{\pi_e,\pi_z}(\boldsymbol{s}|\boldsymbol{z})p^{\pi_e,\pi_z}(\boldsymbol{z})}{p^{\pi_e,\pi_z}(\boldsymbol{s})} \propto \frac{p^{\pi_e,\pi_z}(\boldsymbol{s}|\boldsymbol{z})}{p^{\pi_e,\pi_z}(\boldsymbol{s})}. \tag{66}$$

与 PMA 不同，这不一定对应于均匀分布。例如，在图 12 的红点状态处，我们可以看到，对于相应的潜在动作，$p^{\pi_e,\pi_z}(\boldsymbol{s}|\boldsymbol{z})$ 非零，而对于其他潜在动作则为零。这可能使 DADS 在互信息最大化方面达不到最优。

熵近似。 在边缘熵项 $H(\boldsymbol{S}'|\boldsymbol{S})$ 的近似中也能发现类似差异。PMA 和 DADS 都使用以下近似：

$$H(\boldsymbol{S}'|\boldsymbol{S}) = \mathbb{E}\left[-\log p^{\pi_e,\pi_z}(\boldsymbol{s}'|\boldsymbol{s})\right] \tag{67}$$

$$= \mathbb{E}\left[-\log \int p^{\pi_e,\pi_z}(\boldsymbol{z}|\boldsymbol{s})p^{\pi_e,\pi_z}(\boldsymbol{s}'|\boldsymbol{s},\boldsymbol{z})d\boldsymbol{z}\right] \tag{68}$$

$$\approx \mathbb{E}\left[-\log \int u(\boldsymbol{z})p^{\pi_e,\pi_z}(\boldsymbol{s}'|\boldsymbol{s},\boldsymbol{z})d\boldsymbol{z}\right] \tag{69}$$

$$\approx \mathbb{E}\left[-\log\left(\frac{1}{L}\sum_{i=1}^{L} p^{\pi_e,\pi_z}(\boldsymbol{s}'|\boldsymbol{s},\boldsymbol{z}_i)\right)\right], \tag{70}$$

其中，它将 $p^{\pi_e,\pi_z}(\boldsymbol{z}|\boldsymbol{s})$ 近似为均匀分布 $u(\boldsymbol{z})$，而 $\boldsymbol{z}_i$ 是从 $u(\cdot)$ 中采样的。由于上述相同原因，这种近似在 DADS 中可能并不准确；而 PMA 满足 $p^{\pi_e,\pi_z}(\boldsymbol{z}|\boldsymbol{s}) = u(\boldsymbol{z})$，因此对数中的近似是精确的。

状态覆盖。PMA 和 DADS 的另一项差异在于，PMA 的目标中包含信息增益项。这是因为仅靠互信息项并不一定能覆盖整个状态空间，因为互信息对于输入随机变量的任意可逆变换是不变的，也就是说，对于任意随机变量 $\boldsymbol{X}$ 和 $\boldsymbol{Y}$，以及可逆函数 f 和 g，都有 MI(I; $\boldsymbol{X}$) $= I(f(\boldsymbol{X}); g(\boldsymbol{Y}))$。因此，即使状态覆盖有限，互信息也可以达到最大值，而且不一定会促进探索（Campos Cam´u˜nez 等人，2020年；Strouse 等人，2022年；Park 等人，2022年），这使得 PMA 需要另一个用于最大化状态-动作覆盖的项。

F. 训练流程

F.1. PMA 训练流程

PMA 正在训练采用 SAC（Haarnoja 等人，2018年b）。我们介绍 PMA 的若干其他训练细节 A.

