arXiv:2602.02831v1 [cs.RO] 2 Feb 2026

# 自适应线性路径模型式扩散

Yutaka Shimizu 和 Masayoshi Tomizuka

**摘要—将基于模型的控制方法与扩散模型相结合，正受到越来越多的关注。尽管我们已在具有挑战性的任务中见证了许多令人瞩目的机器人控制成果，但扩散模型的性能对调度参数的选择极为敏感，因此参数调优成为最关键的挑战之一。我们提出线性路径基于模型的扩散（LP-MBD），以受流匹配启发的线性概率路径取代方差保持调度。由此得到一种在几何上易于解释且相互解耦的参数化方式，能够降低调优复杂度，并为自适应提供稳定基础。在此基础上，我们提出自适应 LP-MBD（ALP-MBD），利用强化学习根据任务复杂度和环境条件调整扩散步数与噪声水平。在数值研究、Brax 基准测试以及移动机器人轨迹跟踪任务中，LP-MBD 在保持出色性能的同时简化了调度，而 ALP-MBD 进一步提升了鲁棒性、适应性和实时效率。我们的代码可通过匿名代码仓库获取：https://anonymous.4open.science/r/adaptive_l inear_path_model_based_diffusion-C58C**

## I. 引言

扩散模型在各种现实应用中取得了显著成功，尤其是在图像与视频生成方面。近年来，其应用已扩展至视觉领域之外，越来越多的研究开始探索其在机器人控制中的应用。人们提出了若干方法 [1] [2]，将扩散模型与基于模型的控制相结合，以解决复杂的轨迹优化问题。这些方法与基于采样的优化方法密切相关，并已证明能够有效处理具有非线性、非光滑动力学以及非凸目标函数和约束的问题。

然而，基于模型的扩散的性能在很大程度上取决于噪声调度的设计。在基于扩散的方法中，采样过程由一系列扩散步数组成，噪声逐步注入轨迹，再逐渐减少，以生成可行且经过改进的样本。每一步注入的噪声规模对性能有重要影响；即使采用带有方差保持调度[1], 的简单线性参数化，也需要调整多个参数才能获得最优解。这些参数在塑造整体噪声分布特征时彼此紧密耦合，因此难以找到最优取值。此外，最优参数往往会随系统状态和问题复杂度而变化，这常常导致参数选择过于保守。即使在同一任务中，不同条件下的最优参数取值也可能相差很大。我们通过图 1 所示的自动驾驶汽车示例来说明这一观点。

本文首先介绍线性路径基于模型的扩散（LP-MBD），这是对原始基于模型的扩散（MBD）框架的一种改进，以受 FlowMatching 启发的线性概率路径取代方差保持调度 [3][4]。与原始 MBD 相比，LP-MBD 具有多项优势。线性概率路径在先验分布与目标分布之间提供了简单且具有几何可解释性的插值；在常见的高斯假设下，它可以与最优传输相吻合，从而获得清晰的理论依据。更重要的是，与方差保持调度不同，线性路径公式无需多个参数彼此复杂耦合来确定噪声幅度。因此，LP-MBD 所需的超参数更少，使调优过程更加简便且易于解释。

在此基础上，我们进一步提出自适应线性路径模型扩散（ALP-MBD），通过加入基于强化学习的动态参数调整模块，对 LP-MBD 进行扩展。ALP-MBD 根据任务复杂度和环境条件调整调度参数。例如，在具有挑战性的场景中，它可以增加扩散步数，或提高噪声水平以促进更广泛的探索，从而实现更灵活、更有效的控制。相反，在简单或结构良好的条件下，它可以同时降低噪声幅度和扩散步数，在不牺牲解的质量的前提下提高采样效率。

我们在多种环境和设置下对 LP-MBD 和 ALP-MBD 进行了广泛评估，以展示它们的效率和性能。实验包括数值研究、在 Brax 中实现的基于 MuJoCo 的基准测试，以及一项移动机器人轨迹跟踪任务。

本文的主要贡献有三点：

**1)线性路径基于模型的扩散（LP-MBD）：**我们提出一种受流匹配启发的线性路径调度器，可提供几何上易于解释且相互解耦的参数化。这减轻了调参负担，并为自适应扩展奠定了稳定基础。2) **自适应线性路径 MBD（ALP-MBD）：**在 LP-MBD 的基础上，我们开发了一种自适应调度器，利用强化学习根据环境状态调整扩散步数和噪声水平，从而提升鲁棒性和效率。

Yutaka Shimizu 和 MasayoshiTomizuka 任职于美国加利福尼亚州伯克利市加州大学伯克利分校机械工程系，邮编 94720 {purewater0901, tomizuka}@berkeley.edu

**3)全面评估：**我们通过数值研究、Braxbench 基准测试和移动机器人轨迹跟踪，对 LP-MBD 和 ALP-MBD 进行了验证，展示了其在不同场景下提升的样本效率、控制质量和适应能力。

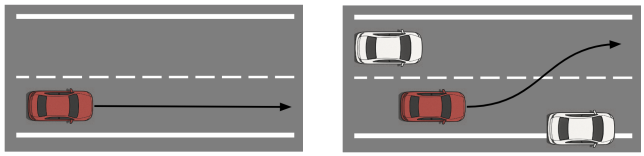


图1. 左图展示了一个简单场景，其中红色自车在无障碍环境中行驶。右图描绘了一个更为复杂的情形：红色自车试图超越一辆白色车辆，同时避开从后方驶来的汽车。在后一种场景中，会施加额外的安全约束，通常需要更多扩散步数才能获得最优轨迹。

## II. 相关工作

轨迹优化是安全、精确的机器人控制的基本组成部分。主要有两种方法：(i) 使用基于梯度的方法 [5], [6], [7], [8],直接求解底层优化问题，以及 (ii) 采用基于采样的方法 [9], [10] 搜索最优轨迹。本文聚焦于后者，因为它能更好地适应复杂的动力学和目标函数。MPPI [11], [12]是使用最广泛的采样算法之一。它可视为交叉熵方法（CEM） [13], 的软加权变体，其中样本被平滑加权，而不是由硬性精英集筛选。CEM本身与CMA-ES[14], 密切相关，但CMA-ES通过有原则地自适应协方差和步长进一步改进了更新，使其对具有强相关性或各向异性的问题更加稳健。

生成模型为捕获复杂分布提供了强大的工具。扩散模型 [15], [16], [17] 和流匹配 [3], [4]因能够表示复杂分布而受到广泛关注，并已应用于规划问题 [18], [19], [20], [21]。近期研究，例如 [22], [23]，探讨了基于采样的方法与扩散模型之间的联系；而 [2]则强调了扩散模型与 MPPI 之间的关联。

