：通过在 备注 强化学习中使用 H 人类 I 直觉来提升 S 示例效率

Amogh Joshi, Adarsh Kosta, and Kaushik Roy
Purdue University, West Lafayette, IN 47907, USA
{*joshi*157, *akosta*, *kaushik*}@*purdue.edu*

***Abstract*— 神经网络执行机器人感知和控制任务（如深度和光流估计、同时定位与地图构建（SLAM）以及自动控制）的能力，近年来已得到广泛应用。深度强化学习（DeepRL）在这些设置中被广泛使用，因为它没有监督学习相关的不可持续的训练成本。然而，DeepRL 存在样本效率低下的问题，即需要大量的环境交互才能收敛到一个可接受的解决方案。现代的 RL 算法如深度 Q 学习和软演员-评论家试图解决这一缺陷，但无法提供自主机器人应用所需的解释性。人类直观地理解了在机器人技术中常见的长时间范围顺序任务。恰当地使用这种直觉可以使 RL 策略更具解释性并增强其样本效率。在这项工作中，我们提出了 SHIRE，一个利用概率图模型（PGMs）编码人类直觉并在 Deep RL 训练管道中使用的新型框架以提高样本效率。我们的框架在我们评估的环境中实现了 25 – 78% 的样本效率提升，并且几乎没有任何额外成本。此外，通过教会 RL 代理编码的基本行为，SHIRE 增强了策略的可解释性。一个实际演示进一步突显了使用我们框架训练的策略的有效性。**

I. 介绍

近年来，人工智能已被应用于人类生活的几乎所有方面。特别是，机器学习技术已在欺诈检测、医疗诊断、图像和模式识别以及语言翻译等众多领域找到了应用。在机器人视觉社区中，机器学习也引起了极大的兴趣，它已成功应用于诸如深度 [1], [2], [3] 和光流估计 [4], [5], [6], [7]、语义分割 [8], [9], [10]、同时定位与地图构建（SLAM）[11] 和自动控制 [12] 等感知任务。

在广泛使用的机器学习技术中，监督学习因其实施的简便性而被最广泛采用。尽管监督学习易于实现，但其简单性却掩盖了数据和计算需求方面的真正成本。现代监督学习模型的训练成本大约为数百 ZFLOPs（1 $Zettaflop = 10^{21} FLOPs$）[13]，一些最先进的模型需要 21YFLOPs（1 $Yottaflop = 10^{24} FLOPs$）来训练 [13]。监督训练需要大量的、标记良好且平衡的数据集，这些数据集通常很难生成 [14]，特别是对于机器人控制任务而言更是如此。此外，训练时间和成本与数据集大小成正比。因此，更大的数据集会导致更高的训练开销，使得监督训练对于自主机器人应用来说不可行。

现有机器学习方法的另一个主要缺点是它们的“黑盒”性质。机器学习模型往往既不可解释也不可理解 [15]。这个问题在监督学习模型中尤为突出，在这些模型中，缺乏明确的决策规则和没有显式的奖励信号使得追踪模型达到某一决策的“推理过程”成为不可能。这使得监督学习在如自主机器人这类安全关键的应用中不可行，因为这两种特性都是必需的。

强化学习（RL）是一种机器学习技术，通过与环境的重复交互以及奖励和惩罚系统来训练一个代理执行任务。RL 的一个主要优势是尽管训练需要大量的环境交互，但这些仍然比在大型监督数据集上的训练计算成本低得多。这使得 RL 非常适合于自主机器人等任务，在这种情况下，由于数据生成困难，数据集要么太小，要么根本不存在。最近，深度强化学习（其中使用神经网络作为代理）已成功应用于自主机器人任务，如灵巧的手部操作 [16] 和四足行走 [17]。尽管取得了这些成功，深度 RL 仍然存在一些缺点。收敛到一个好的策略依赖于诸如随机初始化、奖励函数的质量以及环境表示的粒度等因素。另一个缺点是需要大量的交互才能收敛到一个好政策。这种大量数据需求可能导致深层 RL 算法样本效率低下。这些问题传统上通过向代理提供其环境模型（也称为其世界）来解决。这种方法被称为基于模型的强化学习（MBRL）。尽管 MBRL 显著提高了样本效率，但它易受由世界模型偏差引起的问题的影响。先前的工作如 [18], [19] 表明利用对环境转换动态的先验知识可以在深层 RL 设置中导致可证明更好的样本效率 [20], [21], [22]。然而，在

现实世界中可能并非总是可以获得此类环境信息。

人类对许多机器人任务的解决方案具有直观的理解。尽管这种直觉并不总是最优的，但它包含了解决任务 [23] 的一部分最优解成分。此外，人类展示出的因果推理表现出了对当前行动长期影响的先天理解 [23]。利用这种直觉和因果推理在强化学习训练中可以提高样本效率，并使学习到的策略更加可解释。然而，以一种计算友好方式编码人类直觉仍然是一个开放性问题。

