doi:10.11887/j.issn.1001-2486.24120032 http://journal.nudt.edu.cn

面向长序列自主作业的非对称 Actor-Critic 强化学习方法

任君凯[1,2],瞿宇珂[1,2*],罗嘉威[1,2],倪子淇[3],卢惠民[1,2],叶益聪[3]
(1. 国防科技大学 智能科学学院,湖南 长沙 410073;2. 装备状态感知与敏捷保障全国重点实验室,湖南 长沙 410073;
3. 国防科技大学 空天科学学院,湖南 长沙 410073)

摘 要:长序列自主作业能力已成为制约智能机器人走向实际应用的问题之一。针对机器人在复杂场景中面临的多样化长序列操作技能需求,提出了一种高效鲁棒的非对称 Actor-Critic 强化学习方法,旨在解决长序列任务学习难度大与奖励函数设计复杂的挑战。通过整合多个 Critic 网络协同训练单一 Actor 网络,并引入生成对抗模仿学习为 Critic 网络生成内在奖励,从而降低长序列任务学习难度。在此基础上,设计两阶段学习方法,利用模仿学习为强化学习提供高质量预训练行为策略,在进一步提高学习效率的同时,增强策略的泛化性能。面向化学实验室长序列自主作业的仿真结果表明,该方法显著提高了机器人长序列操作技能的学习效率与行为策略的鲁棒性。

关键词:自主作业机器人;强化学习;Actor-Critic;长序列操作
中图分类号:TP249 **文献标志码**:A **文章编号**:1001-2486(2025)04-111-12

论文拓展

Asymmetric Actor-Critic reinforcement learning for long-sequence autonomous manipulation

REN Junkai[1,2], *QU Yuke*[1,2*], *LUO Jiawei*[1,2], *NI Ziqi*[3], *LU Huimin*[1,2], *YE Yicong*[3]
(1. College of Intelligence Science and Technology, National University of Defense Technology, Changsha 410073, China;
2. National Key Laboratory of Equipment State Sensing and Smart Support, Changsha 410073, China;
3. College of Aerospace Science and Engineering, National University of Defense Technology, Changsha 410073, China)

Abstract: Long-sequence autonomous manipulation capability becomes one of the bottlenecks hindering the practical application of intelligent robots. To address the diverse long-sequence operation skill requirements faced by robots in complex scenarios, an efficient and robust asymmetric Actor-Critic reinforcement learning method was proposed. This approach aims to solve the challenges of high learning difficulty and complex reward function design in long-sequence tasks. By integrating multiple Critic networks to collaboratively train a single Actor network, and introducing GAIL (generative adversarial imitation learning) to generate intrinsic rewards for the Critic network, the learning difficulty of long-sequence tasks was reduced. On this basis, a two-stage learning method was designed, utilizing imitation learning to provide high-quality pre-trained behavior policies for reinforcement learning, which not only improves learning efficiency but also enhances the generalization performance of the policy. Simulation results for long-sequence autonomous task execution in a chemical laboratory demonstrate that the proposed method significantly improves the learning efficiency of robot long-sequence skills and the robustness of behavior policies.

Keywords: autonomous manipulation robot; reinforcement learning; Actor-Critic; long-sequence operation

随着人工智能与机器人技术的快速发展,自主作业机器人逐渐引起了业界的关注。由于其具有较高的执行效率,不受体力限制且能够承担大量复杂重复、危险度高的任务,有望在医疗[1]、服务、工业生产线[2]等多个领域中解放人力成本,改善人类生活方式与生活质量。然而,在上述典型应用场景中,任务通常由多个环环相扣的步骤组成,要求机器人具备较高的长序列自主作业能力,在精准执行每个子任务的同时,还需拥有全局任务理解、动态调整与环境适应的能力[3-4]。因此,设计智能高效

收稿日期:2024-12-16
基金项目:国家自然科学基金资助项目(62373201);国防科技大学自主创新科学基金资助项目(ZK2023-30,24-ZZCX-GZZ-11)
第一作者:任君凯(1991—),男,河北石家庄人,副教授,博士,硕士生导师,E-mail:jk.ren@nudt.edu.cn
***通信作者**:瞿宇珂(1999—),男,湖南岳阳人,博士研究生,E-mail:quyuke999@163.com
引用格式:任君凯,瞿宇珂,罗嘉威,等. 面向长序列自主作业的非对称 Actor-Critic 强化学习方法[J]. 国防科技大学学报,2025,47(4):111-122.
Citation: REN J K, QU Y K, LUO J W, et al. Asymmetric Actor-Critic reinforcement learning for long-sequence autonomous manipulation[J]. Journal of National University of Defense Technology, 2025, 47(4): 111-122.

的长序列任务决策与行为规划策略,保证自主作业机器人在解决复杂实际任务时的作业效率与可靠性,具有重要的理论和实践意义。

近些年,模仿学习与强化学习在机器人自主作业领域取得了显著进展。在物体抓取与放置[5-6]、开关门[7]和积木堆叠[8]等任务中,机器人已经基本能够实现高精度的自主操作。然而,现有技术在面对长序列自主作业任务时仍面临诸多挑战。一方面,模仿学习依赖于预先采集的示教数据,缺乏有效的探索与泛化机制,因此在面对示教样本分布外的新状态时表现欠佳[9];另一方面,虽然强化学习通过试错学习的方式,在一定程度上能够缓解上述问题,但在学习长序列操作技能时,由于探索空间庞大且缺乏先验知识,训练成本较高[10]。此外,强化学习在解决长时序问题时还面临稀疏奖励等挑战。尽管已有研究采用生成对抗模仿学习(generative adversarial imitation learning, GAIL)[11]来学习内在奖励函数,或利用大语言模型推断环境中的奖励函数[12]来尝试解决这一问题。然而,鉴于技能序列的冗长性与复杂性,设计或学习有效的奖励函数依然十分困难。

为了克服上述挑战,研究者设计了分层学习框架,并且在经典模仿学习与强化学习的基础上提出了分层模仿学习[13]与分层强化学习[14]方法,将长序列任务根据其任务特点与控制频率分解为上下两层进行学习。在下层,长序列自主作业任务被拆解为多个简单的子任务进行学习,机器人能学习到多个操作子技能;在上层,通过设计子技能调度规划器,根据当前状态以及其他相关指标来调度下层子技能,通过组合下层子技能顺序,能够完成原本复杂的长序列自主作业任务,从而降低任务的学习难度,并提高技能的可解释性和可迁移性。然而,分层学习在实际运用过程中,也存在一些挑战:由于包含多个子技能模型,分层学习大大增加了模型复杂度[15];大量的子技能会给上层规划器带来更大的规划压力,甚至导致技能调度错误的情况,尤其是多个子技能对应的状态信息可能具有一定的相似度,容易严重影响技能调度的稳定性[16];每个子技能都是单独训练的,缺乏对技能间连接关系的考虑,可能导致在实际应用中技能之间存在“序列交接”失误的问题[17],进而影响整体任务的成功率。

综上,针对强化学习在长序列自主作业任务中存在的奖励函数设计困难、训练效率低以及技能衔接不稳定等关键问题,结合模仿学习与强化学习的优势,提出一种面向长序列自主作业的非对称 Actor-Critic 强化学习方法。首先,采用任务分解策略将复杂长序列任务拆解为若干具备明确语义的子任务,并为每个子技能分别构建独立的 GAIL 模块与 Critic 网络,以降低全局奖励设计复杂度并提升子技能学习精度;然后,提出一种两阶段训练机制,第一阶段通过模仿学习预训练 Actor 网络以提供合理初始化策略,第二阶段在 GAIL 辅助下进行强化训练以进一步优化全局策略;最后,基于 Unity 引擎构建多类长序列操作任务场景,仿真验证所提方法在复杂自主操作任务中的泛化能力、执行效率与鲁棒性。该方法有效提升了机器人在复杂序列操作中的学习效率与策略稳定性,展现出良好的任务完成能力与实用价值。