强化学习（RL） [24] 已被应用于多个领域，包括将基于模型的规划与基于采样的规划器相结合 [25], [26], [27]。这些研究表明，将强化学习与采样方法相结合，相比当前最先进的无模型方法能够取得更优的性能 [28], [29], [30], [31], [32]。除规划之外，强化学习还被用于微调扩散模型[33], [34], [35]。例如， [36] 采用强化学习来优化扩散模型中的初始噪声，从而能够在复杂分布中进行模式选择。在本文中，我们利用强化学习根据环境状态自适应地确定调度参数。

## III. 线性路径基于模型的扩散

在本节中，我们首先回顾基于模型的扩散（MBD） [1]，并讨论扩散调度的选择为何会成为关键的实际瓶颈。随后，我们提出线性路径 MBD（LP-MBD），它融入了来自流匹配的线性概率路径。通过在先验分布与目标分布之间构建最优传输插值，这一形式化不仅简化了超参数调优，还提供了更稳定且在几何上更可解释的轨迹生成过程。

### *A.* 方差保持型基于模型的扩散

轨迹优化（TO）是引导机器人驶向预定目标的基本工具。我们考虑一个有限时域问题，其状态为 $x_t \in \mathbb{R}^{n_x}$，控制量为 $u_t \in \mathbb{R}^{n_u}$:

$$\min_{x_{1:T},\, u_{1:T}} \quad J(x_{1:T}; u_{1:T}) \tag{1a}$$

$$\text{s.t.} \quad x_{t+1} = f_t(x_t, u_t), \tag{1b}$$

$$g_t(x_t, u_t) \le 0, \quad t = 0, \dots, T-1, \tag{1c}$$

其中， $J$ 是用户指定的代价， $f_t$ 表示系统动力学， $g_t$ 是约束。令 $Y = [x_{1:T};\ u_{1:T}]$表示决策变量。按照 [1], ，我们将 TO 重述为从以下目标概率分布中进行采样的问题

$$p_0(Y) \ \propto \ p_d(Y)\, p_J(Y)\, p_g(Y), \tag{2}$$

其中 $p_d$ 确保动力学可行性， $p_g$ 编码约束，而 $p_J(Y) \propto \exp\big(-J(Y)/\lambda\big)$ 则通过温度参数 $\lambda > 0$偏向较低的代价。

MBD [1] 使用扩散过程对决策变量进行采样。标准的前向过程会逐渐扰动初始分布 $p_0$，使其趋近于各向同性高斯分布 $p_N$，噪声由调度方案 $\{\alpha_i\}_{i=1}^N$控制:

$$Y^{(i)} = c_{i,0}\, Y^{(0)} + c_{i,1}\, \varepsilon, \tag{3}$$

$$c_{i,0} = \sqrt{\bar{\alpha}_i}, \quad c_{i,1} = \sqrt{(1-\bar{\alpha}_i)}, \quad \bar{\alpha}_i = \prod_{k=1}^{i} \alpha_k, \tag{4}$$

其中 $\varepsilon \sim \mathcal{N}(0,\ I)$。这里， $Y_{(0)}$ 表示从初始分布 $p_0$中抽取的样本，该分布对应于优化问题中尚未受到任何扰动的原始决策变量。请注意，系数$c_{i,\ 0}$ 和$c_{i,\ 1}$ 定义了前向过程的噪声调度，因此会对扩散模型的性能产生关键影响。

根据式 (3) 和式 (4)，如果将初始分布的方差标准化为 $\mathrm{Var}[Y^{(0)}] = 1$，那么对于所有 $i$， $Y^{(i)}$ 的方差都保持为 1。这种不变性源于系数 $c_{i,0}$ 和 $c_{i,1}$ 的选取满足 $c_{i,\ 0}^2 + c_{i,1}^2 = 1$，从而在整个前向扩散过程中保持总体方差不变。因此，这种调度 $\{\alpha_i\}$ 称为方差保持（VP）噪声调度。在实践中，MBD 使用一种简单的线性VP调度，其中 $\beta_i$ 在 $i = 1, \dots, N$ 上进行线性插值，且 $\beta_0 = 1.0 \times 10^{-4}$和 $\beta_1 = 1.0 \times 10^{-2}$，并且$\alpha_i = 1 - \beta_i$。我们将这种特定的 MBD 实现称为

VP-MBD，以便将其与所提出的方法区分开来。

与从数据中学习得分函数的无模型扩散规划器不同，MBD 利用已知目标函数和动力学来估计得分函数 $\nabla_{Y^{(i)}} \log p_i(Y^{(i)})$，并执行一种蒙特卡洛得分上升式去噪步骤：

$$\begin{aligned} Y^{(i-1)} &= \frac{c_{i-1,0}}{c_{i,0}}\Big(Y^{(i)} + c_{i,1}^2\,\nabla_{Y^{(i)}} \log p_i\big(Y^{(i)}\big)\Big) \\ &= \frac{1}{\sqrt{\alpha_i}}\Big(Y^{(i)} + (1-\bar{\alpha}_i)\,\nabla_{Y^{(i)}} \log p_i\big(Y^{(i)}\big)\Big) \end{aligned} \tag{5}$$

通过贝叶斯定理和前向核，得分函数可表示为对“干净”轨迹的期望：这些轨迹 $Y^{(0)}$从高斯提议分布中采样，并根据基于模型的目标分布重新加权：

$$\begin{aligned} &\nabla_{Y^{(i)}} \log p_i\big(Y^{(i)}\big) \;\approx \\ &-\frac{Y^{(i)}}{1-\bar{\alpha}_i} + \frac{\sqrt{\bar{\alpha}_i}}{1-\bar{\alpha}_i}\;\underbrace{\frac{\sum_{Y^{(0)}\in\mathcal{Y}^{(i)}_{\text{VP-MBD}}} Y^{(0)}\,w(Y^{(0)})}{\sum_{Y^{(0)}\in\mathcal{Y}^{(i)}_{\text{VP-MBD}}} w(Y^{(0)})}}_{\text{(importance-weighted average)}} \end{aligned} \tag{6}$$

其中 $w(Y) = p_J(Y)\,p_g(Y)$，且 $\mathcal{Y}^{(i)}$ VP-MBD 服从高斯分布：

$$\mathcal{Y}^{(i)}_{\text{VP-MBD}} \sim \mathcal{N}\Big(\frac{Y^{(i)}}{\sqrt{\bar{\alpha}_i}}, \frac{1-\bar{\alpha}_i}{\bar{\alpha}_i}\Big) \tag{7}$$

对于 TO，候选 $Y^{(0)} = [x_{1:T};\ u_{1:T}]$ 通过展开 $x_{t+1} = f_t(x_t, u_t)$（射击法）使其满足动力学可行性，然后通过 $w(Y)$ 进行评分。将式（6）代入式（5），得到

$$Y^{(i-1)} = \sqrt{\bar{\alpha}_{i-1}}\,\frac{\sum_{Y^{(0)}\in\mathcal{Y}^{(i)}_{\text{VP-MBD}}} Y^{(0)}\,w(Y^{(0)})}{\sum_{Y^{(0)}\in\mathcal{Y}^{(i)}_{\text{VP-MBD}}} w(Y^{(0)})} \tag{8}$$