我们提出了 SHIRE（利用 H 人类 I 直觉增强 S 示例效率的 备注 强化学习框架），该框架旨在形式化编码人类直觉的过程，并将此形式主义与标准深度 RL 算法结合使用以提高样本效率。在 SHIRE 上训练的策略是 “可解释” 的，因为它们学习了在 SHIRE 框架中编码的基本行为。据我们所知，这是此类的第一个框架。我们的工作主要贡献总结如下：

- 一个框架（SHIRE），用于将人类直觉编码为概率图模型（PGM），并使用它来提高深度强化学习的样本效率和可解释性。
- 提供实验证据以表明我们的框架在简单环境中将性能提高了 25%，在复杂环境中提升了超过 78%。
- 显示在多个任务中训练时间和所需样本数量的显著减少。
- 将使用我们的框架训练的策略与实际机器人集成，并提供相同的视频演示。

II. 相关工作

据我们所知，没有旨在提高样本效率的工作利用人类直觉为来自智能体滚动/经验回放缓存的训练样本添加归纳偏置。在以下部分中，我们将提供当前文献中旨在提高强化学习中的样本效率和可解释性的研究方向的大致概述。

早期提高强化学习中样本效率的努力集中在基于模型的方法上。最近的研究包括 Hafner 等人提出的 “Dream to Control”，[20]，该方法使用想象的轨迹来学习环境模型。作者在 [21] 中使用 SimPLe 算法结合视频预测，在数据量较少的情况下进行策略学习。Janner 等人 [22] 从实际的非策略性数据分支出来，生成短暂的世界模型生成回放以用于策略训练。虽然这些方法展示了有希望的结果，但它们面临着准确建模环境、模型生成数据偏差以及计算开销随模型复杂度增加而增长等挑战。

世界建模的挑战导致了开发替代方法，这些方法专注于通过算法设计提高样本效率。熵最大化技术用于在策略 [24] 和离策略 [25] 算法中改进探索性和鲁棒性，通过将奖励最大化目标与熵最大化相结合来实现这一点。软演员-评论家（SAC）[26] 是一种无模型的离策略算法，它结合了熵最大化的做法与演员-评论家框架，适用于连续状态和动作空间，从而实现高效的样本学习策略。然而，SAC 需要一个大的经验回放缓冲区，导致高内存开销，并且由于演员网络不准确地抽样后验概率而遭受不稳定性的困扰。

到目前为止讨论的策略缺乏可解释性，这是现实世界机器人领域中至关重要的因素。在机器人领域中的可解释性可以被理解为根据（a）与其交互的环境或（b）代理对其自身动态的理解来解释策略决策的能力。PILCO[27] 依赖于第一种解读，并使用高斯过程模型处理环境因素，但通过忽略时间相关性低估了未来不确定性。DeepPILCO[28] 在 PILCO 的基础上进行了改进，采用了贝叶斯深度模型，但仍依赖于对奖励函数和世界状态的访问，这些在实践中通常是不可用的数量。Chua 等人 [29] 使用概率性和确定性世界模型的集合来提升对未来状态不确定性的估计，在使用 PETS 算法时实现了接近无模型方法的渐近性能。然而，它们仍然会在长期任务时间范围内遭受误差累积的问题。

Mutti 等人的 [19]C-PSRL 算法类似于 SHIRE，都使用因果图的先验知识来指导学习过程，但它们服务于不同的目的。C-PSRL 使用因果先验知识来学习环境的一个完整的因果模型，该模型可以转换为因子化马尔可夫决策过程（FMDP）以实现更高效的强化学习，如 [30], [31] 所示。然而，在 FMDP 模型中进行精确规划在计算上是不可行的 [32], [33]。此外，由于 C-PSRL 中使用的因果先验知识编码了环境动态的先前知识，它们并没有赋予代理期望的行为，导致缺乏可解释性。进一步全面分析受制于这项工作中缺少实验。相比之下，SHIRE 使用称为 “直觉网” 的因果图来建模人类对代理（而非环境）的知识，使其无模型、计算更简单且更具可解释性，解决了 C-PSRL 的主要缺点。

III. SHIRE 框架

在本节中，我们简要概述了马尔可夫决策过程、标准的深度强化学习训练算法以及我们的框架如何与它

们集成。

A. 马尔可夫决策过程

强化学习问题被定义为有限时段马尔可夫决策过程 (MDP)。一个 MDP ($\mathcal{M}$) 由一个元组 $(\mathcal{S}, \mathcal{A}, p, r, T)$ 组成，其中 $\mathcal{S}$ 是环境的状态空间，其基数为 S，$\mathcal{A}$ 是代理的动作空间，其基数为 A，p 是一个马尔可夫转换模型，使得 $p(s_{t+1}|s_t, a)$ 是在给定当前状态 s_t 和动作 a 的情况下下一个状态 s_{t+1} 的条件概率。$r : \mathcal{S} \times \mathcal{A} \to [0, 1]$ 是状态-动作对上的确定性奖励函数 (s, a)。T 表示剧集范围。