算法 1可预测MDP抽象（PMA）

初始化动作解码器 $\pi_z(\boldsymbol{a}|\boldsymbol{s},\ \boldsymbol{z})$、VLB预测模型 $\hat{}p_z(\boldsymbol{s}'|\boldsymbol{s},\ \boldsymbol{z};\ \boldsymbol{\phi})$、集成预测模型 $\{\hat{}p_z(\boldsymbol{s}'|\boldsymbol{s},\ \boldsymbol{z};\ \boldsymbol{\theta_i})\}$，（可选）探索策略 $\pi_e(\boldsymbol{z}|\boldsymbol{s})$、同策略随机缓冲区 $\mathcal{D}_S$、同策略确定性缓冲区 $\mathcal{D}_D$、回放缓冲区 $\mathcal{D}$
对于 $i \leftarrow 1$ 到（训练轮次数）**执行**
  对于 $j \leftarrow 1$ 到（每轮步数）$/2$ **执行**
    采样潜在动作 $\boldsymbol{z} \sim \pi_e(\boldsymbol{z}|\boldsymbol{s})$
    采样动作 $\boldsymbol{a} \sim \pi_z(\boldsymbol{a}|\boldsymbol{s},\ \boldsymbol{z})$
    将状态转移 $(\boldsymbol{s},\ \boldsymbol{z},\ \boldsymbol{a},\ \boldsymbol{s}')$ 添加到 $\mathcal{D}_S$、$\mathcal{D}$
  结束循环
  对于 $j \leftarrow 1$ 至（每轮步数）$/2$ **执行**
    采样潜在动作 $\boldsymbol{z} \sim \pi_e(\boldsymbol{z}|\boldsymbol{s})$
    计算确定性动作 $\boldsymbol{a} = \mathbb{E}[\pi_z(\cdot|\boldsymbol{s},\boldsymbol{z})]$
    将状态转移 $(\boldsymbol{s},\boldsymbol{z},\boldsymbol{a},\boldsymbol{s}')$ 添加到 $\mathcal{D}_D$
  结束循环
  使用来自 $\mathcal{D}_S$的小批次拟合 VLB预测模型
  使用来自 $\mathcal{D}_D$的小批次拟合集成预测模型
  使用 $r(\boldsymbol{s},\boldsymbol{z},\boldsymbol{a},\boldsymbol{s}') = \log \hat{}p_z(\boldsymbol{s}'|\boldsymbol{s},\boldsymbol{z};\ \boldsymbol{\phi}) - \log$ 训练动作解码器 $\frac{1}{L}\sum_{i=1}^{L} \hat{}p_z(\boldsymbol{s}'|\boldsymbol{s},\boldsymbol{z}_i;\ \boldsymbol{\phi}) + \beta \cdot \mathrm{Tr}[\mathbb{V}_i[\boldsymbol{\mu}(\boldsymbol{s},\boldsymbol{z};\ \boldsymbol{\theta}_i)]]$ 使用来自 $\mathcal{D}$的小批次进行 SAC 训练
  （可选）使用来自 $\mathcal{D}_S$的小批次，通过 SAC 训练探索策略 π_e
  清空在策略缓冲区 $\mathcal{D}_S, \mathcal{D}_D$
结束循环

回放缓冲区。由于我们同时训练动作解码器和模型，因此在使用旧的离策略样本训练 PMA 的各个组件时必须格外谨慎。只要重新计算内在奖励，我们就可以使用旧样本训练动作解码器 $\pi_z(\boldsymbol{a}|\boldsymbol{s},\ \boldsymbol{z})$（因为 SAC 是一种离策略算法）；但不能使用旧样本训练预测模型或探索策略 $\pi_e(\boldsymbol{z}|\boldsymbol{s})$。因此，我们仅为动作解码器使用回放缓冲区，并使用在策略数据训练其他组件。

采样策略。 PMA 有两种不同类型的预测模型：用于近似 $I(\boldsymbol{S}';\boldsymbol{Z}|\boldsymbol{S})$ 的 VLB预测模型 $\hat{}p_z(\boldsymbol{s}'|\boldsymbol{s},\ \boldsymbol{z};\boldsymbol{\phi})$，以及由 E 模型组成的集成 $\{\hat{}p_z(\boldsymbol{s}'|\boldsymbol{s},\ \boldsymbol{z};\boldsymbol{\theta_i})\}$，用于近似 $I(\boldsymbol{S}';\boldsymbol{\Theta}|\boldsymbol{\mathcal{D}}\ \boldsymbol{Z})$。虽然可以直接使用集成输出的均值作为 VLB预测模型，但我们发现，采用不同的采样策略分别训练它们会有所帮助。具体而言，我们使用随机轨迹训练 VLB 预测模型，并使用确定性轨迹训练集成模型。由于测试时我们始终使用预训练动作解码器输出的确定性动作，因此使用这类确定性动作训练潜在预测模型会更有利。因此，在每个训练轮次中，我们使用确定性动作采样一半的轮次状态转移来训练集成模型，另一半则使用随机动作来训练其他组件。测试时，我们使用集成模型输出的均值进行基于模型的规划或基于模型的强化学习。