1 长序列自主作业任务分析与建模

1.1 长序列自主作业任务分析

在执行长序列自主作业任务时,如图 1 所示,任务的长时性和顺序依赖性凸显,要求机器人执行一系列较长的动作序列,这些动作可归纳为不同的操作行为,如推拉、抓取、放置等。这些操作间存在着“序列交接”问题[17],即前一操作的结束状态直接决定了后一操作的起始条件,对后续操作的成功执行具有决定性影响。因此,机器人不仅需要精准掌握每个独立操作的执行技巧,还必须深入理解并有效学习各操作间的交接过程,以确保整个任务序列的连贯性和最终的成功率。

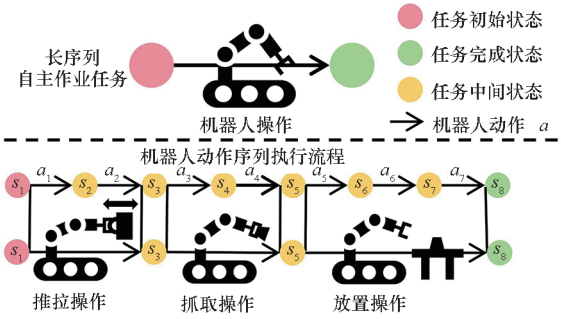


图 1 机器人执行长序列自主作业任务示意图

Fig. 1 Schematic diagram of robot executing long sequence operation tasks

在强化学习的理论框架内,可以将长序列自主作业任务表示为马尔可夫决策过程(Markov decision processes, MDP)。一个典型的 MDP 模型全面涵盖了状态空间、动作空间、奖励函数以及状态转移概率矩阵。值得注意的是,在深度强化学习中,通过训练神经网络能够隐式地学习环境中的状态转移规律,从而替代了显式构建状态转移

矩阵的需求。鉴于此,本节将聚焦于状态空间、动作空间以及奖励函数的设计,这些要素在强化学习系统中扮演着至关重要的角色。

1.2　环境与智能体

在介绍 MDP 的关键要素之前,首先将对环境与智能体进行简要说明。本文的研究场景设定在一个化学实验室中,仿真环境如图 2 所示,旨在学习完成长序列自主作业任务的操作技能。在这个环境中,共有五个可操作的目标物体、两个不可操作的目标物体以及一个智能体。其中,智能体为 Franka 机械臂,通过控制其末端执行器与环境交互来执行一系列操作。环境中的目标物体包括可操作对象(抽屉把手、柜门把手、试管架、盖子)与不可操作对象(试管架平台、盖子平台)。

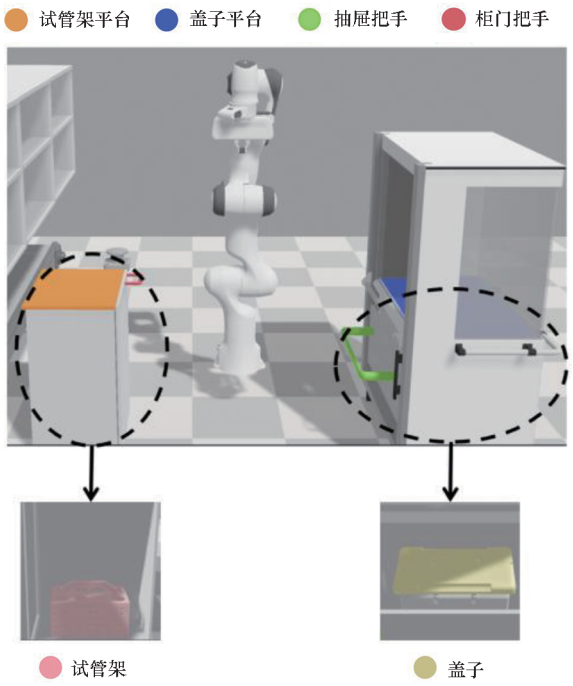


图 2　化学实验室仿真环境[16]

Fig. 2　Chemical laboratory simulation environment[16]

为了验证学习效果,设计了两种不同的长序列任务,在每个任务中智能体需要交互的对象不同,与对象的交互逻辑也有所差异。具体任务细节如下:

任务一:智能体需要打开抽屉,取出抽屉里的盖子,并将其放置于相应的平台上,最后末端执行器需要回到复位点。若抽屉未打开,智能体将无法取出盖子。

任务二:智能体需要打开绕单轴旋转的柜门,从柜子中取出试管架,并将其放置在指定的试管架平台上,最后末端执行器需要回到复位点。若柜门未打开,智能体将无法取出试管架。

实验室中两类长序列操作任务均需严格满足“前序操作成功→后续动作可行”的依赖关系(如抽屉未完全开启则无法抓取内部物体),能够有效模拟真实场景中因操作失误引发的“序列交接”问题[17]。在操作模式上,两类任务分别包含了线性平移与旋转运动的差异化操作需求,体现了实验室常见操作模式。在方法验证方面,两个任务均要求智能体完成“打开容器→取出物体→放置→复位”的完整流程,覆盖了抓取、移动、状态判断等能力,这种设计能够检验方法在组合性动作序列中的适应性,同时为后续扩展至其他类似长序列任务提供了基础验证。

1.3　状态空间设计

智能体的状态空间 S 包含了三类信息:智能体的本体信息 G、目标物体信息 M 以及部分特定的环境信息 F。智能体的本体信息 G 主要通过以下两个方面进行描述:末端执行器的速度 V,以及末端夹爪的开合状态 K。对于目标物体信息 M,采用相对于末端执行器的极坐标 m 进行表示,如图 3 所示。采用了一种特殊的表示方法:相对极坐标 m 中的角度 θ 用目标物体与末端执行器之间的归一化相对位置向量 $\boldsymbol{e}_{\text{norm}}=(p_x,p_y,p_z)$ 来替代,而极坐标的半径 ρ 则代表目标物体与末端执行器之间的距离。环境信息则包含一些难以通过相对位置量化的信息,例如抽屉的开启程度 d_{norm}(通过距离归一化值表示)、柜门的开启程度 θ_{norm}(通过角度归一化表示)以及额外的环境信息 x_{extra}。表 1 展现了每个任务的状态空间组成。

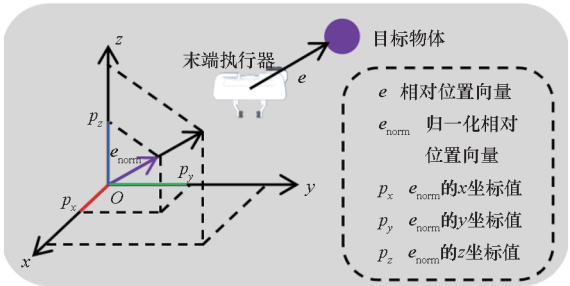


图 3　极坐标 m 中角度 θ 的特殊表示方法示意图

Fig. 3　Schematic diagram of special representation method for angle θ in polar coordinate m

$$S=\{G,M,F\} \tag{1}$$

$$G=\{\boldsymbol{V},\ K\} \tag{2}$$

$$M=\{\boldsymbol{m}_1,\boldsymbol{m}_2,\cdots,\boldsymbol{m}_n\} \tag{3}$$

$$\boldsymbol{m}_i=(p_x,p_y,p_z,\rho) \tag{4}$$

$$F=\{d_{\text{norm}},\theta_{\text{norm}},\ x_{\text{extra}}\} \tag{5}$$

表 1 每个任务的状态空间组成

Tab. 1 Composition of state space for each task

状态空间	任务 1	任务 2
智能体本体信息	末端执行器的速度和末端夹爪的开合状态	末端执行器的速度和末端夹爪的开合状态
目标物体信息	抽屉把手、盖子和盖子平台相对于末端执行器的极坐标	柜门把手、试管架和试管架平台相对于末端执行器的极坐标
环境信息	复位点相对于末端执行器的极坐标，抽屉的开启程度	复位点相对于末端执行器的极坐标，柜门的开启程度