在每一集中，代理以如下方式与环境互动：初始状态是从 $\Delta(\mathcal{S})$ 中的初始状态分布中随机选择的，其中 $\Delta(\mathcal{S})$ 是 $\mathcal{S}$ 上的概率单纯形。在每一步 $t < T$，代理选择一个动作 $a_t \in \mathcal{A}$，并收到奖励 $r_t \sim r(s_t, a_t)$。同时，状态从 s_t 过渡到 $s_{t+1} \sim p(\cdot|s_t, a_t)$。当 $t = T$ 发生或任务完成时，这一集结束。

代理在状态 s_t 中选择采取行动 a_t 的策略由随机策略 $\{\pi_t\}_{t \in T} \in \Pi$ 定义，其中 Π 是策略空间，每个 π_t 都是一个函数，使得在第 t 步，在状态 s 中选择行动 a 的条件概率由 $\pi_t(a|s)$ 给出。为了评估策略 π_t 的质量，我们使用价值函数 $V_t^\pi(s) : \mathcal{S} \to [0, T]$，这是从状态 s 开始并采取步骤 t 时使用 π 所获得的奖励期望和。因此，价值函数可以定义为公式 中的形式。1

$$V_t^\pi(s) := \mathop{\mathbb{E}}_{\pi}\left[\sum_{t'=t}^{T} r(s_{t'}, a_{t'}) \middle| s_t = s\right] \forall s \in \mathcal{S}, t \in [T] \quad (1)$$

最优策略 π^* 的价值函数由 $V_t^*(s)$ 给出

$$V_t^*(s) = \mathop{argmax}_{\pi} V_t^\pi(s) \quad (2)$$

深度强化学习中的策略优化算法通过智能体与环境的交互收集经验，并试图将其收敛到一个价值函数 $V_t^{\pi'}(s)$ 尽可能接近 $V_t^*(s)$ 的策略 π'。由于几乎所有策略优化算法都采用了基于一阶优化、小的学习率、随机策略和低效探索的方法，这些方法在样本效率方面表现不佳。

B. 策略优化算法

文献中存在大量的强化学习策略优化算法。其中，最著名的有 PPO[34]、DQN[35] 和 SAC[26]。在这项工作中，我们使用修改版的 PPO 算法来训练我们的策略。PPO 算法的损失函数如公式 3 所示

$$loss_{PPO} = loss_{policy} + loss_{entropy} + loss_{value} \quad (3)$$

其中，$loss_{policy}$ 是使用裁剪优势函数计算的策略损失，$loss_{entropy}$ 是策略中的不确定性程度，而 $loss_{value}$ 是预期奖励与实际奖励之间的均方误差。SHIRE 框架在此损失中添加了一个额外项，这将在下一节中进行描述。

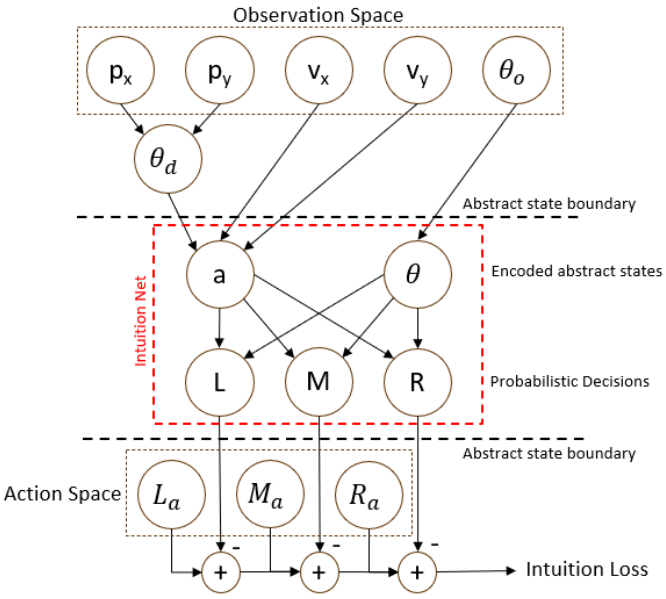

Fig. 1. 直觉编码用于月球着陆器环境

C. 直觉编码

人类具备一种非凡的能力，能够对所解决的任何问题 [23] 形成直观的理解。这种能力在机器人学中常见的长时间跨度、顺序任务中尤为明显，在这些任务中，每个动作都必须在考虑到其在整个任务时间范围内的影响后才可执行。虽然这种人类直觉并不总是任务的最佳解决方案，但它包含了对任务或机器人动力学的基本理解，这对于高效学习任务是必要的。这样的直觉有可能显著加速强化学习训练，从而提高样本效率，并为策略添加一种隐含的解释性概念。然而，以机器友好的方式编码人类直觉是一项非平凡的任务。