我们在算法中总结了 PMA 的完整训练流程1。

F.2. MPPI 训练流程

MPPI（Williams 等人，2016年）是一种基于模型预测控制（Testud 等人，1978）的零阶规划算法，通过迭代优化随机采样的动作来寻找最优动作序列。具体而言，在步数 t，MPPI 旨在通过 M次迭代优化，找到长度为 H 的最优（潜在）动作序列（$\boldsymbol{z}_t,\ \boldsymbol{z}_{t+1},\ \ldots\ldots,\ \boldsymbol{z}_{t+H-1}$）。每次迭代中，MPPI 从高斯分布 $\boldsymbol{z}_t^{(i)}{}_{:\ t+H-1} \sim \mathcal{N}(\boldsymbol{\mu}_{t:t+H-1},\ \boldsymbol{\Sigma})$ 中采样 N 个动作序列，其中 $i \in [N]$，$\boldsymbol{\mu}_{t:t+H-1}$ 是当前均值参数，$\boldsymbol{\Sigma}$ 是固定的对角协方差矩阵。随后，它使用潜在预测模型 $\hat{}p_z(\boldsymbol{s}'|\boldsymbol{s},\ \boldsymbol{z})$ 根据预测状态计算预测奖励之和 $\{R^{(i)} := \sum_{j=t}^{t+H-1} \hat{}r_j^{(i)}\}$，并按如下方式更新均值参数：

$$\boldsymbol{\mu}_{t:t+H-1} \leftarrow \frac{\sum_{i\in[N]} e^{\alpha R^{(i)}} \boldsymbol{z}_{t:t+H-1}^{(i)}}{\sum_{i\in[N]} e^{\alpha R^{(i)}}}, \tag{71}$$

算法2使用 PMA 的 MPPI

初始化均值参数 $\boldsymbol{\mu}_{0:T-1}$
对 $t \leftarrow 0$ 至 $T-1$ **执行**
  对 $m \leftarrow 0$ 至 $M-1$ **执行**
    采样 N 潜在动作序列 $\boldsymbol{z}_{t:t+H-1}^{(i)} \sim \mathcal{N}(\boldsymbol{\mu}_{t:t+H-1}, \boldsymbol{\Sigma})$，用于 $i \in [N]$
    使用潜在预测模型的预测状态计算预测奖励之和 $\{R^{(i)} := \sum_{j=t}^{t+H-1} \hat{r}_j^{(i)}\}$ $\hat{p}_z(\boldsymbol{s}'|\boldsymbol{s}, \boldsymbol{z})$
    更新 $\boldsymbol{\mu}_{t:t+H-1}$，使用公式 (71)
  结束循环
  执行单个动作 $\boldsymbol{a}_t = \mathbb{E}[\pi_z(\cdot|\boldsymbol{s}_t, \boldsymbol{z}_t)]$，使用 $\boldsymbol{z}_t = \boldsymbol{\mu}_t$ 并从环境中获取 $\boldsymbol{s}_{t+1}$
结束循环

算法3使用 PMA 的 MBPO

1 $\pi(\boldsymbol{z}|\boldsymbol{s})$ $\mathcal{D}_{\text{frozen}}$ $\mathcal{D}$**：初始化任务策略（）、冻结回放缓冲区、回放缓冲区：对于至（# 训练轮次）执行** 3：$d \leftarrow$ **TRUE** 4：**对于** $j \leftarrow 1$ **至（# 每轮步数）执行** 5：**如果** $d =$ **TRUE，则** 6：**以** P **的概率从** $\mu(\cdot)$ **中采样** $\boldsymbol{s}$**，或以** $(1-P)$ **的概率从** $\mathcal{D}_{\text{frozen}}$ **中采样** 7：**结束 if** 8：**采样潜在动作** $\boldsymbol{z} \sim \pi(\boldsymbol{z}|\boldsymbol{s})$ 9：**预测下一个状态** $\boldsymbol{s}' = \mathbb{E}[\hat{p}_z(\cdot|\boldsymbol{s}, \boldsymbol{z})]$10：**使用** $\boldsymbol{s}$ **和** $\boldsymbol{s}'$ **计算预测奖励** r **和预测终止** d 11：**如果（当前时域长度）** $\geq H$**，则** 12：$d \leftarrow$ **TRUE** 13：**结束 if** 14：**将状态转移（**$\boldsymbol{s}, \boldsymbol{z}, r, \boldsymbol{s}'$**）添加到** $\mathcal{D}$ 15：**结束 for** 16：**使用来自** $\mathcal{D}$**的小批次，通过 SAC 训练任务策略略** 17：**结束 for** 18：**对于** $i \leftarrow 1$ **至（# 评估轨迹展开）执行** 19：**当未终止时循环** 20：**计算潜在动作** $\boldsymbol{z} = \mathbb{E}[\pi(\cdot|\boldsymbol{s})]$21：**计算动作** $\boldsymbol{a} = \mathbb{E}[\pi_z(\cdot|\boldsymbol{s}, \boldsymbol{s})]$22：**从环境中获取** r **和** $\boldsymbol{s}'$ 23：**结束 while** 24：**结束 for**