1.4 动作空间设计

采用四维连续动作空间 $A=\{\alpha_1,\alpha_2,\alpha_3,\alpha_4\}$，具体设计如图 4 所示，包含两个方面：一是末端执行器的加速度，由 α_1（对应 $a_{x\pm}$）、α_2（对应 $a_{y\pm}$）和 α_3（对应 $a_{z\pm}$）三个参数表示；二是末端执行器夹爪的开合，该状态由 a_4（对应 x_g）来控制。由于夹爪的控制界面有限，它只提供打开和关闭的二进制命令。为了弥补这一点，引入一个函数 $f(x_g)$ 来解释夹爪状态的切换过程。这种关系显示为：

$$f(x_g)=\begin{cases}\text{打开} & x_g\in(0.9,1.0]\\ \text{保持现状} & x_g\in[-0.9,0.9]\\ \text{关闭} & x_g\in[-1.0,-0.9)\end{cases}\tag{6}$$

当 $x_g\in[-0.9,0.9]$ 时，夹爪保持当前状态；当且仅当 x_g 超出 $[-0.9,0.9]$ 的区域时，夹持器的状态将切换为打开或关闭。

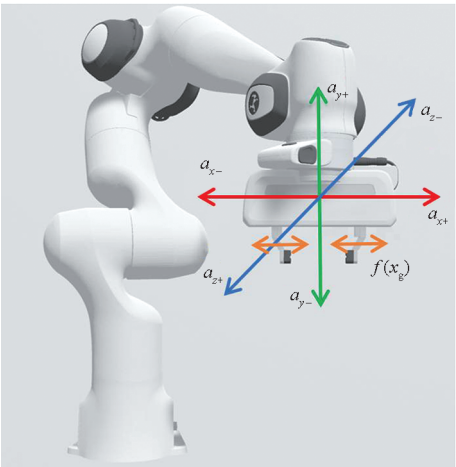


图 4 智能体的四维连续动作空间

Fig. 4 The four-dimensional continuous action space of the agent

1.5 奖励函数设计

在强化学习中，奖励函数的设计是否合理，会直接影响智能体学习到的行为策略。通过设计能够准确反映任务本质的奖励函数，不仅能显著增进学习效果，还能提升任务完成的成功率。如式(7)所示，奖励函数 R 包括内在奖励 r_i 和外在奖励 r_e 两个部分。

$$R=r_i+r_e\tag{7}$$

式中，内在奖励 r_i 由 GAIL 模块产生。GAIL 模块包括鉴别器网络和执行器网络，通过演示数据集中的演示轨迹和执行器网络生成的实际轨迹来训练鉴别器网络，鉴别器网络作为判别模型，通过比较演示轨迹（标注为真，$D(s)=1$）与智能体轨迹（标注为伪，$D(s)=0$）构建对抗训练损失值。鉴别器网络对智能体轨迹的判别得分越高，表示智能体轨迹与演示轨迹越相似。内在奖励是根据鉴别器网络的分数产生的，即：

$$r_i=-\ln[1-D(s_t)]\tag{8}$$

外在奖励 r_e 来自智能体与环境的交互，每个长序列自主作业任务会被划分为不同的子任务进行学习，针对子任务设计不同的奖励函数，奖励函数设置如式(9)所示。r_{step} 是单步奖励，智能体的每步决策都会得到单步奖励 r_{step}，r_{step} 根据子任务完成程度的提升而增加；$r_{success}$ 是成功奖励，智能体在完成子任务时会得到成功奖励 $r_{success}$。

$$r_e=\begin{cases}r_{step} & \text{随着子任务进程增加}\\ r_{success} & \text{当子任务完成时}\end{cases}\tag{9}$$

2 方法框架

如图 5 所示，本文设计了一种两阶段训练方法来帮助智能体优化 Actor 网络。在首个阶段，利用模仿学习方法来初始化 Actor 网络的权重参数，旨在为后续的强化学习过程奠定一个良好的起点。进入第二阶段，采用了近端策略优化（proximal policy optimization，PPO）算法对 Actor 网络进行训练。为了应对长序列自主作业任务在训练过程中存在的挑战，设计了非对称 Actor-Critic 强化学习方法。通过上述方法，智能体最终能够成功且稳定地完成多个长序列自主作业任务。

2.1 模仿学习

为了减少强化学习过程中的无效探索，加速智能体的学习，引入模仿学习，为智能体 Actor 网络注入先验知识。采用人工示教的方式进行模仿学习数据采集，专家操作者使用键盘控制智能体

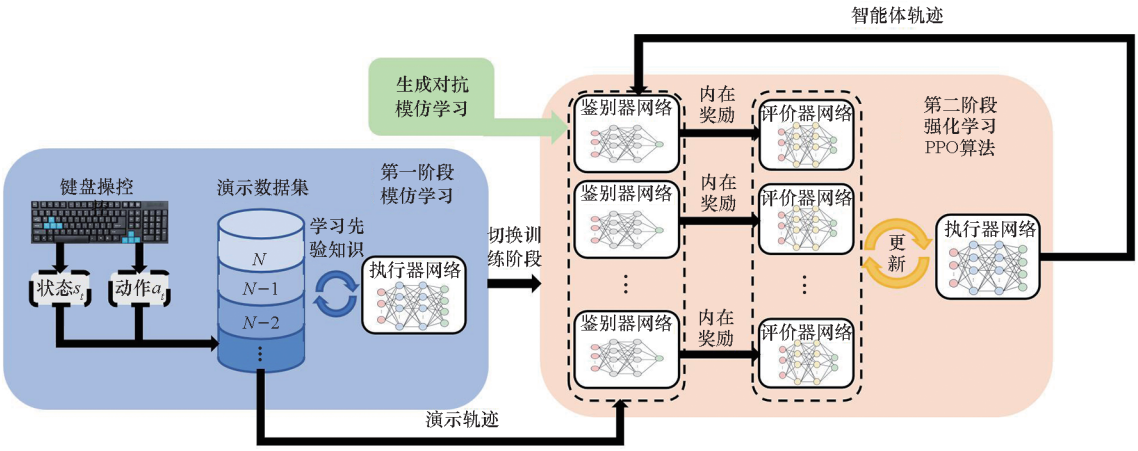


图 5　两阶段学习方法

Fig. 5　Two stage learning method

完成长序列自主作业任务,系统以 25 Hz 采样频率同步记录智能体的状态 s_t 和动作 a_t,共采集 20 个完整成功回合的轨迹数据。利用这些示教过程中产生的状态 s_t 与动作 a_t 序列,构建出用于模仿学习的演示数据集。在模仿学习的阶段,通过最小化演示动作与 Actor 网络输出动作之间的均方误差损失,来实现 Actor 网络对先验知识的学习。值得注意的是,在训练过程中会避免只使用模仿学习。原因在于,本文方法的演示数据集相对较小,这可能导致策略模型迅速过拟合,一个过拟合的模型并不是一个理想的学习起点。因此,在模仿学习的同时,也会进行强化学习,以确保网络的潜在的泛化性。此外,在 PPO 算法中 Actor 网络会输出动作所遵循的高斯分布的均值 μ 和方差 σ,根据这个高斯分布来选择执行的动作 a_t。然而,如果将该动作 a_t 用于模仿学习,会导致方差 σ 迅速下降,从而降低 Actor 网络的贪婪程度。这不利于训练初期智能体对环境的探索。因此,仅采用动作均值 μ 来进行模仿学习。