尽管 VP-MBD 性能出色，但噪声调度参数仍然难以调优。在采用简单线性调度的方差保持型（VP）形式中，调度由三元组 $(\beta_0,\ \beta_1,\ T)$ 指定，它们共同决定 $\{\alpha_i\}$ 和累积噪声水平 $\{\bar{\alpha}_i\}$。具体而言，最大噪声水平并非仅由端点 $\beta_0$ 和 $\beta_1$ 控制，还会受到扩散步数总量 $T$ 的影响。因此，这些参数以复杂的方式相互作用，决定每一步的有效噪声尺度。这种相互依赖性使调优过程更加困难，因为每个参数对探索—细化权衡的影响都高度间接且取决于具体任务，通常需要反复试错才能获得令人满意的性能。

### *B.* 线性路径基于模型的扩散

受流匹配启发，我们在干净轨迹 $Y^{(0)}$ 与标准高斯分布 $\varepsilon \sim \mathcal{N}(0,\ I)$ 之间采用线性概率路径：

$$Y_t \;=\; (1-t)\,Y^{(0)} \;+\; t\,\varepsilon, \qquad t\in[0,1]. \tag{9}$$

在均匀网格 $t_i = \frac{i}{N}$, $i = 0,\dots,N$ 上离散化 $t$，得到的调度会使噪声水平随“时间”线性增加。

$$Y^{(i)} \;=\; (1-t_i)\,Y^{(0)} \;+\; t_i\,\varepsilon \tag{10}$$

当 $c_{i,0} = 1-t_i$ 且 $c_{i,1} = t_i$ 时，这对应于公式（3）。在端点处，$i = 0$ $(t_i = 0)$ 给出 $Y_{(i)} = Y_{(0)}$（干净轨迹），而 $i = N$ $(t_i = 1)$ 则得到 $Y^{(N)} = \varepsilon$（标准高斯噪声）。因此，线性概率路径在一个端点还原干净样本，在另一个端点得到纯噪声，从而在这两个分布之间提供了清晰且易于理解的插值。

我们采用相同的蒙特卡洛得分函数上升型去噪步骤：

$$Y^{(i-1)} = \frac{1-t_{i-1}}{1-t_i}\Big(Y^{(i)} + t_i^2\,\nabla_{Y^{(i)}} \log p_i\big(Y^{(i)}\big)\Big) \tag{11}$$

其中，得分函数为

$$\begin{aligned} &\nabla_{Y^{(i)}} \log p_i\big(Y^{(i)}\big) \;\approx \\ &-\frac{Y^{(i)}}{t_i^2} + \frac{1-t_i}{t_i^2}\;\underbrace{\frac{\sum_{Y^{(0)}\in\mathcal{Y}^{(i)}_{\text{LP-MBD}}} Y^{(0)}\,w(Y^{(0)})}{\sum_{Y^{(0)}\in\mathcal{Y}^{(i)}_{\text{LP-MBD}}} w(Y^{(0)})}}_{\text{(importance-weighted average)}} \end{aligned} \tag{12}$$

与 VP-MBD 类似，我们从以下高斯分布中采样 $Y^{(0)}$

$$\mathcal{Y}^{(i)}_{\text{LP-MBD}} \sim \mathcal{N}\Big(\frac{Y^{(i)}}{1-t_i}, \frac{t_i^2}{(1-t_i)^2}\Big) \tag{13}$$

将式 (12) 代入式 (11)，可得

$$Y^{(i-1)} = (1-t_{i-1})\,\frac{\sum_{Y^{(0)}\in\mathcal{Y}^{(i)}_{\text{LP-MBD}}} Y^{(0)}\,w(Y^{(0)})}{\sum_{Y^{(0)}\in\mathcal{Y}^{(i)}_{\text{LP-MBD}}} w(Y^{(0)})} \tag{14}$$

尽管 LP-MBD 使用了直观的噪声调度，但式（13）中的高斯提议分布具有标准差 $\sigma_i = \frac{t_i}{1-t_i}$，它会随着 $t_i \to 1$ 增大而发散。这意味着初始反向去噪步骤会对整个轨迹空间进行采样；这在理论上有效，但在实践中没有必要。在轨迹优化中，控制输入通常受系统约束限制，因此从无界域进行采样效率低下。将噪声调度限制在自然范围内，可以在不损失正确性的前提下提高效率，因此我们采用有界调度策略。我们不将插值延伸至 $t_i = 1.0$，而是将调度截断于最大值 $t_{\max} < 1$，从而确保 $\mathcal{Y}^{(i)}$ LP-MBD 的方差保持有限：

$$t_i \in [0,\, t_{\max}], \qquad t_{\max} < 1. \tag{15}$$

当 $t_i = t_{\max}$ 时，高斯分布 $\mathcal{Y}^{(i)}$ LP-MBD 的标准差达到最大值。

$$\sigma_{\max} \;=\; \frac{t_{\max}}{1-t_{\max}} \quad \Big(\text{equivalently, } t_{\max} = \frac{\sigma_{\max}}{1+\sigma_{\max}}\Big) \tag{16}$$

在实践中，$\sigma_{\max}$ 可以根据控制输入的可行范围确定，从而提供一种直接且易于理解的方式来设定探索上限。因此，$t_{\max}$ 可以根据式（16）计算得出。

### C. VP-MBD 与 LP-MBD 的区别

VP-MBD 与 LP-MBD 的主要区别在于它们的参数化方式，以及这对调优和适应所产生的影响。在 VP-MBD 的公式中，噪声调度由三个参数（$\beta_0, \beta_1, T$）控制，这些参数紧密耦合：改变扩散时域 $T$ 不仅会改变离散化方式，还会增大或减小最大噪声方差。这种相互依赖关系使调优过程更加复杂，也难以区分各参数对探索和细化的影响。

相比之下，LP-MBD 仅由两个参数决定：最大噪声尺度 $\sigma_{\max}$ 和扩散步数 $T$。这两个参数相互解耦，且具有明确的几何解释：$\sigma_{\max}$ 直接设定高斯提议分布的最大方差，而$T$ 则控制插值的分辨率。重要的是，调整 $T$ 不会影响噪声尺度的极值。这种结构上的简洁性不仅让噪声调度更易于调节，也使其更具几何可解释性，从而直观地展示了干净轨迹与噪声之间的插值过程。除了简化调节过程，这种解耦还使下一节中的自适应扩展成为可能。借助两个相互独立的参数，参数自适应方法能够高效地对其进行优化，使 LP-MBD 成为 ALP-MBD 的坚实基础。

## IV. 自适应线性路径模型式扩散

在 VP-MBD 中，扩散步数和噪声调度参数一经选定便固定下来，并在执行过程中保持不变。然而在实际应用中，即使是同一任务，轨迹优化的难度也可能有很大差异。例如，开放空间中的导航可能只需要较少的扩散步数，而障碍物密集的场景则需要更多细化和更广泛的探索。因此，有必要在线调整扩散过程，以平衡鲁棒性与效率。