概率图模型（PGMs）将随机变量之间的依赖关系编码为结构化图形。贝叶斯网络是 PGMs 的一个特殊类别，它们使用有向无环图（DAGs）来表示一组随机变量的联合分布。正如文献 [36] 所示，贝叶斯网能够以紧凑、计算友好且因子化的形式捕捉随机变量之间的依赖关系（或独立性）。贝叶斯网络的另一个优势是它们能够在概率分布的形式中隐式编码知识不确定性。出于这些原因，SHIRE 使用贝叶斯网络来编码简单的任务特定人类直觉。

SHIRE 框架可以大致分为三个阶段——直觉网络构建、抽象状态编码和直觉损失计算。图 1 以图形方式说明了该工作流程在月球着陆器环境 [37] 中的应用。此环境的目标（如图 3 所示）是使用三个推进器——主推力器和左右方向控制推力器，将飞船平稳地降落在由两面旗帜标记的着陆点上。代理的观察结果包括其位置和速度的 x 和 y 分量，以及其姿态和角速度。此环境的直觉很简单——飞船的速度矢量必须始终指向着陆点。预期该代理能够自行学习将飞船底部对准着陆区。请注意，其他直觉也是可能的。接下来的段落解释 SHIRE 框架如何利用这种直觉来加快月球着陆器任务中强化学习策略的训练。

1) 直觉网络构建: 我们将特定任务的编码人类直觉的 PGMs 称为“直觉网络”。由于直觉网络用于计算损失项，因此直觉网络的子节点始终对应于任务的动作空间。在月球着陆器环境的情况下，这会产生如图 1 所示的三个子节点，每个节点对应于三个推进器之一——主推进器（M）、左姿态控制推进器（L）和右姿态控制推进器（R）。这三个节点中的每一个都有两种状态，分别对应于推进器的“$idle$”和“$fire$”状态。控制飞船的速度向量需要知道飞船的姿态以及期望加速的方向。因此，直觉网络的父节点是期望的加速度方向（a）和飞行器的姿态（θ）。请注意，节点 a 只有两个状态 $positive$ 和 $negative$ 对应于期望加速度的方向，而节点 θ 有四个状态——每个象限一个。由于直觉网络的简单性，并确保期望的直觉得到准确反映，我们选择手动编码控制此直觉网络推理的概率分配。我们有意在直觉网络中引入不确定性以防止策略记住编码后的直觉。

2) 摘要状态编码: 在此阶段，从记录的 rollout 缓冲区中观察到的值被编码为 PGM 父节点的抽象状态。由于着陆点位于原点，航天器位置的观测值给出了速度向量 (θ_d) 的方向为 $\theta_d = \tan^{-1}\frac{p_y}{p_x}$ 这个期望方向结合了速度分量的观测值 v_x 和 v_y，得出了期望加速度方向，该方向被编码为节点 a 的抽象状态。航天器姿态的观测值经过阈值处理后给出了节点 θ 的抽象状态。

3) 直觉损失计算: 在此阶段，策略生成的动作与直觉网络的相应预测进行比较，并将两者之间的不匹配项存储在不匹配向量中。然而，这样的向量之和不是凸的。因此，为了确保收敛性，我们将其转换为一种称为“直觉损失”的凸铰链损失。此过程在算法 1 中进行了描述。在我们的实现中，算法 1 中的 for 循环被向量化以加快计算速度。这种直觉损失对每个批次进行计算，并如方程 4 所示添加到损失中。对于 Lunar Lander，由于主推进器对航天器状态的影响较大，我们选择对主推进器状态的不匹配项比方向推进器的不匹配项施加更大的惩罚。

$$loss = loss_{PPO} + loss_{intuition} \tag{4}$$

由于输入直觉网络的抽象状态编码仅依赖于回放缓冲区中的观测数据，因此与所使用的策略优化算法无关。因此，SHIRE 框架可以通过简单地添加一个标志参数集成到任何现有的策略优化算法中，如图 2 所示。在下一节中，我们将展示使用近端策略优化（PPO）算法的多种环境下的 SHIRE 框架结果，PPO 是一种在线策略算法。