2.2　近端策略优化算法

PPO 算法是一种基于策略梯度的强化学习算法,通过引入一种简单而有效的裁剪机制,限制了 Actor 网络更新步长的大小,从而避免了因大幅度更新而导致的训练不稳定问题。PPO 算法在 Actor-Critic 框架下初始化两个神经网络——Actor 网络和 Critic 网络。Actor 网络负责根据当前状态输出动作概率分布,而 Critic 网络则评估当前状态的价值。算法利用 Critic 网络的输出值(即状态价值预估)与实际的奖励信息,进一步计算出优势函数 $\bar{A}_t$,该函数量化了采取特定动作相较于平均行为所展现出的优势程度。为了确保更新后的新 Actor 网络不过度偏离旧 Actor 网络,算法通过裁剪概率比 $r_t(\theta)$ 对 Actor 网络更新幅度进行控制,其数学表示为:

$$r_t(\theta)=\frac{\pi_\theta(a_t|s_t)}{\pi_{\theta\text{old}}(a_t|s_t)} \tag{10}$$

$$L(\theta)=-\frac{1}{T}\sum_{t=1}^{T}\min\{r_t(\theta)\bar{A}_t,C[r_t(\theta),1-\varepsilon,1+\varepsilon]\bar{A}_t\} \tag{11}$$

其中:θ 代表 Actor 网络参数;$r_t(\theta)$ 代表新旧 Actor 网络在状态 s_t 下选择动作 a_t 的概率比值;$L(\theta)$ 是 Actor 网络的损失函数;C 是一个裁剪函数,用于将 $r_t(\theta)$ 的值限制在$[1-\varepsilon,1+\varepsilon]$的范围内,$\varepsilon$ 是一个人为设定的超参数。简单来说,当新旧 Actor 网络之间的差异超出预设阈值时,裁剪机制将介入调整,以确保 Actor 网络更新维持在一个合理的幅度范围之内。

2.3　非对称 Actor-Critic 强化学习

为了提高长序列操作任务的学习效率,提出了一种非对称 Actor-Critic 强化学习方法,使用多个 GAIL 模型和 Critic 共同训练一个 Actor 网络,具体结构如图 6 所示。

2.3.1　算法结构

将复杂的长序列任务拆解为多个简单的子任务,以此降低长序列任务学习的难度。针对每个拆解后的子任务,独立设计了奖励函数,并分配了特定的 Critic 网络来学习各子任务的状态价值函数。这些 Critic 网络均具备相同的网络架构,包括一层标准化层(normalize)、三层隐藏层(每层包含 256 个神经元)和一层用于输出价值函数的输出层。所有 Critic 网络共同训练一个 Actor 网络,Actor 网络的网络架构跟 Critic 网络大致一样,区别在于最后一层为四维的动作均值 μ 输出层,动作方差 σ 由一个可学习的四维参数进行表

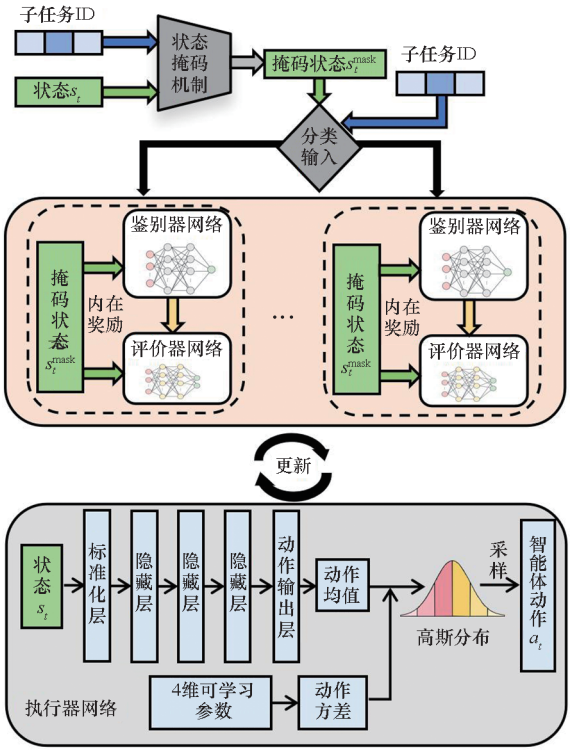


图 6 非对称 Actor-Critic 结构

Fig. 6 Asymmetric Actor-Critic structure

示。同时,为了进一步提升学习效果,每个子任务都配置了独立的 GAIL 模块,用于生成子任务的内在奖励。每个 GAIL 模块中的鉴别器网络也采用了相同的网络结构,包括一层标准化层、两层隐藏层(每层各含 128 个神经元)和一层负责输出对当前状态评估值的输出层。GAIL 模块中的鉴别器网络在训练过程中采用与第一阶段模仿学习中一样的数据集。在训练子任务时,所有 GAIL 模块共用同一执行器,智能体通过该执行器在不同子任务环境中采集实际交互轨迹。数据集中的演示轨迹按子任务类别进行分割,并将分割后的演示轨迹与智能体在对应子任务中采集的实际轨迹共同输入该子任务对应的 GAIL 模块。

Actor 网络状态输入为长序列任务中所有的状态信息 s_t,而在构建 Critic 网络和 GAIL 模块的输入时,本文方法设计了一种状态掩码机制。该机制根据子任务 ID,选择性地排除与当前子任务无关的状态信息。具体来说,掩码机制基于子任务所涉及的目标物体来判断状态与任务的相关性,筛选有效状态信息。例如,任务一被分解为“打开抽屉”和“取出盖子并复位”两个子任务,在子任务 1(仅涉及智能体与柜子交互)中,掩码状态 s_t^{mask} 只包含表 1 中的智能体本体信息,即抽屉把手相对于末端执行器的极坐标和抽屉的开启程度。那么相应的子任务 2 中的掩码状态 s_t^{mask} 只包含智能体本体信息,即盖子和盖子平台相对于末端执行器的极坐标,以及复位点相对于末端执行器的极坐标。通过预定义的任务相关性规则,掩码机制丢弃无关状态维度,最终输出与当前子任务匹配的掩码状态 s_t^{mask},根据子任务 ID 将掩码状态 s_t^{mask} 输入相应的 Critic 网络和 GAIL 模块。基于上述操作,Critic 网络和 GAIL 模块能够专注于与当前子任务高度关联的状态空间,从而提高学习的准确性和效率。