LP-MBD 解耦且具有几何可解释性的参数化，为这种自适应提供了自然基础。与 VP-MBD 不同，VP-MBD 的参数（$\beta_0$、$\beta_1$、{v7>）彼此紧密耦合，且改变 $T$ 也会影响有效噪声尺度；LP-MBD 则将最大方差 $\sigma_{\max}$ 与扩散时域 $T$ 分离。这种解耦确保调整 $T$ 时只会细化离散化，而不会无意中改变探索范围。因此，学习 LP-MBD 最优参数的训练过程更加稳定、简单。相比之下，将强化学习应用于 VP-MBD 会面临不稳定性，因为参数更新可能导致底层噪声分布特征发生不可预测的变化。因此，LP-MBD 天生更适合自适应参数学习，从而能够设计出自适应 LP-MBD（ALP-MBD）。

### A. 将问题表述为强化学习问题

强化学习（RL）旨在优化被建模为马尔可夫决策过程（MDP）的环境中的决策策略（$\mathcal{S}$，$\mathcal{A}$，$\mathcal{P}$，$\mathcal{R}$，$\rho_0$，$\gamma$）。其中，$\mathcal{S}$表示状态空间，$\mathcal{A}$ 动作空间，$\mathcal{P}$ 状态转移动态，$\mathcal{R}$ 奖励函数，$\rho_0$ 初始状态分布，$\gamma$ 折扣因子。在每个时间步长，智能体观察$s \in \mathcal{S}$，选择动作 $a \in \mathcal{A}$，转移到 $s' \sim \mathcal{P}(\cdot|s, a)$，并获得奖励 $r(s, a)$。

在我们的表述中，状态空间 $\mathcal{S}$ 对应于标准强化学习中的环境状态。然而，与动作直接控制系统的传统强化学习不同，动作空间 $\mathcal{A}$ 由噪声调度参数 $T$ 和 $\sigma_{\max}$ 组成。随后，LP-MBD 使用这些估计参数生成实际应用于系统的控制信号 $u$。因此，自适应噪声调度器充当策略，在每个步骤 $t$ 输出

$$\begin{aligned}(T_t,\ \sigma_{\max,t}) &\sim \pi_\phi(\cdot \mid s_t),\\ \pi_\phi(T, \sigma_{\max} \mid s) &= \pi_T(T \mid s)\, \pi_\sigma(\sigma_{\max} \mid s),\end{aligned} \tag{17}$$

其中，$\phi$ 表示策略参数，$\pi_T$ 是关于 $\mathcal{T} = \{T_{\min}, \ldots, T_{\max}\}$的类别分布，$\pi_\sigma$是高斯分布。因此，时间步长 $t$处的动作定义为 $a_t = (T_t, \sigma_{\max,t})$。

为提高效率，我们在奖励中加入对较大扩散步数的惩罚：

$$\tilde{r}_t(s_t, a_t, u_t) = r(s_t, u_t) - w_T \frac{T_t}{T_{\max}}, \tag{18}$$

其中，$r(s_t, u_t)$是原始环境奖励，$w_T > 0$ 用于平衡任务表现与计算成本。将 LP-MBD 表示为 $\pi_{\text{LP-MBD}}(\cdot \mid s_t, a_t)$，则强化学习目标为

$$\max_{\phi} J(\phi) = \mathbb{E}_{\substack{s_t \sim \mathcal{B},\, a_t \sim \pi_\phi(\cdot|s_t),\\ u_t \sim \pi_{\text{LP-MBD}}(\cdot|s_t, a_t)}} \left[ \sum_{t=0}^{\infty} \gamma^t \tilde{r}_t \right], \tag{19}$$

其中，折扣因子 $\gamma \in$ 取值范围为(0,1)，经验回放缓冲区为 $\mathcal{B}$。图 2 展示了所提出的自适应噪声调度系统的架构。

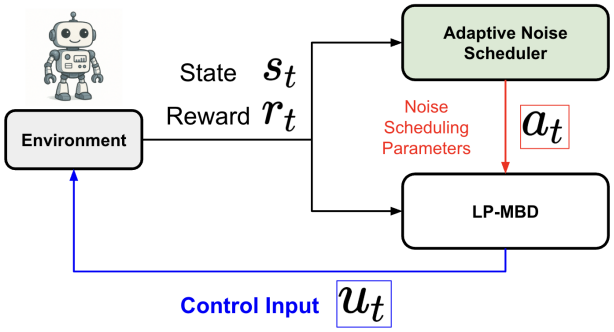



图 2. ALP-MBD 概览。环境向自适应噪声调度器提供当前状态 $s_t$ 和奖励 $r_t$，调度器输出噪声调度参数 $a_t = (T_t, \sigma_{\max,t})$。这些参数被输入 LP-MBD，以生成控制输入 $u_t$，并将其作用于环境。

### B. 实现细节

为训练自适应线性路径调度器，我们采用近端策略优化 (PPO) [37]算法。选择 PPO 是因为它在同策略训练中具有稳定性，且

在控制和离散任务中均得到广泛应用。尽管如此，所提出的框架并不依赖于特定的强化学习算法，其他方法（例如 DDPG[31], SAC[29][38][39], 或 TD3[32]）也可以应用，且不会影响其通用性。

我们在算法1中总结了训练过程。在每次迭代中，自适应噪声调度器根据当前状态采样调度参数（$T_t$，$\sigma_{\max,\ t}$），随后 LP-MBD 使用这些参数生成控制输入 $u_t$。环境返回下一状态$s_{t+1}$ 和奖励 $r_t$，并将奖励重塑为 $\tilde{r}_t$，以惩罚较大的扩散步数。随后，PPO算法使用从轨迹缓冲区收集的轨迹更新策略参数 $\phi$ 和价值函数参数 $\psi$。

**算法1**使用 PPO 训练 ALP-MBD

初始化策略参数 $\phi$、价值函数参数 $\psi$
**对于**每次迭代**执行**
  **对于** 每个环境步骤 $t$ **执行**
    观测状态 $s_t$
    样本调度参数 $a_t = (T_t,\ \sigma_{\max,\ t}) \sim \pi_\phi(\cdot|s_t)$
    生成控制输入 $u_t \sim \pi_{\text{LP-MBD}}(\cdot|s_t,\ a_t)$
    将 $u_t$ 应用于环境，接收 $(s_{t+1},\ r_t)$
    计算修改后的奖励，公式（18）
    将 $(s_t,\ a_t,\ u_t,\ \tilde{r}_t,\ s_{t+1})$ 存入缓冲区 $\mathcal{B}$
  **结束循环**
  使用 PPO 更新，根据 $\mathcal{B}$ 中的数据优化 $\phi,\ \psi$
  清空缓冲区 $\mathcal{B}$
**结束循环**