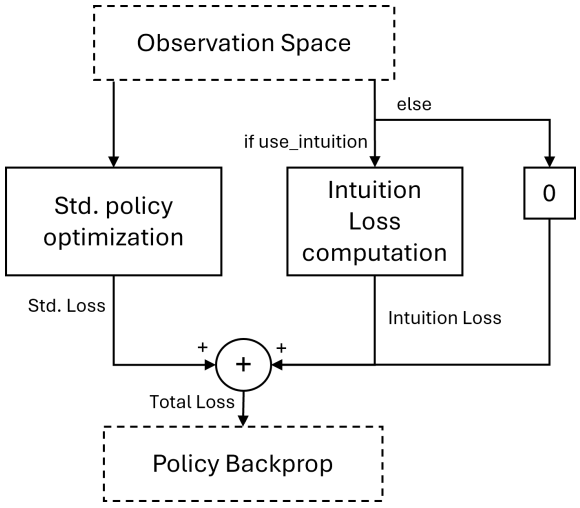

Fig. 2. 直觉损失与现有 RL 策略优化算法的集成

IV. 实验

为了评估 SHIRE 框架的有效性，我们在各种标准 gym 环境中进行了策略优化实验。在所有环境中，我们调优了一个两层的普通代理，每层有 64 个神经元，并保存了最佳结果。然后，我们启用了我们的框架，并使用相同的种子在每个环境上重新运行最佳普通实验以进行比较。因此，普通和 SHIRE 实验仅在直觉损失缩放系数方面有所不同。表 I 显示了达到解决状态所需的样本数量（即与环境的交互次数），分别有或没有使用 SHIRE。在本节中，我们描述了每个 Intuition Net 中编码的直觉，并提供了通过使用我们的框架所取得的进步以及该框架强加的开销的定量分析。请注意，当策略在一个环境中实现了超过 100 个连续评估

Algorithm 1 直观损失计算

Require: $batch \sim RolloutBuffer$ ▷ sample batch of size n from the rollout buffer

$a \leftarrow A(batch)$ ▷ Get actions from rollout buffer

$o \leftarrow O(batch)$ ▷ Get observations from rollout buffer

for $i \leftarrow 1, n$ **do** ▷ n is the batch size

$s_i \leftarrow S(o)$ ▷ encode abstract states

$e_i \leftarrow PGM(s_i)$ ▷ compute "intuitive" actions using probabilistic inference

$m_i \leftarrow M(e_i, a_i)$ ▷ compute mismatch vector

end for

$loss_{intuition} \leftarrow \sum_{i=1}^{n} max(0, 1 - m_i a_i)$ ▷ convert mismatch vector to hinge loss

return $loss_{intuition}$ ▷ return intuition loss

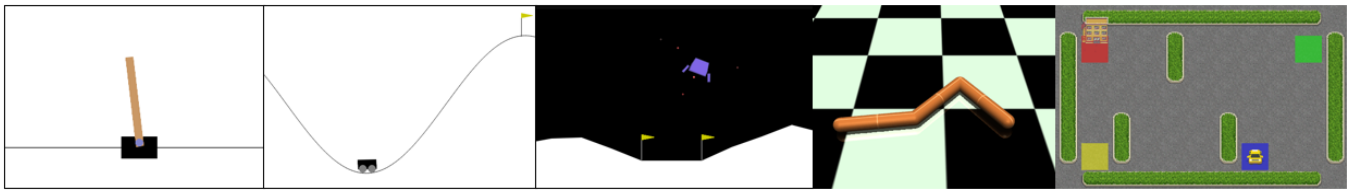

Fig. 3. 用于评估 SHIRE 框架的 Gymnasium 环境。从左到右：CartPole、MountainCar、LunarLander、Swimmer 和 Taxi。

周期内平均奖励大于等于 "解决状态奖励"（SSR）时，我们认为该环境被 "解决了"。

A. 编码直觉

对于每个测试环境，"f" 是直觉损失计算函数，并且由 Intuition Net 编码的直觉以及 "解决状态" 的定义如下：

1) 小车杆子环境: Cart Pole 环境中，代理试图保持倒立摆直立，这是我们提供结果的最简单的环境。当一个策略在 100 个回合中获得平均奖励 500.0 时，就说它 "解决了" 这个环境。直观地说，如果摆不是直立的，则将小车向摆倾斜的方向移动。这种直觉被编码为 SHIRE 中的一个简单两节点 Intuition Net，产生由公式 5 给出的直觉损失，其中 θ 是杆与垂直方向之间的角度。

$$intuition\ loss = f(\theta) \tag{5}$$

2) 山地车环境: Mountain Car 是 Andrew Moore 在 1990 年首次介绍的一种确定性 MDP[38]。该环境的目标是向汽车施加力量，使其如图所示到达山顶 3。这个环境定义 "解决" 为在 100 轮中获得超过 -110.0 的平均奖励。由于一次性爬上山是不可能的，逻辑解决方案是在山的两侧上下震荡，直到积累足够的动量来攀爬。这种直觉可以进一步简化为 "向汽车当前速度的方向施加力量"。再次，SHIRE 将这种简化的直觉编码为由方程 6 给出的两个节点的 Intuition Net，其中 v 是汽车的速度。

$$intuition\ loss = f(v) \tag{6}$$

3) 月球着陆器环境: 登月着陆器环境基于同名的 Atari 游戏，玩家的任务是在由两面旗帜标记的着陆区内尽可能平稳地降落飞船。代理可以采取的唯一行动是独立点燃（或不点燃）主引擎和左右方向推进器。一个在 100 个回合中平均奖励大于 200.0 的代理被认为 "解决了" 这个环境。解决此环境的一个简单直觉是在上一节解释的，确保着陆飞船的速度向量始终指向着陆区。这导致了由公式 7 给出的直觉损失，

$$intuition\ loss = f(p_x, p_y, v_x, v_y, \theta_o) \tag{7}$$

其中 p_x, p_y 是着陆器位置的 x- 和 y- 分量，v_x, v_y 是着陆器速度的 x- 和 y- 分量，而 θ_o 是着陆器的姿态。此外，确保着陆器的速度和方向向量之间的反平行性可以保证主推进器用于制动，从而实现更加柔和且受控的着陆。这反过来进一步提高了采样效率，如表 I 所示。这些概念被编码为五节点直觉网络，如图 1 所示。