其中 α 是一个温度超参数。经过 M 次迭代优化后，智能体仅执行第一个潜在动作，并重复此过程以寻找下一个最优序列 $(\boldsymbol{z}_{t+1}, \boldsymbol{z}_{t+2}, \ldots\ldots, \boldsymbol{z}_{t+H})$。我们在算法 2中总结了 MPPI 的完整训练流程。

F.3. MBPO训练流程

MBPO（Janner 等，2019）是一种 Dyna风格（Sutton，1991）的基于模型的强化学习算法，它从中间环境状态开始进行截断的基于模型的展开，并在其基础上训练无模型强化学习方法。在我们的零样本设置中，MBPO 使用无监督训练后恢复的回放缓冲区来采样起始状态。具体而言，在每个训练轮次中，MBPO 使用学习得到的预测模型生成多条长度为 H 的基于模型的截断轨迹（$\boldsymbol{s}_0, \boldsymbol{a}_0, r_0, \boldsymbol{s}_1, \boldsymbol{a}_1, r_1, \ldots, \boldsymbol{s}_{H-1}, \boldsymbol{a}_{H-1}, r_{H-1}$），其中 $\boldsymbol{s}_0$ 以 P 的概率从真实的初始状态分布 $\mu(\cdot)$ 中采样，或者以 $(1-P)$ 的概率从恢复的回放缓冲区 $\mathcal{D}_{\text{frozen}}$ 中采样。随后，它使用 SAC（Haarnoja 等，2018a）根据收集到的轨迹更新任务策略 $\pi(\boldsymbol{z}|\boldsymbol{s})$，并重复这一过程。请注意，恢复的回放缓冲区仅用于提供起始状态。此外，如果我们将 P 设为 1，并将 H 设为原始时域长度 T，则 MBPO 等同于原始软演员-评论家算法

完全基于模型的状态转移进行训练。完成 MBPO 的训练后，我们通过在真实环境中测试学习到的任务策略来衡量其性能。我们在算法 3 中总结了 MBPO 的完整训练流程。

G. 实现细节

我们在 LiSP（Lu 等人，2021年）公开发布的代码库基础上实现了 PMA。我们在以下代码仓库中公开了我们的实现：https://github.com/seohongpark/PMA。我们的实验在由 A5000 或类似 GPU 组成的内部集群上运行。每次实验运行耗时不超过两天。

G.1. 环境

我们的基准测试采用 OpenAI Gym（Todorov等人，2012年；Brockman等人，2016年）中的七种 MuJoCo 机器人环境：HalfCheetah、Ant、Hopper、Walker2d、倒立摆、InvertedDoublePendulum 和 Reacher。我们基本沿用Sharma 等人（2020年）使用的环境配置。所有环境的回合长度均设为200 。对于 HalfCheetah、Ant、Hopper 和 Walker2d，我们从策略的输入中排除全局坐标。对于 Hopper 和 Walker2d，我们将动作重复设为5 ，使其离散化时间尺度与其他环境相近。关于提前终止，我们遵循原始环境配置：Ant、Hopper、Walker2d、倒立摆和 InvertedDoublePendulum 设有提前终止条件，而 HalfCheetah 和 Reacher 没有。

G.2. 任务

我们描述了 13 个任务的奖励函数。我们基本沿用各环境原始任务的奖励方案。对于会提前终止的环境，智能体还会在每一步额外获得 1 的奖励。

HalfCheetah 前进： 奖励为智能体的 x速度。

HalfCheetah 后退： 奖励为智能体的 x速度的相反数。

Ant 向东： 奖励为智能体的 x速度。

Ant 向北： 奖励为智能体的 y速度。

Hopper 前进： 奖励是智能体的 x速度。

Hopper 跳跃： 奖励是 0 和智能体的 z速度中的最大值。

Walker2d 前进： 奖励是智能体的 x速度。

Walker2d 后退： 奖励是智能体的 x速度的相反数。

I倒立摆保持： 奖励是智能体 x位置平方的相反数 t.

