arXiv:2301.07608v1 [cs.LG] 18 Jan 2023

DeepMind

开放式任务空间中的人类时间尺度适应

自适应智能体团队[1]
DeepMind

基础模型在监督学习和自监督学习问题中展现出令人瞩目的适应能力和可扩展性，但迄今为止，这些成功尚未充分延伸至强化学习（RL）。在这项工作中，我们证明，大规模训练强化学习智能体能够产生一种通用的上下文学习算法，使其能够像人类一样迅速适应开放式的新颖具身三维问题。在广阔的留出环境动力学空间中，我们的自适应智能体（AdA）展现出即时的假设驱动探索、对已获得知识的高效利用，并且能够通过第一人称示范成功提示。适应能力源于三个要素：（1）在广阔、平滑且多样的任务分布上进行元强化学习；（2）将策略参数化为大规模的基于注意力的记忆架构；（3）采用有效的自动化课程，优先安排处于智能体能力前沿的任务。我们展示了网络规模、记忆长度以及训练任务分布丰富度方面具有特征性的缩放定律。我们认为，我们的成果为打造日益通用且适应性更强、能够在规模不断扩大的开放领域中表现出色的强化学习智能体奠定了基础。

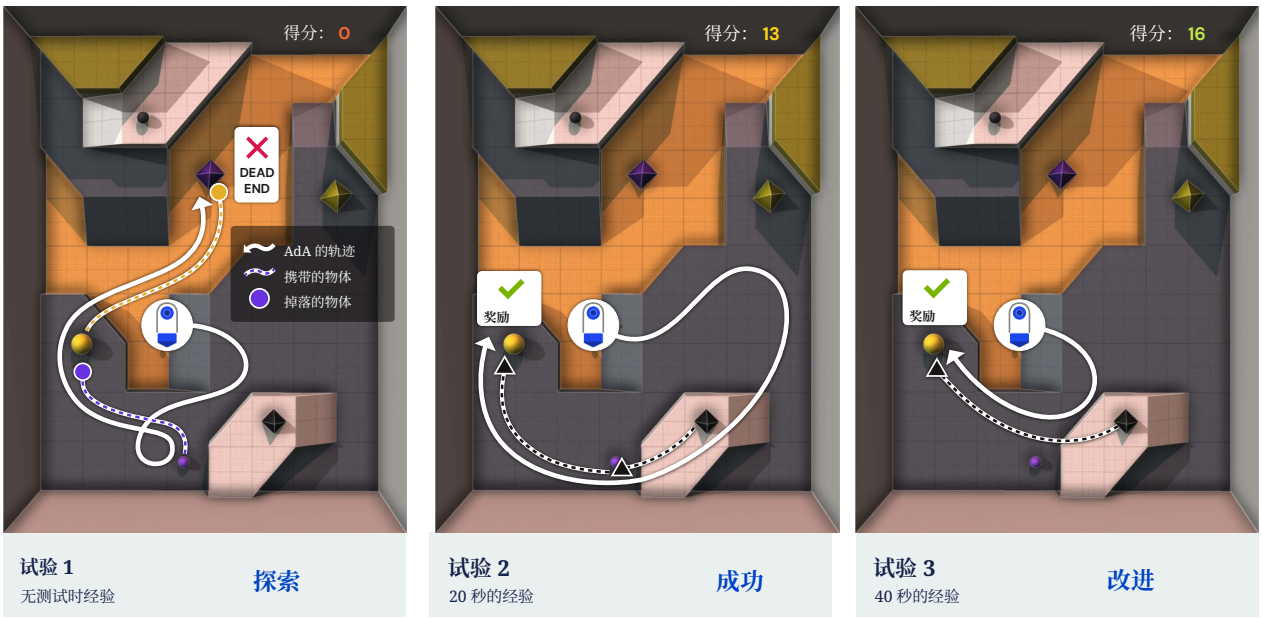


图 1 | **人类时间尺度适应。** 我们的智能体（AdA）在测试时经验仅数分钟内，无需进一步训练，便在复杂三维环境中解决留出任务的轨迹示例。初始试次（**探索**）展现了一种能够揭示隐藏环境动态的策略。仅经过数秒的测试时经验（**成功**）后，AdA 找到了任务的有效解决方案。随后（**改进**），它进一步完善这一解决方案，逐步找到奖励更高的行为。实线白线表示智能体的移动。虚线彩线表示智能体携带相应颜色物体时的移动。有关任务的完整描述，请参见图 B.1。AdA 行为的视频可在我们的微型网站及配套的 成果视频集中查看。

通讯作者：*Feryal Be* *hbahani*（*feryal@deepmind.com*）和 *Edward Hughes*（*edwa* *rdhughes@deepmind.com*）。

© 2023 DeepMind。保留所有权利

1. 引言

在几分钟内实现适应的能力，是人类智能的显著特征，也是迈向通用智能的重要里程碑。无论有限理性达到何种程度，总会存在这样一类任务：智能体不可能仅靠零样本泛化策略取得成功，但如果能够通过反馈进行极快速的上下文学习，就有可能取得进展。为了在现实世界中发挥作用并与人类交互，我们的人工智能体应当能够仅凭少量交互实现快速而灵活的适应，并随着更多数据的出现持续适应。为了将适应这一概念转化为可操作的目标，我们希望训练一种智能体，使其在测试时仅凭未见环境中的少数几个回合，就能完成需要试错探索的任务，随后还能不断改进解决方案，使其趋近于最优行为。