TABLE I

解决标准 GYM 环境所需的步骤（交互）数量。样本效率的提升在与相应参数相邻的单元格中指定。SSR：已解决问题奖励，BBR：最佳基线奖励。

环境	SSR/BBR	PPO 基线		PPO 与 SHIRE			
		N(steps) to solve	Time to solve (minutes)	N(steps) to solve	Gain (%)	Time to solve (minutes)	Gain (%)
CartPole	500.0	8192	2.07	5120	**37.5**	2.51	**-21.25**
MountainCar	-110.0	510k	150.96	110k	**78.43**	36.29	**75.96**
LunarLander (w/o anti-parallelism)	200.0	120k	10.81	90k	**25**	6.83	**36.82**
LunarLander (w anti-parallelism)	200.0	120k	10.81	70k	**41.67**	5.3	**50.97**
Swimmer	110.0	3.37M	804	1.395M	**58.61**	313.8	**60.97**
Taxi	8.1	1.19M	104.34	845k	**28.99**	72	**30.99**

4) 游泳者环境: Swimmer 环境是由 Remi Coulum 介绍的一种变长、多段机器人控制环境 [39]。该环境的目标是让代理使机器人（称为“游泳者”）尽可能多地向正 X 方向移动。为了完成此任务，代理可以采取的唯一行动是对相邻段之间的铰链关节施加扭矩。由于这种环境中独立状态和控制变量的数量较大，使得它比迄今为止介绍的其他环境更具挑战性。然而，尽管难度更大，解决该环境的直觉非常简单——必须使施加在相邻关节上的扭矩方向相反。这有助于执行完成此任务所需的蛇形运动。该环境的直觉损失由方程 8 给出

$$intuition\ loss = f(\theta_1, \theta_2) \tag{8}$$

其中 θ_i 是 i^{th} 关节链接之间的角度。

游泳者是一个“未解决”的环境，即不存在设定的 SSR 值。因此，为了确保公平比较，我们训练了一个不使用我们框架的策略，并将其达到的最佳奖励作为 SSR 的替代。我们将这个奖励称为“最佳基线奖励”（BBR）。样本效率计算为达到此奖励值所需交互次数（步骤数）。我们的实验使用一个三元素机器人。

5) 出租车环境: 出租车环境是一个网格世界导航任务，其中代理（一名出租车司机）必须导航到四个固定位置之一接客，然后将乘客运送到另一个四点中的目的地。在每一步中，代理可以向上、下、左或右移动一个格子。该网格世界包含未知大小和位置的障碍物，增加了寻找必要路径的复杂性。此环境编码了出租车的位置、乘客的初始位置及其目的地为一个整数，导致直觉损失如公式 9 中所示，其中 o 是编码后的观察值。

$$intuition\ loss = f(o) \tag{9}$$

像 Swimmer 一样，Taxi 是一个未解决的环境。因此，为了评估我们的框架实现的样本效率提升，我们使用与 Swimmer 环境相同的策略进行评价。我们在现实世界中通过 TurtleBot 和 NVIDIA Jetson Nano 实现了该环境，并将视频演示附在了本文档中。

TABLE II

SHIRE 计算开销（μs/样本）

Environment	Intuition Net Size	Overhead per Sample (μs)
CartPole	2	223
MountainCar	2	215
LunarLander (w/o antiparallelism)	5	254
LunarLander (w/ antiparallelism)	6	257
Swimmer	4	232
Taxi	4	235

B. 样本效率增益

表 I 所示的结果突出了即使是简单直观的想法也能对 RL 性能产生显著影响。SHIRE 在所有测试环境中实现了 $> 25\%$ 的样本效率提升。在 CartPole（我们提供的最简单的环境）中，我们在牺牲 21.25% 更多的实际时间的情况下，达到了 37.5% 的样本效率增益。训练时间变差可以通过 CartPole 环境的简单性来解释，这允许非常快速的模拟，即使像表 II 所示的小开销也会导致更差的实际时间性能。在 MountainCar 环境中，SHIRE 推动代理学习必要的振荡行为，分别获得了 78%和 76%的样本效率和实际时间增益。在 Lunar Lander 中，简单的速度直觉带来了适度的 25%样本效率增益。强制速度向量与飞行器方向向量之间反平行使该增益提升至 41%! 教 Swimmer 代理必要的蛇形运动导致超过 58%的样本效率增益，同时使用如出租车位置和乘客目的地等观察数据鼓励代理朝目标位置移动可以提高 29%的样本效率。我们注意到性能增益与