## V. 实验

本节中，我们开展了一系列实验，以评估所提出的 LP-MBD 和 ALP-MBD。这些实验旨在验证以下几个方面：(i) VP-MBD 与 LP-MBD 在行为和性能上的差异；(ii) LP-MBD 在减轻调优负担方面的优势；以及 (iii) ALP-MBD 根据观测到的状态调整关键参数（主要是扩散时域$T$ 和最大噪声标准差 $\sigma_{\max}$）的能力，以及这些自适应调整在多大程度上提升了效率。所有实验均在单块 NVIDIA RTX 4090 GPU 上进行。在整个实验过程中，我们对 VP-MBD 使用 $\beta_0 = 0.0001$ 和 $\beta_1 = 0.01$，对 LP-MBD 使用 $\sigma_{\max} = 1.8$。

### A. 数值实验

我们首先进行简单的数值实验，以评估 LP-MBD 和 ALP-MBD，并与 VP-MBD 进行比较，从而在受控环境中展示它们的相对优势。我们的数值研究（i）说明了为何 VP-MBD 的参数调优比 LP-MBD 更困难，（ii）阐明了 LP-MBD 简单且相互独立的参数集如何更好地支持自适应调度。

图3给出了三个一维示例。在来自 [1], 的第一个示例中，VP-MBD 和 LP-MBD 均在20步内达到最优解。然而，在第二个和第三个高斯目标函数中，VP-MBD 未能收敛，而 LP-MBD 仅用几步便迅速达到最优。VP-MBD 与 LP-MBD 之间的性能差距可归因于它们噪声调度的差异。例如，考虑扩散步数$T = 2$的第二个示例，我们比较 VP-MBD（式(7)）与 LP-MBD（式(13)）的高斯提议分布的标准差：

$$\begin{aligned} &\text{VP-MBD} \quad \sqrt{\frac{1-\bar{\alpha}_i}{\bar{\alpha}_i}} = [0.1,\ 0.01] \quad (i = 1, 0), \\ &\text{LP-MBD} \quad \frac{t_i}{1-t_i} = [1.8,\ 0.0] \quad (i = 1, 0). \end{aligned} \tag{20}$$

在此情况下，LP-MBD 的最大标准差为 $\sigma_{\max} = 1.8$；无论扩散时域 $T$ 如何变化，该值都保持不变，因为 $\sigma_{\max}$ 是独立确定的。相比之下，使用与第一个示例相同参数（$\beta_0 = 10^{-2}$、$\beta_1 = 10^{-4}$）的 VP-MBD 会产生方差极小的高斯提议分布，从而抑制随机性，无法进行有意义的探索。在第三个示例中，高斯混合目标与扩散步长 $T = 5$ 也出现了类似问题：LP-MBD 的最大标准差仍达到 1.8，而 VP-MBD 则仅为 0.16。因此，改变 VP-MBD 中的扩散步长 $T$ 会显著影响其噪声尺度，需要额外调节 $\beta_0$ 和 $\beta_1$，才能使其行为与 LP-MBD 相匹配。这一比较表明，所提出的 LP-MBD 采用了更直接、直观的参数化方式，相较于 VP-MBD，简化了调优过程。请注意，在这三个示例中，我们仅改变扩散步长 $T$，并对 VP-MBD 沿用相同的 $\beta_0$ 和 $\beta_1$，对 LP-MBD 沿用 $\sigma_{\max}$。

接下来，我们在一个简单的二维目标函数上评估 ALP-MBD。我们使用 REINFORCE [40],联合优化连续噪声上限 $\sigma_{\max}$ 和扩散时域 $T$，并对步数施加惩罚权重 $w_T = 1.0$，以避免不必要地增大 $T$。作为比较，我们对 VP- MBD 采用相同的强化学习流程，对 $(\beta_0, \beta_1, T)$ 进行优化，并将此基线称为自适应 VP-MBD（AVP-MBD）。为确保公平比较，两种方法均在相同的超参数和评估预算下进行相同次数的策略梯度更新（30 次），并使用各自学得的参数进行评估。

图 4 展示了受线性约束的二维高斯混合目标函数上的结果。在 $T = 3$，AVP-MBD（上图）的分布仍相对分散，样本散布在两个众数中；而 ALP-MBD（下图）则迅速将概率质量集中到高值可行区域，并在步骤 $T = 3$ 时形成紧凑的簇。这些结果表明，ALP-MBD 成功发现了一对最优参数（$\sigma_{\max}$，$T$），从而提升了收敛速度和解的质量；而 AVP-MBD

在相同的训练预算下收敛得较慢，这是因为（$\beta_0$，$\beta_1$，$T$）之间的参数耦合更强，且超参数搜索空间更大。

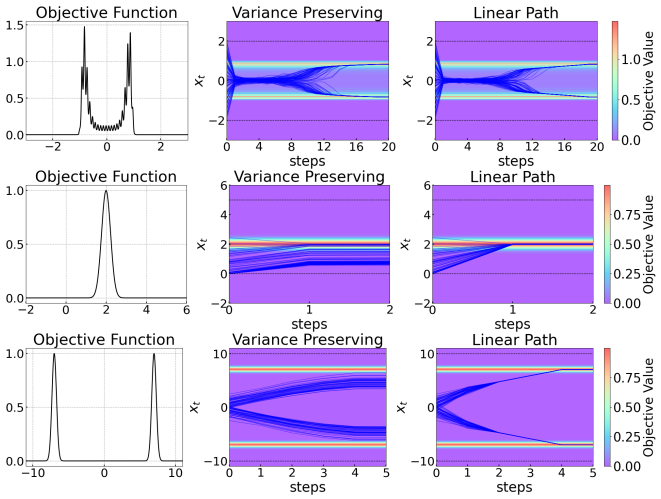



图 3. 一维示例。（上）与 [1] 中相同的示例。（中）一个简单的高斯目标函数。（下）一个包含两个众数的高斯混合目标。

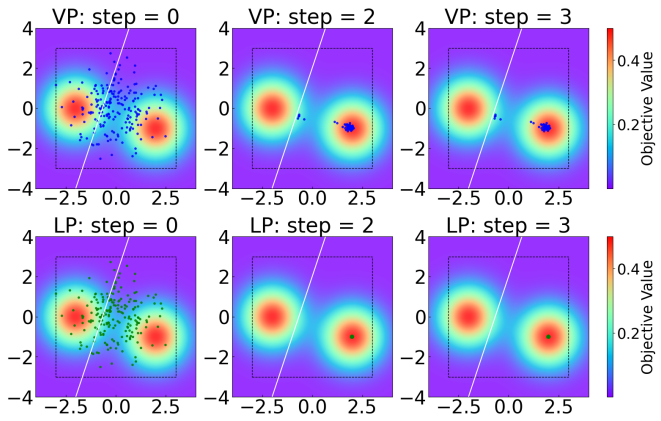



图 4. AVP-MBD 与 ALP-MBD 的比较。白线表示约束 $3x_0 - x_1 \geq 2.0$。上排显示 VP-MBD 的结果，下排显示使用估计参数的 ALP-MBD 的结果。经过 30 个训练步骤后，对于 AVP-MBD，我们得到 $\beta_0 = 0.000028$, $\beta_1 = 0.361$ $T = 3$ $\sigma_{\max} = 2.11$ $T = 3$ ；对于 ALP-MBD，我们得到 。