InvertedPendulum 前进： 奖励是智能体的 x速度。

InvertedDoublePendulum 静止： 奖励是智能体 x位置平方的相反数。

InvertedDoublePendulum 前进： 奖励是智能体 x速度。

Reacher 触及： T 奖励是目标位置与指尖之间欧几里得距离的相反数 .

G.3. 超参数

我们在表1至表3中列出了实验所用的超参数。对于图5中的 MPPI结果，我们针对每种方法和任务分别调整 MOPO 惩罚项 λ（表2），其中考虑了 $\lambda \in \{0, 1, 5, 20, 50\}$。对于图6中的 MBPO 结果，我们针对每种方法和任务分别调整 MOPO 惩罚项 λ、展开时域长度 H 和重置概率 P（表3），其中考虑了 $\lambda \in \{0, 1, 5, 20, 50\}$ 和（H, P） $\in${(1, 0)、(5, 0)、(15, 0)、(15, 0.5)、(50, 0)、(50, 0.5)、(200, 1)}。对于图7中的 SAC 结果，我们使用（H, P） $=$(200, 1)。在 Ant 中训练 MBPO 时，我们还将 max(0, ·) 应用于惩罚增强奖励，以防止智能体为了避免负奖励而倾向于提前终止。

表 *1.* 超参数。

超参数	值
# 训练轮次	10000
# 每轮环境步数	4000
# 每轮梯度步数	64（策略）、32（模型）、1（随机网络蒸馏网络）
回合长度 T	200
小批量大小	256
折扣因子 γ	0.995
经验回放缓冲区大小	100000
# 隐藏层	2
# 每层隐藏单元数	512
非线性函数	ReLU
优化器	Adam（Kingma & Ba，2015年）
学习率	3×10^{-4}
目标网络平滑系数 τ	0.995
集成规模	5
奖励尺度	10
潜在动作维度 $\lvert\mathcal{Z}\rvert$	$\lvert\mathcal{A}\rvert$
PMA、DADS 熵近似样本数 L	100
PMA集成方差尺度 β	5（Walker2d），50（Hopper），0.03（其他）
CM（分歧）集成方差尺度 β	0.3（Walker2d），1（HalfCheetah、Ant），3（Hopper），10（其他）
MPPI 时域长度 H	15
MPPI 种群规模 N	256
MPPI 迭代次数 M	10
MPPI 温度 α	1
MPPI 方差 Σ	I

表*2.* MOPO λ 的 MPPI结果（图 5）。

Task	PMA	DADS	CM（随机）	CM（分歧）	CM（RND）
HalfCheetah 前进	1	1	1	1	1
HalfCheetah 后退	1	1	1	1	1
Ant 向东	20	20	20	20	20
Ant 向北	20	20	20	20	20
Hopper 前进	5	5	1	5	5
Hopper 跳跃	1	1	5	5	5
Walker2d 前进	1	1	1	1	1
Walker2d 后退	1	1	1	1	1
InvertedPendulum 静止	1	1	1	1	1
InvertedPendulum 前进	5	5	5	5	5
InvertedDoublePendulum 静止	0	5	5	5	5
InvertedDoublePendulum 前进	5	5	1	5	1
Reacher 触及	0	0	0	0	0

表 3。（MOPO λ，MBPO H，MBPO P）的 MBPO 结果（图 6）。

Task	PMA	DADS	CM（随机）	CM（分歧）	CM（RND）
HalfCheetah 前进	(1, 15, 0.5)	(1、15、0。5)	(1，15，0.5)	(1, 15,0.5)	(1,15, 0.5)
Ant 向东	(50,15, 0.5)	(10,15, 0.5)	(50,15, 0.5)	(10，15，0.5)	(50，15，0。5)
Hopper 前进	(1, 200,1)	(1, 200,1)	(1，200，1)	(1，50，0.5)	(1，15，0。5)
Walker2d 前进	(1、50、0)	(1、50、0)	(1，15，0)	(1，50，0)	(1，15，0)