环境复杂性成正比增加。这合乎直觉，因为对基础原理有更深的理解（编码到我们的直观网络中）有助于学习复杂的任务。这种行为也证明了我们的假设，即 SHIRE 通过教代理基本行为使其策略可解释。样本效率的提升伴随着与直观损失计算相关的微小训练时间开销，这一部分将在下文进一步说明。

C. SHIRE 费用

由直觉损失计算管道引起的延迟开销大约与直觉网络的大小成正比。表 II 显示了所有测试环境中直觉损失计算的开销。很明显，SHIRE 引入的开销相对于每个环境步骤的平均时间来说可以忽略不计，通常为数十毫秒的数量级。由于 SHIRE 的样本效率提升，如表 I 所示的时间墙钟时间增益所示，任何由此引起的训练时间增加都被抵消了。

V. 结论

我们提出了 SHIRE，一个新颖且计算友好的框架，用于将特定任务的人类直觉编码为 PGM 以增强 RL 训练样本效率和可解释性。实验表明，SHIRE 在各种环境中提供了超过 25%的样本效率增益，并具有极小的计算开销，这一开销被改进的效率所抵消。

本工作实现了快速原型设计 RL 策略，便于检查各种策略架构的有效性，从而改善 RL 策略开发生命周期。我们希望这项工作能激发更多关于高效且可解释的 RL 的研究，使得能够为自动驾驶等关键安全任务开发出稳健的 RL 策略。

致谢：本工作得到了 CoCoSys 的支持，它是 JUMP 2.0 中的七个中心之一，而 JUMP 2.0 是由 DARPA 资助的 SRC 计划。

References

[1] A. Z. Zhu, L. Yuan, K. Chaney, and K. Daniilidis, "Unsupervised event-based learning of optical flow, depth, and egomotion," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[2] M. S. Junayed, A. Sadeghzadeh, M. B. Islam, L.-K. Wong, and T. Aydin, "Himode: A hybrid monocular omnidirectional depth estimation model," 2022. [Online]. Available: https://arxiv.org/abs/2204.05007

[3] Z. Shen, C. Lin, K. Liao, L. Nie, Z. Zheng, and Y. Zhao, "Panoformer: Panorama transformer for indoor 360 depth estimation," 2022. [Online]. Available: https://arxiv.org/abs/2203.09283

[4] M. Gehrig, M. Millhäusler, D. Gehrig, and D. Scaramuzza, "E-raft: Dense optical flow from event cameras," in *International Conference on 3D Vision (3DV)*, 2021.

[5] C. Lee, A. K. Kosta, and K. Roy, "Fusion-flownet: Energy-efficient optical flow estimation using sensor fusion and deep fused spiking-analog network architectures," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 6504–6510.

[6] W. Ponghiran, C. M. Liyanagedera, and K. Roy, "Event-based temporally dense optical flow estimation with sequential learning," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 9827–9836.

[7] Z. Huang, X. Shi, C. Zhang, Q. Wang, K. C. Cheung, H. Qin, J. Dai, and H. Li, "Flowformer: A transformer architecture for optical flow," in *Computer Vision – ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23 – 27, 2022, Proceedings, Part XVII*, 2022, p. 668 – 685. [Online]. Available: https://doi.org/10.1007/978-3-031-19790-1_40

[8] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, P. Dollar, and R. Girshick, "Segment anything," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2023, pp. 4015–4026.

[9] A. Kirillov, K. He, R. Girshick, C. Rother, and P. Dollar, "Panoptic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[10] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, and P. Luo, "Segformer: Simple and efficient design for semantic segmentation with transformers," in *Advances in Neural Information Processing Systems*, M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, Eds., vol. 34. Curran Associates, Inc., 2021, pp. 12 077–12 090.

[11] J. Czarnowski, T. Laidlow, R. Clark, and A. J. Davison, "Deepfactors: Real-time probabilistic dense monocular slam," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 721–728, 2020.

[12] A. Joshi, S. Sanyal, and K. Roy, "Real-time neuromorphic navigation: Integrating event-based vision and physics-driven planning on a parrot bebop2 quadrotor," 2024. [Online]. Available: https://arxiv.org/abs/2407.00931

[13] R. Rahman, D. Owen, and J. You, "Tracking large-scale ai models," 2024, accessed: 2024-09-03. [Online]. Available: https://epochai.org/blog/tracking-large-scale-ai-models

[14] T. Gebru, J. Morgenstern, B. Vecchione, J. W. Vaughan, H. Wallach, H. D. Iii, and K. Crawford, "Datasheets for datasets," *Communications of the ACM*, vol. 64, no. 12, pp. 86–92, 2021.

[15] M. T. Ribeiro, S. Singh, and C. Guestrin, "" why should i trust you?" explaining the predictions of any classifier," in *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, 2016, pp. 1135–1144.