### *B. LP-MBD* 的评估

我们在 Brax 环境 [41]中的一系列任务上评估 LP-MBD 的性能。Brax 是一款基于物理的模拟器，专为大规模强化学习研究而设计，可为多种连续控制任务（如运动、操作和导航）提供快速且可微的动力学。

表I总结了CEM[13],MPPI[11][12]，VP-MBD[1]，以及所提出的LP-MBD所获得的每步奖励。为进行公平比较，我们使用相同的扩散步数$T$、时域 $H$和采样数量。每个数值表示在5个随机种子上的平均性能及其标准差。总体而言，LP-MBD取得了具有竞争力或在大多数任务上均取得更优性能。例如，在Hopper、HalfCheetah和Walker2D中，我们的方法获得了最高奖励，凸显了其在提升控制质量方面的有效性。

然而，值得注意的是，在 Pusher 环境中，LP-MBD 的得分低于 VP-MBD。我们推测，这种差异源于 Pusher 任务的维度更高、复杂性更强。在此类情形下，相较于简单的线性概率路径，方差保持噪声调度可能具有更强的表达能力，从而带来更好的性能。更深入的研究留待未来工作。

表 I 不同任务的每步奖励。

| 任务 | CEM | MPPI | VP-MBD | LP-MBD |
|---|---|---|---|---|
| Ant | **3.80 ± 0.43** | 2.06 ± 0.44 | 3.67 ± 0.29 | 3.76 ± 0.15 |
| Hopper | 2.24 ± 0.05 | 2.31 ± 0.05 | 2.74 ± 0.03 | **2.74 ± 0.01** |
| HalfCheetah | 1.65 ± 2.33 | 2.10 ± 0.11 | 2.53 ± 0.31 | **2.73 ± 0.12** |
| Walker2D | 2.07 ± 0.02 | 2.08 ± 0.04 | 2.31 ± 0.02 | **2.36 ± 0.02** |
| Reacher | -0.28 ± 0.04 | -0.81 ± 0.06 | **-0.17 ± 0.04** | **-0.17 ± 0.04** |
| Pusher | -0.95 ± 0.11 | -3.47 ± 0.26 | **-0.53 ± 0.11** | -0.74 ± 0.1 |

### *C. ALP-MBD* 的评估

最后，我们在自定义类 Gym 环境中实现的移动机器人轨迹跟踪任务上评估 ALP-MBD。机器人采用运动学车辆模型进行建模，其状态为 $x = [x, y, \theta, v]$，其中 $(x,\ y)$ 表示位置，$\theta$ 表示偏航角，$v$ 表示速度。控制输入为加速度 $a$ 和转向角速度 $w$，其动力学演化如下：

$$\begin{aligned} x_{k+1} &= x_k + v_k \cos(\theta_k)\Delta t, \\ y_{k+1} &= y_k + v_k \sin(\theta_k)\Delta t, \\ \theta_{k+1} &= \theta_k + w_k \Delta t, \\ v_{k+1} &= v_k + a_k \Delta t, \end{aligned} \tag{21}$$

时间步长为 $\Delta t$。目标是在这些动力学条件下跟踪预定义的参考路径，同时确保行为安全平稳。为量化性能，我们采用奖励函数，对偏离参考轨迹、航向不匹配、速度误差和碰撞进行惩罚：

$$\begin{aligned} r = &-w_{\text{lat}}\, d_{\text{lat}} - w_{\text{yaw}}\,(\theta - \theta_{\text{ref}})^2 \\ &- w_v\,(v - v_{\text{ref}})^2 - w_{\text{collision}}\,\mathbf{1}_{\text{collision}}, \end{aligned} \tag{22}$$

其中，$d_{\text{lat}}$ 表示相对于参考路径的绝对横向偏差，$\psi_{\text{ref}}$ 和 $v_{\text{ref}}$ 分别表示期望偏航角和速度，$\mathbf{1}_{\text{collision}}$ 表示碰撞指示变量，$w_\bullet$ 表示各惩罚项的权重系数。强化学习参数调优模块的输入状态表示为 $(d_{\text{lat}},\ d_\theta,\ d_{\text{vel}},\ dx_{\text{obs}},\ dy_{\text{obs}})$，其中 $d_\theta = \theta - \theta_{\text{ref}}$、$d_{\text{vel}} = v - v_{\text{ref}}$ 和 $(dx_{\text{obs}},\ dy_{\text{obs}})$ 表示自车与周围障碍物之间的相对距离。我们在具有 S 形轨迹的环境中训练 ALP-MBD，该轨迹在图 5 中以黑线表示。图中的灰色矩形表示环境中的静态障碍物。训练完成后，我们比较 ALP-MBD、VP-MBD 和 LP-MBD 的性能。

为确保比较公平，我们为所有算法设置相同的规划时域 $H = 50$ 和采样数量 100。在本实验中，我们使用 Python 3.11 和 PyTorch 实现各方法。

表 II 汇总了每种算法在五个随机种子上评估得到的平均单回合奖励。在三种 MBD 变体中，所提出的自适应方法取得了最高奖励。对于 ALP-MBD，所报告的扩散步数 $T$ 和最大噪声标准差 $\sigma_{\max}$ 是在五个种子上取平均得到的。就运行时间而言，VP-MBD 和 LP-MBD 仅测量生成控制输入所需的时间，而 ALP-MBD 的运行时间还包括参数估计过程（一次用于确定 $T$ 和 $\sigma_{\max}$的策略前向传播）。这一额外步骤导致其每步延迟高于 LP-MBD 和 VP-MBD。尽管如此，参数估计引入的开销很小，ALP-MBD 仍然非常适合实时控制应用。

我们也在图 5 中展示了 ALP-MBD 生成的轨迹。ALP-MBD 在避障时会自适应地同时增加扩散步数和最大噪声标准差，而在沿参考线稳定巡航时则将其降低。这种自适应行为反映了底层目标的复杂度：当自车靠近障碍物时，目标函数变得更加复杂，需要更多迭代次数和更广的探索范围才能收敛到可行解。相比之下，当车辆在附近没有障碍物的情况下跟随参考轨迹时，目标保持简单，从而能够以更少的扩散步数和更小的方差实现收敛。这种自适应性通过仅在具有挑战性的场景中分配更多计算资源来提升 VP-MBD 的效率，同时在更简单的环境中保持效率。

此外，我们在具有不同参考轨迹的新环境中评估了训练后 ALP-MBD 模型的泛化能力。如图 6 所示，ALP-MBD 呈现出与先前结果一致的规律：当自车靠近障碍物时，会增加扩散步数和最大噪声标准差；当车辆距离障碍物较远时，则会减少这两者。我们还观察到，车辆在参考轨迹上行驶急转弯时，会增加扩散步数和最大标准差。在这种情况下，车辆必须施加最大转向角，才能与参考路径保持对齐，因此需要增加扩散步数和噪声方差。值得注意的是，ALP-MBD 在训练期间并未接触过这条轨迹，展现了其泛化至此前未见场景的能力。