2.3.2 网络微调

一旦完成各个子任务的学习阶段,Actor 网络便成功掌握了多个子技能。为了习得这些子技能间的平滑过渡行为,需对 Actor 网络实施进一步微调。具体而言,在非对称 Actor-Critic 强化学习方法中加入针对长序列自主作业任务的 Critic 网络和 GAIL 模块,用于训练 Actor 网络。鉴于 Actor 网络已具备各子技能的知识基础,无须构建高度复杂的奖励函数,仅需设计简洁的引导机制,促使 Actor 网络能够自主地将离散化的子技能融合为连贯的长序列操作技能。具体的奖励函数设计细节参见式(12)与式(13),其中 w_i 代表每个子任务完成前后的奖励值。

$$r_e = \begin{cases} r_{\text{step}} = w_1^{\text{step}} + w_2^{\text{step}} + \cdots + w_n^{\text{step}} \\ r_{\text{success}} \end{cases} \tag{12}$$

$$w_i^{\text{step}} = \begin{cases} 0 & \text{子任务 } i \text{ 没有完成} \\ \alpha & \text{子任务 } i \text{ 已经完成} \end{cases} \tag{13}$$

在推进长序列任务学习的同时,持续维护对各个子任务的学习,旨在防止 Actor 网络在专注于长序列任务学习时,可能会对已掌握的知识造成潜在破坏。

3 实验与分析

为了验证本文算法的有效性,本节首先在 Unity 仿真平台上设计了如 1.2 节所描述的操作任务,并与文献[16]中的下层专家技能学习方法(后续简称为 Roman-Expert[16])、GAIL 算法[18](PPO 算法 + 内在奖励)、PPO 算法[19](仅外在奖励)进行对比。Roman-Expert 是一种端到端的操作技能学习方法,其主体网络为传统的 Actor-Critic 架构。然后,对第一阶段的模仿学习和第二阶段的状态掩码机制展开消融实验,进一步验证了本文提出的两阶段学习方法和状态掩码机制的优势。最后,将本文方法与现有的分层学习算法进行对比与讨论。

3.1　实验参数设置

仿真环境中物理更新频率为 1 000 Hz,相应的 Franka 机械臂的控制频率为 1 000 Hz,智能体决策频率为 25 Hz,即每 0.04 s 就会进行一次动作决策,并且在两次决策之间,智能体会沿用上一次决策的动作进行控制。为了实验的公开性,表 2 为算法的主要超参数。其中:Beta 决定了算法的"贪婪程度",Beta 越大,动作方差下降得越慢或上升得越快;Epsilon 决定了新旧 Actor 网络之间更新的步长上限,等价于 2.2 节中的参数 ε;Lambda 为广义优势估计(generalized advantage estimation, GAE[20])中的正则化参数;Buffer_size 为在更新网络前每个子任务应采集的样本量;Batch_size 为更新网络时每个子任务的样本批次。

表 2　算法超参数设置

Tab. 2　Hyperparameter settings for algorithm

参数	值	参数	值
学习率	3×10^{-4}	Lambda	0.95
Beta	1×10^{-2}	Buffer_size	10 240
Epsilon	2×10^{-1}	Batch_size	1 024

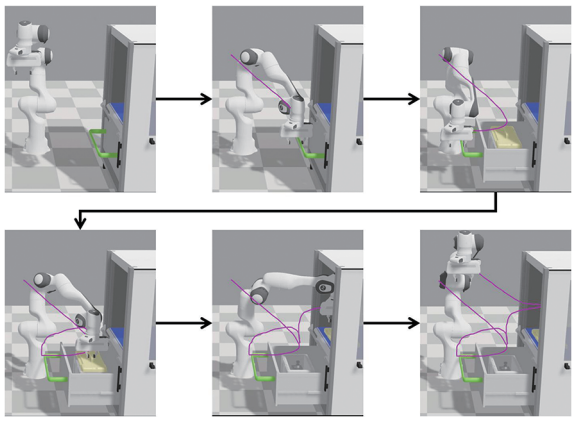
3.2　对比实验

3.2.1　关于运动轨迹的对比实验

任务一和任务二的具体描述如 1.2 节所示。基于非对称 Actor-Critic 结构,任务一可被拆分成两个子任务进行学习:拉动抽屉——智能体拉开关闭着的抽屉;抓取物体并放置——智能体从抽屉抓取盖子,将其放在相应的平台上后复位末端执行器。基于非对称 Actor-Critic 结构,任务二同样可被拆分成两个子任务进行学习:开启柜门——智能体打开绕着单轴旋转的柜门;抓取物体并放置——智能体从柜子里抓取试管架,将其放在柜子上后复位末端执行器。

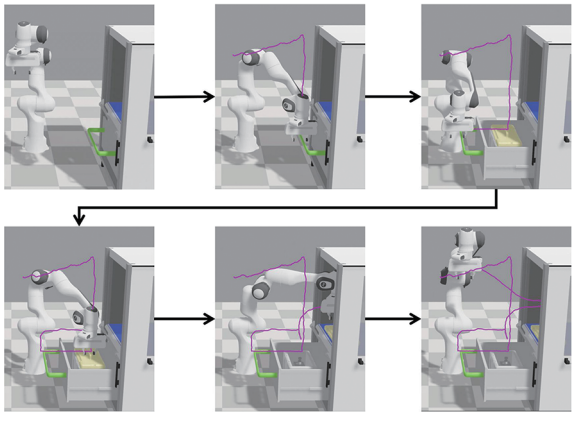
图 7 和图 8 是在不同训练方法下,智能体完成长序列自主作业任务的过程,其中紫色曲线为末端执行器的轨迹。可以看出,本文方法训练出的智能体,其末端执行器的运动轨迹更加平滑。

图 9 和图 10 是在不同训练方法下,Actor 网络动作不确定性的熵值,熵值越大代表动作方差 σ 越大。值得注意的是,在任务一中,子任务学习阶段完成后,Actor 网络需要进行参数微调,才能学会子技能之间的过渡行为。而在任务二中,Actor 网络即使不经过参数微调,也顺利掌握了子技能之间的过渡行为。因此,任务一的完整训练步数比任务二多 6×10^6 步。从图 9 和图 10 可以明显发现,本文方法的动作熵值更低,这说明 Actor 网络的动作方差较小,进一步解释了末端执行器的运动轨迹更加平滑。



(a) 本文方法的运动轨迹

(a) Motion trajectory of our method

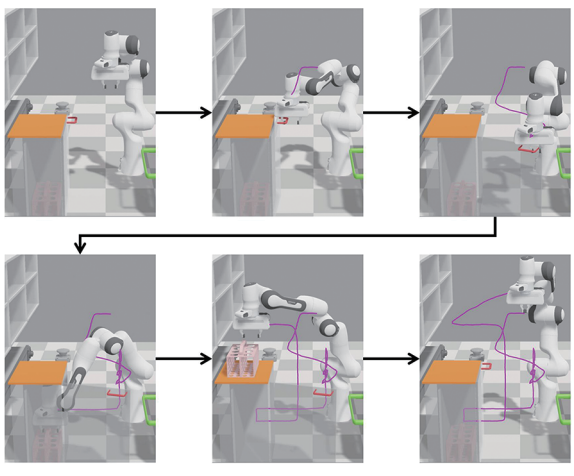


(b) Roman-Expert[16] 的运动轨迹

(b) Motion trajectory of Roman-Expert[16]

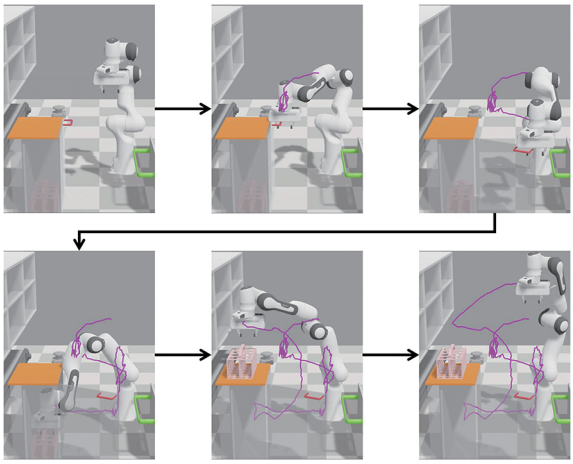
图 7　任务一中不同训练方法的运动轨迹对比

Fig. 7　Comparison of motion trajectory of different training methods in task 1



(a) 本文方法的运动轨迹

(a) Motion trajectory of our method



（b）Roman-Expert [16] 的运动轨迹

（b）Motion trajectory of Roman-Expert [16]