[16] O. M. Andrychowicz, B. Baker, M. Chociej, R. Jozefowicz, B. McGrew, J. Pachocki, A. Petron, M. Plappert, G. Powell, A. Ray, *et al.*, "Learning dexterous in-hand manipulation," *The International Journal of Robotics Research*, vol. 39, no. 1, pp. 3–20, 2020.

[17] T. Haarnoja, S. Ha, A. Zhou, J. Tan, G. Tucker, and S. Levine, "Learning to walk via deep reinforcement learning," *Robotics: Science and Systems*, 2019.

[18] I. Osband, D. Russo, and B. Van Roy, "(more) efficient reinforcement learning via posterior sampling," *Advances in Neural Information Processing Systems*, vol. 26, 2013.

[19] M. Mutti, R. D. Santi, M. Restelli, A. Marx, and G. Ramponi, "Exploiting causal graph priors with posterior sampling for reinforcement learning," in *The Twelfth International Conference on Learning Representations*, 2024. [Online]. Available: https://openreview.net/forum?id=M0xK8nPGvt

[20] D. Hafner, T. Lillicrap, J. Ba, and M. Norouzi, "Dream to control: Learning behaviors by latent imagination," in *International Conference on Learning Representations*, 2020. [Online]. Available: https://openreview.net/forum?id=S1lOTC4tDS

[21] Łukasz Kaiser, M. Babaeizadeh, P. Miłos, B. Osiński, R. H. Campbell, K. Czechowski, D. Erhan, C. Finn, P. Kozakowski, S. Levine, A. Mohiuddin, R. Sepassi, G. Tucker, and H. Michalewski, "Model based reinforcement learning for atari," in *International Conference on Learning Representations*, 2020. [Online]. Available: https://openreview.net/forum?id=S1xCPJHtDB

[22] M. Janner, J. Fu, M. Zhang, and S. Levine, "When to trust your model: Model-based policy optimization," *Advances in neural information processing systems*, vol. 32, 2019.

[23] D. G. Myers, "Intuition's powers and perils," *Psychological Inquiry*, vol. 21, no. 4, pp. 371–377, 2010.

[24] B. O'Donoghue, R. Munos, K. Kavukcuoglu, and V. Mnih, "Combining policy gradient and q-learning," *arXiv preprint arXiv:1611.01626*, 2016.

[25] T. Haarnoja, H. Tang, P. Abbeel, and S. Levine, "Reinforcement learning with deep energy-based policies," in *International conference on machine learning*. PMLR, 2017, pp. 1352–1361.

[26] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," in *International conference on machine learning*. PMLR, 2018, pp. 1861–1870.

[27] M. Deisenroth and C. E. Rasmussen, "Pilco: A model-based and data-efficient approach to policy search," in *Proceedings of the 28th International Conference on machine learning (ICML-11)*, 2011, pp. 465–472.

[28] Y. Gal, R. McAllister, and C. E. Rasmussen, "Improving pilco with bayesian neural network dynamics models," in *Data-efficient machine learning workshop, ICML*, vol. 4, no. 34, 2016, p. 25.

[29] K. Chua, R. Calandra, R. McAllister, and S. Levine, "Deep reinforcement learning in a handful of trials using probabilistic dynamics models," *Advances in neural information processing systems*, vol. 31, 2018.

[30] X. Chen, J. Hu, L. Li, and L. Wang, "Efficient reinforcement learning in factored mdps with application to constrained rl," *arXiv preprint arXiv:2008.13319*, 2020.

[31] I. Osband and B. Van Roy, "Near-optimal reinforcement learning in factored mdps," *Advances in Neural Information Processing Systems*, vol. 27, 2014.

[32] M. Mundhenk, J. Goldsmith, C. Lusena, and E. Allender, "Complexity of finite-horizon markov decision process problems," *Journal of the ACM (JACM)*, vol. 47, no. 4, pp. 681–720, 2000.

[33] C. Lusena, J. Goldsmith, and M. Mundhenk, "Nonapproximability results for partially observable markov decision processes," *Journal of artificial intelligence research*, vol. 14, pp. 83–103, 2001.

[34] "Proximal policy optimization," https://stable-baselines3.readthedocs.io/en/master/modules/ppo.html, accessed: 2024-09-14.

[35] "Deep q-network," https://stable-baselines3.readthedocs.io/en/master/modules/dqn.html, accessed: 2024-09-14.

[36] D. Koller and N. Friedman, *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

[37] "Lunar lander environment," https://gymnasium.farama.org/environments/box2d/lunar_lander/, accessed: 2024-09-14.

[38] A. W. Moore, "Efficient memory-based learning for robot control," University of Cambridge, Computer Laboratory, Tech. Rep., 1990.

[39] R. Coulom, "Reinforcement learning using neural networks, with applications to motor control," Ph.D. dissertation, Institut National Polytechnique de Grenoble-INPG, 2002.