表 II 平均单回合奖励

| **方法** | 步数 $T$ | $\sigma_{\max}$ | 奖励 | 运行时间 [ms] |
|---|---|---|---|---|
| VP-MBD | 17 | - | -3498 ± 116 | 38.3 |
| LP-MBD | 17 | 1.8 | -3465 ± 147 | **36.9** |
| ALP-MBD | 16.9 ± 1.14 | 1.82 ± 0.18 | **-3342 ± 128** | 45.5 |

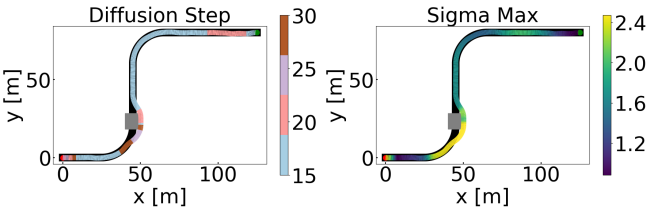



图 5. ALP-MBD 在移动机器人轨迹跟随任务中的表现。红点表示起点，绿点表示目标点。黑线表示参考轨迹，灰色矩形为障碍物。

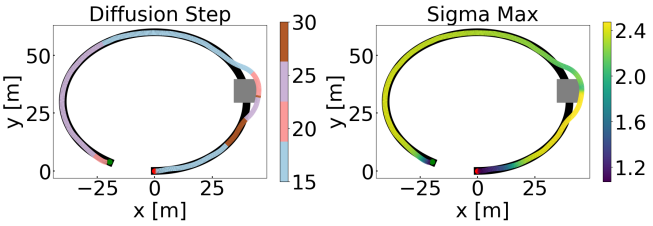



图 6. ALP-MBD 在移动机器人椭圆路径轨迹跟随任务中的表现。红点表示起点，绿点表示目标点。黑线表示参考轨迹，灰色矩形为障碍物。我们使用了在不同环境中训练的 ALP-MBD。

## VI. 结论

我们提出了线性路径基于模型的扩散（LP-MBD）及其自适应扩展（ALP-MBD）。LP-MBD 以受流匹配启发的线性概率路径取代方差保持调度，从而实现了具有几何可解释性且相互解耦的参数化。这不仅减轻了调参负担，还提供了强化学习自适应所需的结构稳定性。在此基础上，ALP-MBD 根据任务复杂度动态调整扩散步数和噪声水平，从而兼顾鲁棒性与效率。通过数值示例、Brax 基准测试以及移动机器人的轨迹跟踪实验，我们证明 LP-MBD 在保持较强性能的同时简化了调度，而 ALP-MBD 则进一步提升了适应性和实时控制能力。

## 参考文献


[1] C. Pan、Z. Yi、G. Shi 和 G. Qu，“基于模型的扩散用于轨迹优化”，载于第三十八届神经信息处理系统年会，2024年。 [在线]。可访问：https://openreview.net/forum?id=BJndYScO6o [2] H. Xue、C. Pan、Z. Yi、G. Qu 和 G. Shi，“通过扩散式退火进行基于全阶采样的模型预测控制，实现力矩级运动控制”，*2025年IEEE国际机器人与自动化会议（ICRA）*，第4974–4981页，2024年。 [在线]。可访问：https://api.semanticscholar.org/CorpusID:272832507

[3] Y. Lipman、R. T. Q. Chen、H. Ben-Hamu、M. Nickel 和 M. Le，“用于生成建模的流匹配”，载于第十一届国际学习表征会议，2023年。 [在线]。可访问：https://openreview.net/forum?id=PqvMRDCJT9t [4] Y. Lipman、M. Havasi、P. Holderrieth、N. Shaul、M. Le、B. Karrer、R. T. Q. Chen、D. Lopez-Paz、H. Ben-Hamu 和 I. Gat，“流匹配指南与代码”，2024年。 [在线]。可访问：https://arxiv.org/abs/2412.06264