图 8　任务二中不同训练方法的运动轨迹对比

Fig. 8　Comparison of motion trajectory of different training methods in task 2

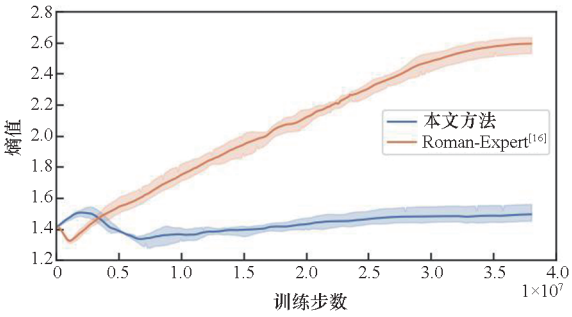


图 9　任务一中不同方法训练过程的 Actor 网络熵值曲线

Fig. 9　Entropy curves of Actor networks trained using different methods in task 1

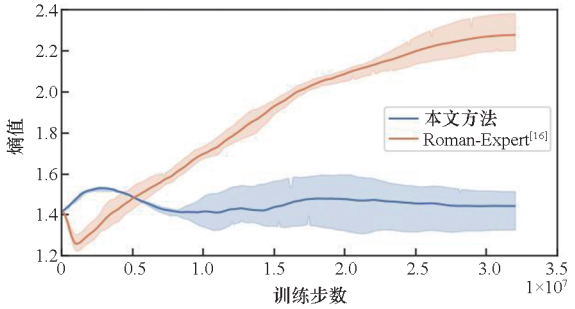


图 10　任务二中不同方法训练过程的 Actor 网络熵值曲线

Fig. 10　Entropy curves of Actor networks trained using different methods in task 2

3.2.2　关于任务完成情况的对比实验

表 3 和表 4 描述了不同方法下的任务成功率和平均回合步数，表中每个数据表示进行了 200 次实验的平均结果。回合步数为智能体在一个回合内的决策步数，在一个回合中智能体的决策步数到达上限或者智能体完成子任务时，该回合将会结束，因此回合步数越低代表任务完成越快，训练效果越好。β 代表位置噪声，通过高斯分布拟合出一种动态变化的偏移量施加在物体的实际位置上，β 的绝对值越大，表示物体实际位置越不准确，状态空间中的相对极坐标也越不准确。GAIL 只采用了内在奖励，没有完成时的奖励，无法学习固定复位到某个位置，因此在测试时增大了 GAIL 的复位阈值，即末端位置和复位点位置距离小于 0.3 m 视为成功，其他方法需要上述距离小于0.1 m才视为成功。从表 3 ~4 中可看出，本文方法保持着最高的成功率，PPO 算法在没有

表 3　任务一中各类方法在不同位置噪声 β 影响下的性能对比

Tab. 3　Performance comparison of various methods in task 1 under different position noise β

对比方法	β = ±0.5 cm		β = ±1.0 cm		β = ±2.0 cm	
	成功率/%	平均回合步数	成功率/%	平均回合步数	成功率/%	平均回合步数
本文方法	**99.5**	**745**	**100.0**	**732**	**100.0**	**724**
Roman-Expert [16]	**99.5**	885	**100.0**	853	**100.0**	876
GAIL[18]	89.5	2 491	94.5	2 335	83.5	2 752
PPO[19]	0	4 000	0	0	0	0

对比方法	β = ±3.0 cm		β = ±4.0 cm		β = ±5.0 cm	
	成功率/%	平均回合步数	成功率/%	平均回合步数	成功率/%	平均回合步数
本文方法	**100.0**	**724**	**98.0**	**801**	**95.0**	**921**
Roman-Expert[16]	99.5	914	97.0	1 125	85.0	1 629
GAIL[18]	67.0	3 026	24.5	3 700	4.5	3 959
PPO[19]	0	0	0	0	0	0

表 4　任务二中对比方法在不同位置噪声 β 影响下的性能对比

Tab. 4　Performance comparison of various methods in task 2 under different position noise β

对比方法	$\beta=\pm0.5$ cm		$\beta=\pm1.0$ cm		$\beta=\pm2.0$ cm	
	成功率/%	平均回合步数	成功率/%	平均回合步数	成功率/%	平均回合步数
本文方法	**98.5**	**1 378**	**98.0**	**1 373**	**97.5**	**1 434**
Roman-Expert[16]	81.5	2 259	81.0	2 303	81.5	2 278
GAIL[18]	44.0	3 522	39.5	3 569	43.5	3 547
PPO[19]	0	4 000	0	0	0	0

对比方法	$\beta=\pm3.0$ cm		$\beta=\pm4.0$ cm		$\beta=\pm5.0$ cm	
	成功率/%	平均回合步数	成功率/%	平均回合步数	成功率/%	平均回合步数
本文方法	**96.0**	**1 525**	**92.5**	**1 556**	**94.0**	**1 557**
Roman-Expert[16]	82.5	2 210	84.0	2 189	82.5	2 270
GAIL[18]	44.0	3 484	44.5	3 486	35.5	3 601
PPO[19]	0	0	0	0	0	0

内部奖励的情况下,没有学习到完整的操作。随着位置噪声 β 的增大,本文方法仍然能保持较高的成功率。并且在相近成功率的情况下,本文方法也能以较短的步数完成任务。由此可见本文方法能够高效鲁棒地完成长序列自主作业任务。

3.3　消融实验

图 11 和图 12 分别展示了任务一和任务二中关于模仿学习的消融实验结果。曲线阴影表示同时学习多个子任务时的平均奖励曲线,奖励越大代表训练效果越好。可以看出,有模仿学习的平均奖励曲线优于没有模仿学习的平均奖励曲线,证明了两阶段的学习方法可以提高算法的收敛速度。图 13和图 14 分别展现了任务一和任务二中关于状态掩码机制的消融实验结果。可以看出,有状态掩码机制的平均奖励曲线皆优于没有状态掩码机制的平均奖励曲线,证明了状态掩码机制通过排除无关的状态信息有效提高了智能体的学习效率。