[5] C. Hargraves 和 S. Paris，“使用非线性规划与配点法进行直接轨迹优化”，《制导、控制与动力学杂志》，第10卷，第4期，第338–342页，1987年。 [在线]。可访问：https://doi.org/ 10 2514 3 20223 。/ 。[6] T. A. Howell、B. E. Jackson 和 Z. Manchester，“Altro：一种用于约束轨迹优化的快速求解器”，载于2019年IEEE/RSJ国际智能机器人与系统会议（IROS），2019年，第7674–7679页。[7] S.-J. Kim、K. Koh、M. Lustig、S. P. Boyd 和 D. M. Gorinevsky，“用于大规模 L1 正则化最小二乘的内点法”，IEEE《信号处理精选主题期刊》，第1卷，第606–617页，2007年。 [在线]。可访问：https://api.semanticscholar.org/ CorpusID:[8] G. Shi、W. Honig、X. Shi、Y. Yue 和 S.-J. Chung，“Neural-swarm2：利用学习得到的交互关系进行异构多旋翼无人机群的规划与控制”，IEEE《机器人学汇刊》，第38卷，第2期，第1063–1079页，2022年。[9] D. Fox、W. Burgard 和 S. Thrun，“用于避碰的动态窗口法”，IEEE《机器人与自动化杂志》，第4卷，第1期，第23–33页，1997年。[10] T. M. Howard、C. J. Green、A. Kelly 和 D. Ferguson，“在复杂环境中对高性能移动机器人导航的可行运动进行状态空间采样”，J. Field Robot.，第25卷，第6–7期，第325–345页，2008年。[11] G. Williams、P. Drews、B. Goldfain、J. M. Rehg 和 E. A. Theodorou，“使用模型预测路径积分控制进行激进驾驶”，载于2016年IEEE国际机器人与自动化会议（ICRA），2016年，第1433–1440页。[12] ——，“信息论模型预测控制：理论及其在自动驾驶中的应用”，IEEE《机器人学汇刊》，第34卷，第6期，第1603–1622页，2018年。[13] R. Rubinstein，“组合优化与连续优化的交叉熵方法”，Method. Comput. Appl. Prob.，第1卷，第2期，第127–190页，1999年9月。 [在线]。可访问：https://doi.org/10.1023/A: 1010091220143[14] N. Hansen、S. D. M¨uller 和 P. Koumoutsakos，“通过协方差矩阵自适应（CMA-ES）降低去随机化进化策略的时间复杂度”，《进化计算》，第11卷，第1期，第1–18页，2003年。[15] J. Ho、A. Jain 和 P. Abbeel，“去噪扩散概率模型”，载于《第34届神经信息处理系统国际会议论文集》，NIPS ’20系列。美国纽约州雷德胡克：Curran Associates Inc.，2020年。[16] J. Song、C. Meng 和 S. Ermon，“去噪扩散隐式模型”，载于国际学习表征会议，2021年。 [在线]。可访问：https://openreview.net/forum?id=St1giarC HLP[17] D. Podell、Z. English、K. Lacey、A. Blattmann、T. Dockhorn、J. M¨uller、J. Penna 和 R. Rombach，“SDXL：改进潜在扩散模型以实现高分辨率图像合成”，载于第十二届国际学习表征会议，2024年。 [在线]。可访问：https://openreview.net/forum?id=di52zR8xgf[18] C. Chi、S. Feng、Y. Du、Z. Xu、E. Cousineau、B. Burchfiel 和 S. Song，“扩散策略：通过动作扩散进行视觉运动策略学习”，载于《机器人学：科学与系统》（RSS），2023年。[19] C. Chi、Z. Xu、S. Feng、E. Cousineau、Y. Du、B. Burchfiel、R. Tedrake 和 S. Song，“扩散策略：通过动作扩散进行视觉运动策略学习”，《国际机器人研究杂志》，2024年。[20] M. Janner、Y. Du、J. Tenenbaum 和 S. Levine，“基于扩散的规划用于灵活行为合成”，载于国际机器学习会议，2022年。[21] X. Dai、D. Yu、S. Zhang 和 Z. Yang，“安全流匹配：利用控制障碍函数进行机器人运动规划”，CoRR，第abs/ 2504 08661, 2025 [ ] 10 4 卷。4月。在线。可访问：https://doi.org/ 。/arXiv。[22] J. Berner、L. Richter 和 K. Ullrich，“从最优控制视角看基于扩散的生成建模”，Trans. Mach. Learn. Res.，第2024卷，2022年。 [在线]。可访问：https://api.semanticscholar.org/ CorpusID:[23] V. Kurtz 和 J. W. Burdick，“生成式预测控制：面向动态且难以示范的任务的流匹配策略”，2025 [ ] 2502 13406 。在线。可访问：https://arxiv.org/abs/ 。[24] R. S. Sutton 和 A. G. Barto，《强化学习：导论》。美国马萨诸塞州剑桥：A Bradford Book，2018年。[25] N. Hansen、X. Wang与H. Su，“用于模型预测控制的时序差分学习”，载于ICML，2022年。[26] N. Hansen、H. Su与X. Wang，“TD-MPC2：面向连续控制的可扩展、鲁棒世界模型”，载于第十二届国际学习表征会议，2024年。 [在线]。可获取：https://openreview.net/forum?id=Oxh5CstDJU[27] Y. Shimizu与M. Tomizuka，“用于模型预测控制的双模拟度量”，载于第十三届国际学习表征会议，2025年。 [在线]。可获取：https: //openreview.net/forum?id = 07 7 3 F ic huE[28] M. Laskin、A. Srinivas与P. Abbeel，“Curl：面向强化学习的对比无监督表征”，载于第37届国际机器学习会议论文集，系列ICML’20。JMLR.org，2020年。[29] T. Haarnoja、A. Zhou、P. Abbeel与S. Levine，“软演员-评论家：带随机行动者的离策略最大熵深度强化学习”，载于第35届国际机器学习会议论文集，系列《机器学习研究论文集》，J. Dy与A. Krause编，第80卷。PMLR，2018年7月10–15日，第1861–1870页。 [在线]。可获取：https: //proceedings.mlr.press/v80/haarnoja18b.html[30] D. Silver、G. Lever、N. Heess、T. Degris、D. Wierstra与M. Riedmiller，“确定性策略梯度算法”，载于第31届国际机器学习会议论文集第32卷，系列ICML’14。JMLR.org，2014年，第I–387–I–395页。[31] T. P. Lillicrap、J. J. Hunt、A. Pritzel、N. Heess、T. Erez、Y. Tassa、D. Silver与D. Wierstra，“基于深度强化学习的连续控制”，载于第四届国际学习表征会议，ICLR 2016，波多黎各圣胡安，2016年5月2–4日，会议轨道论文集，Y. Bengio与Y. LeCun编，2016年。[ ] 1509 02971 在线。可获取：http://arxiv.org/abs/ 。[32] J. Wu、Q. M. J. Wu、S. Chen、F. Pourpanah与D. Huang，“A-TD3：面向连续动作空间的自适应异步双延迟深度确定性”，IEEE Access，第10卷，第128 077–128 089页，2022年。[33] M. Uehara、Y. Zhao、K. Black、E. Hajiramezanali、G. Scalia、N. L. Diamant、A. M. Tseng、S. Levine与T. Biancalani，“扩散模型的反馈高效在线微调”，载于第41届国际机器学习会议论文集，系列ICML’24。JMLR.org，2024年。[34] K. Clark、P. Vicol、K. Swersky与D. J. Fleet，“在可微奖励上直接微调扩散模型”，载于第十二届国际学习表征会议，2024年。[在线]。可获取：https://openreview.net/forum?id=1 vmSEVL19f[35] Y. Fan、O. Watkins、Y. Du、H. Liu、M. Ryu、C. Boutilier、P. Abbeel、M. Ghavamzadeh、K. Lee与K. Lee，“用于微调文生图扩散模型的强化学习”，载于第三十七届神经信息处理系统会议，2023年。[在线]。可获取：https://openreview.net/forum?id=8 OTPepXzeh[36] A. Wagenmaker、M. Nakamoto、Y. Zhang、S. Park、W. Yagoub、A. Nagabandi、A. Gupta与S. Levine，“利用潜空间强化学习引导扩散策略”，arXiv，2025年。[37] J. Schulman、F. Wolski、P. Dhariwal、A. Radford与O. Klimov，“近端策略优化算法”，CoRR，第abs/1707.06347卷，2017年。[38] T. Haarnoja、A. Zhou、K. Hartikainen、G. Tucker、S. Ha、J. Tan、V. Kumar、H. Zhu、A. Gupta、P. Abbeel与S. Levine，“软演员-评论家算法及应用”，CoRR，第abs/ 1812 05905, 卷。2018 [ ] 1812 在线。可获取：http://arxiv.org/abs/ 。 05905 [39] T. Haarnoja、S. Ha、A. Zhou、J. Tan、G. Tucker与S. Levine，“通过深度强化学习学习行走”，载于《机器人学：科学与系统》论文集，德国弗赖堡因布赖斯高，2019年6月。[40] R. S. Sutton、D. McAllester、S. Singh与Y. Mansour，“带函数逼近的强化学习策略梯度方法”，载于第13届神经信息处理系统会议论文集，系列NIPS’99。美国马萨诸塞州剑桥：麻省理工学院出版社，1999年，第1057–1063页。[41] C. D. Freeman、E. Frey、A. Raichuk、S. Girgin、I. Mordatch与O. Bachem，“Brax——面向大规模刚体仿真的可微物理引擎”，载于第三十五届神经信息处理系统会议数据集与基准测试轨道（第1轮），2021年。[在线]。可获取：https://openreview.net/forum?id=VdvDlnnjzIN