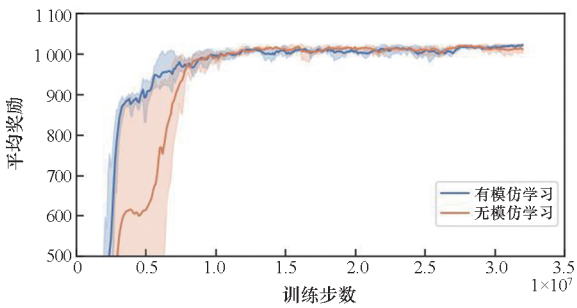


图 11　任务一中关于模仿学习的消融实验结果

Fig. 11　Ablation experiment result on imitation learning in task 1

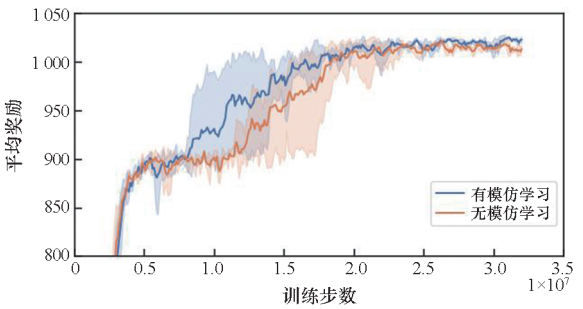


图 12　任务二中关于模仿学习的消融实验结果

Fig. 12　Ablation experiment result on imitation learning in task 2

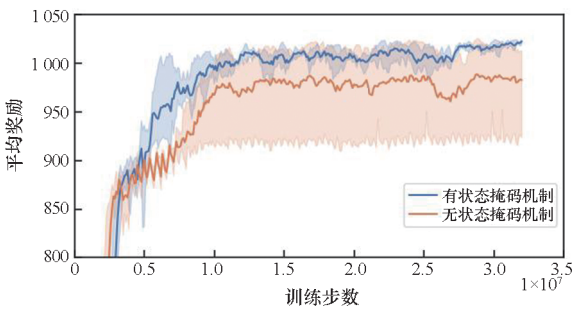


图 13　任务一中关于状态掩码机制的消融实验结果

Fig. 13　Ablation experiment result on state mask mechanism in task 1

3.4　泛化性实验

针对机器人在多种环境初始状态下的策略泛化性能开展了验证实验。结果如表 5 和表 6 所示,当抽屉或柜门开启程度变化时,本文方法仍能保持较高成功率。具体来说,在任务一中本文方法的成功率在抽屉开启程度小于 75% 时保持稳

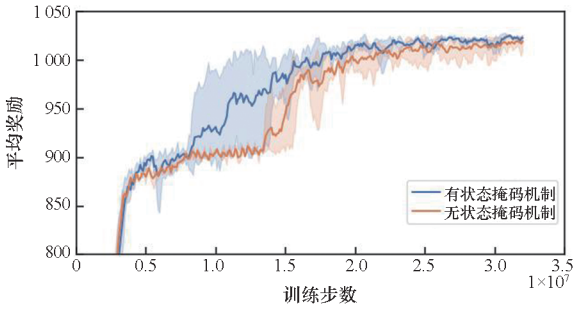


图 14 任务二中关于状态掩码机制的消融实验结果

Fig. 14 Ablation experiment result on state mask mechanism in task 2

定,当抽屉开启程度达到[75%,85%]区间时下降至 74.0%。此现象源于智能体误判当前开启程度已满足抓取条件,因而没有执行打开操作,但未完全开启的抽屉将导致后续抓取成功率降低。在任务二中,本文方法一直保持着高成功率。因此,本文方法对环境初始状态变化具有良好的泛化能力。

表 5 本文方法在任务一中面对不同环境初始状态时的成功率

Tab. 5 Success rate of our method in task 1 when facing different initial states of the environment

抽屉开启程度/%	任务一成功率/%	抽屉开启程度/%	任务一成功率/%
[0,25)	99.5	[55,65)	100.0
[25,35)	100.0	[65,75)	96.0
[35,45)	99.0	[75,85]	74.0
[45,55)	100.0		

3.5 与分层学习的对比分析

本节将系统性地对比与分析现有分层学习方法与本文方法。基于第 3.2.2 节中仅依靠外在奖励无法有效训练操作技能的实验结论,参考所对比的分层强化学习方法[21]中的网络架构与任务分解设计,将总任务分解为两个下层技能模型与一个上层规划模型,并采用 PPO + GAIL 算法进行训练以确保训练效果。在任务一中,本文方法在微调阶段进行了 6×10^6 步训练,而在任务二中则未进行微调训练。为降低训练步数差异对结果的影响,分层学习方法在下层技能模型训练中分配的总步数高于本文方法中的子任务训练总步数,优先保障底层技能的执行可靠性,从而有效缓解因下层技能执行失败对整体任务成功率的负面影响。在训练分层学习方法的上层规划模型时,分别设置了 6×10^6 步和 12×10^6 步两组对比实验。

表 6 本文方法在任务二中面对不同环境初始状态时的成功率

Tab. 6 Success rate of our method in task 2 when facing different initial states of the environment

柜门开启程度/%	任务二成功率/%	柜门开启程度/%	任务二成功率/%
[0,10)	98.5	[40,50)	98.5
[10,20)	99.5	[50,60)	97.5
[20,30)	99.0	[60,70]	99.0
[30,40)	98.0		

实验结果如表 7 ~9 所示。在任务一中,本文方法在与不同训练步数的分层学习方法的对比下均保持了更高的成功率,在任务一中随着训练步数的增多,分层学习方法的成功率反而降低。在任务二中,本文方法相比于 6×10^6 步训练的分层方法在大部分噪声程度下具有更高的成功率,与 12×10^6 步训练的分层方法相比成功率虽略低但在 $|\beta|\leqslant3.0$ 时基本持平。在计算成本评估方面,本研究通过对比不同模型决策频率对 Unity 编辑器渲染帧率的影响,验证方法效能。实验数据表明,本文方法在不同决策频率下均维持更高渲染帧率,尤其在 1 000 Hz 高频决策时仍保持 455.7 帧/s 的平均帧率,相较之下分层学习方法出现显著性能衰减(345.1 帧/s),从上述数据可以看出本文方法具有更小的计算成本。

表 7 任务一中本文方法和分层学习方法在不同位置噪声 β 影响下的成功率对比

Tab. 7 Success rate comparison of the proposed method and hierarchical learning method in task 1 under different position noise levels β

%

对比方法	$\beta=\pm0.5$ cm	$\beta=\pm1.0$ cm	$\beta=\pm2.0$ cm	$\beta=\pm3.0$ cm	$\beta=\pm4.0$ cm	$\beta=\pm5.0$ cm
本文方法(微调 6×10^6 步)	**99.5**	**100.0**	**100.0**	**100.0**	**98.0**	**95.0**
分层学习方法[21](6×10^6 步)	89.5	84.0	83.0	69.5	62.5	53.0
分层学习方法[21](12×10^6 步)	81.5	81.0	76.0	0.64	57.0	51.0

表 8　任务二中本文方法和分层学习方法在不同位置噪声 β 影响下的成功率对比

Tab. 8　Success rate comparison of the proposed method and hierarchical learning method in task 1 under different position noise levels β

%

对比方法	$\beta=\pm0.5$ cm	$\beta=\pm1.0$ cm	$\beta=\pm2.0$ cm	$\beta=\pm3.0$ cm	$\beta=\pm4.0$ cm	$\beta=\pm5.0$ cm
本文方法(没微调)	98.5	**98.0**	97.5	96.0	92.5	94.0
分层学习方法[21](6×10^6 步)	96.0	91.5	97.0	97.5	95.5	90.5
分层学习方法[21](126×10^6 步)	**100.0**	**98.0**	**99.5**	**98.0**	**99.5**	**99.5**

表 9　任务一中本文方法和分层学习方法在不同决策频率下渲染帧率的比较

Tab. 9　Comparison of rendering frame rates between the proposed method and hierarchical learning method under different decision frequencies in task 1

单位:帧/s

对比方法	不同决策频率/Hz						
	10	20	25	40	100	200	1 000
本文方法	**552.8**	**550.3**	**550.2**	**538.6**	**535.6**	**518.8**	**455.7**
分层学习方法[21]	537.5	534.2	535.2	523.4	522.3	513.2	345.1

值得注意的是,本文方法的核心目标是学习端到端的长序列操作技能。基于分层学习方法的任务分解范式,本文方法要求单一模型同时掌握两种不同子技能(对应于分层中的下层技能模型)以及任务状态判断能力(对应于分层中的上层规划模型)。虽然分层学习方法通过任务解耦降低了学习难度,但上下层的分离可能导致状态评估不全面,这在任务一中表现为短暂学习难以获得优秀的上层任务规划模型。而在任务二中,本文方法无须微调即可直接应用,因此分层方法通过短期训练即可获得优秀的上层规划器。本文方法在评价器层面进行解耦,在执行器层面保持耦合,在降低学习难度的同时确保了状态评估的全面性。

此外,本文方法也能为未来的分层学习提供帮助:首先,本文方法能够提供长序列操作技能,减少上层规划所需调度的技能数量,从而降低学习难度;其次,针对上层规划难以学习的技能间调度衔接问题,可通过本方法将本来需要衔接的两个技能进行耦合学习,形成新的长序列技能供上层规划模型调用,从而减轻上层规划模型的负担。因此,本文方法与分层学习方法在未来研究中可相互补充,共同提升学习效果与表现。

4　结论

本文聚焦于机器人自主作业任务中面临的长序列技能学习挑战,创新性地提出了一种非对称 Actor-Critic 强化学习方法。该方法将冗长复杂的长序列自主作业任务分解为多个简洁易处理的子任务进行学习,并且每个子任务均配备独立的 GAIL 模块和 Critic 网络。该方法显著降低了 GAIL 模块和 Critic 网络的学习难度,使 Critic 网络能为 Actor 网络提供更有针对性的评估与反馈。此外,通过任务分解简化了奖励函数的设计难度。在此基础上,为进一步提升长序列技能学习效率,设计了两阶段学习方法。通过模仿学习为后续强化学习提供了优质的预训练行为策略,加速了整体学习效率。在实验部分,本文验证了该方法在多个长序列任务下的有效性,将其性能与经典 Actor-Critic 深度强化学习算法、分层学习方法进行了对比分析。结果表明,本文方法能够高效学习长序列自主作业行为策略,在任务成功率、执行效率以及鲁棒性等方面展现出显著优势。

本文方法也存在一定的局限性。首先,微调策略需要人工判断是否实施,这不仅增加了操作复杂性,影响方法的自动化程度,还可能引入主观偏差;此外,实验任务种类有限,未能覆盖更广泛的任务场景。未来研究可开发自动化微调策略,提升方法的智能化水平,并在更多样化的任务类型和复杂场景中验证方法的通用性,以进一步拓展其应用范围。

参考文献(References)

[1]　张辉, 王耀南, 易俊飞, 等. 面向重大疫情应急防控的智能机器人系统研究[J]. 中国科学: 信息科学, 2020, 50(7): 1069 – 1090.
ZHANG H, WANG Y N, YI J F, et al. Research on

intelligent robot systems for emergency prevention and control of major pandemics [J]. Science China (Information Sciences), 2020, 50(7): 1069 – 1090. (in Chinese)

[2] 汪开普, 马晓艺, 卢超, 等. 基于强化学习与遗传算法的机器人并行拆解序列规划方法[J]. 国防科技大学学报, 2025, 47(2): 24 – 34.
WANG K P, MA X Y, LU C, et al. Robotic parallel disassembly sequence planning method based on reinforcement learning and genetic algorithm [J]. Journal of National University of Defense Technology, 2025, 47(2): 24 – 34. (in Chinese)

[3] BILLARD A, KRAGIC D. Trends and challenges in robot manipulation[J]. Science, 2019, 364(6446): eaat8414.

[4] TEE K P, CHEONG S, LI J, et al. A framework for tool cognition in robots without prior tool learning or observation[J]. Nature Machine Intelligence, 2022, 4(6): 533 – 543.

[5] ZHANG J Z, GIREESH N, WANG J L, et al. GAMMA: graspability-aware mobile manipulation policy learning based on online grasping pose fusion[C]//Proceedings of the 2024 IEEE International Conference on Robotics and Automation (ICRA), 2024: 1399 – 1405.

[6] FUJITA Y, UENISHI K, UMMADISINGU A, et al. Distributed reinforcement learning of targeted grasping with active vision for mobile manipulators[C]//Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2020: 9712 – 9719.

[7] XIA F, LI C S, MARTIN-MARTIN R, et al. ReLMoGen: integrating motion generation in reinforcement learning for mobile manipulation [C]//Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), 2021: 4583 – 4590.

[8] JIA Z W, THUMULURI V, LIU F C, et al. Chain-of-thought predictive control[C]//Proceedings of the 41st International Conference on Machine Learning, 2024: 21768 – 21790.

[9] HUANG X Y, BATRA D, RAI A, et al. Skill transformer: a monolithic policy for mobile manipulation[C]//Proceedings of the 2023 IEEE/CVF International Conference on Computer Vision (ICCV), 2023: 10818 – 10828.

[10] THOR M, MANOONPONG P. Versatile modular neural locomotion control with fast learning[J]. Nature Machine Intelligence, 2022, 4(2): 169 – 179.

[11] LI C H, VLASTELICA M, BLAES S, et al. Learning agile skills via adversarial imitation of rough partial demonstrations[C]//Proceedings of the 6th Conference on Robot Learning, 2023: 342 – 352.

[12] TRIANTAFYLLIDIS E, CHRISTIANOS F, LI Z B. Intrinsic language-guided exploration for complex long-horizon robotic manipulation tasks [C]//Proceedings of the IEEE International Conference on Robotics and Automation, 2024: 7493 – 7500.

[13] ZHANG D D, LI Q, ZHENG Y, et al. Explainable hierarchical imitation learning for robotic drink pouring[J]. IEEE Transactions on Automation Science and Engineering, 19(4): 3871 – 3887.

[14] LI C S, XIA F, MARTÍN-MARTÍN R, et al. HRL4IN: hierarchical reinforcement learning for interactive navigation with mobile manipulators[C]//Proceedings of the Conference on Robot Learning, 2020: 603 – 616.

[15] JIANG R, CHENG X, SANG H R, et al. GTHSL: a goal-task-driven hierarchical sharing learning method to learn long-horizon tasks autonomously [J]. IEEE Transactions on Industrial Electronics, 2024, 72(4): 3994 – 4005.

[16] TRIANTAFYLLIDIS E, ACERO F, LIU Z C, et al. Hybrid hierarchical learning for solving complex sequential tasks using the robotic manipulation network ROMAN [J]. Nature Machine Intelligence, 2023, 5(9): 991 – 1005.

[17] GU J Y, CHAPLOT D S, SU H, et al. Multi-skill mobile manipulation for object rearrangement [C]//Proceedings of the 11th International Conference on Learning Representations (ICLR), 2022.

[18] HO J, ERMON S. Generative adversarial imitation learning[C]// Proceedings of the 30th International Conference on Neural Information Processing Systems, 2016: 4572 – 4580.

[19] SCHULMAN J, WOLSKI F, DHARIWAL P, et al. Proximal policy optimization algorithms[EB/OL]. (2017 – 08 – 28) [2024 – 11 – 01]. https://arxiv.org/abs/1707.06347.

[20] SCHULMAN J, MORITZ P, LEVINE S, et al. High-dimensional continuous control using generalized advantage estimation [C]//Proceedings of the High-Dimensional Continuous Control Using Generalized Advantage Estimation, 2016.

[21] MARZARI L, PORE A, DALL′ALBA D, et al. Towards hierarchical task decomposition using deep reinforcement learning for pick and place subtasks[C]//Proceedings of the 20th International Conference on Advanced Robotics (ICAR), 2021.