元强化学习已被证明能够有效实现快速的上下文内适应（例如，Yu 等人 (2020年）；Zintgraf (2022年））。然而，在奖励稀疏、任务空间广阔且多样的情境中，元强化学习的成功有限（Yang 等人，2019年）。在强化学习之外，基础模型在半监督学习领域引起了广泛关注（Bommasani 等人，2021年），原因在于它们能够通过不同任务的示范，在少样本条件下实现适应。这些模型旨在提供强大的通用知识与技能基础，并可通过微调或利用示范进行提示，进一步构建并适应新情境（Brown 等人，2020年）。基于注意力的记忆架构（如 Transformer）（Vaswani 等人，2017年）对这一成功至关重要；随着参数数量增加，这类架构的性能呈幂律缩放（Tay 等人，2022年）。

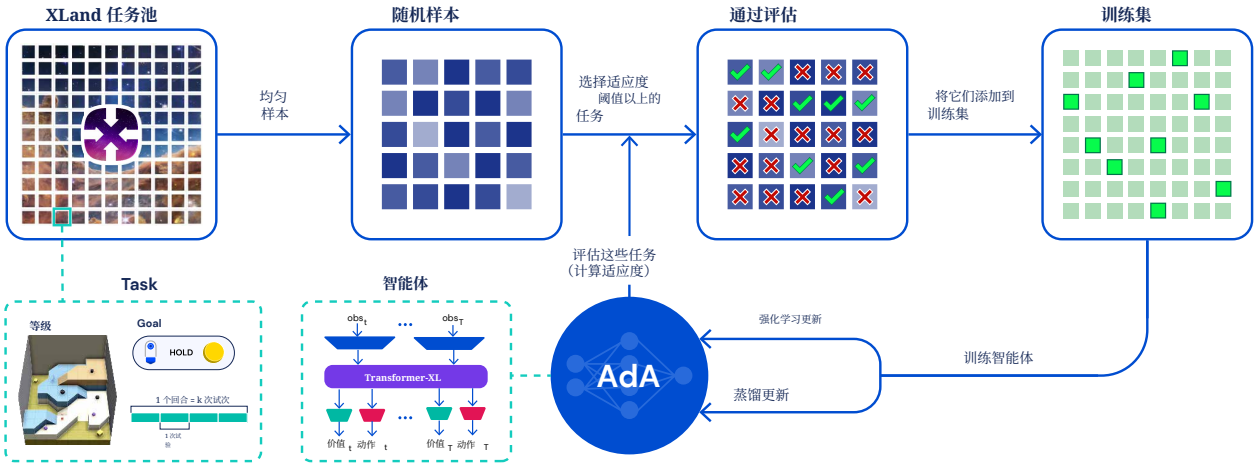


图 2 | **训练我们的自适应智能体（AdA）**。我们在 XLand 中使用元强化学习训练一个大型 Transformer 模型。训练期间，任务会被均匀采样，随后经过筛选，形成一个不断变化的任务训练池，这些任务处于智能体能力的前沿。在这些任务上训练后，智能体能够像人类一样有效且高效地适应未见过的手工编写任务。

在本研究中，我们为训练强化学习基础模型铺平了道路；也就是说，这是一种在庞大任务分布上预训练的智能体，并且能够在测试时通过少样本适应于广泛的下游任务。我们引入了 自适应智能体 （AdA），这是一种能够在人类时间尺度上，在具有稀疏奖励的庞大开放式任务空间中进行适应的智能体。AdA 无需任何提示（Reed 等，2022年）、微调（Lee 等，2022年）或访问离线数据集（Laskin 等，2022年；Reed 等，2022年）。相反，AdA 展现出假设驱动的探索行为，利用即时获得的信息来改进其策略，并取得接近最优的性能。AdA 能够高效获取知识，在部分可观测的三维环境中，面对具有第一人称像素观测的复杂留出稀疏奖励任务时，可在数分钟内完成适应。一项人类研究证实，其时间尺度

适应能力与经过训练的人类玩家相当。AdA 在一个具有代表性的留出任务中的适应行为如图 1 所示。通过使用第一人称示范进行零样本提示，AdA 还能够实现更优的性能，这与语言领域中的基础模型类似。

我们选择 Transformer 作为架构，以通过基于模型的强化学习[2] 扩展上下文内快速适应（Duan 等，2017；Melo，2022；Wang 等，2016）。基础模型通常需要大量多样化的数据集，才能实现其通用性（Brown 等，2020；Mahajan 等，2018；Schuhmann 等，2022；Sun 等，2017；Zhai 等，2022）。为了在强化学习环境中实现这一点——在这种环境中，智能体收集自己的数据——我们扩展了近期推出的 XLand 环境（OEL Team 等，2021），构建出一个庞大的开放式世界，其中包含超过 10^{40} 种可能的任务。这些任务需要多种不同的在线适应能力，包括实验、导航、协同、劳动分工，以及应对不可逆性。鉴于可能任务的范围十分广泛，我们采用自适应自动课程，优先安排处于智能体能力前沿的任务（Jiang 等，2021a；OEL Team 等，2021）。最后，我们采用蒸馏（Schmitt 等，2018），使模型规模得以扩展至超过 500M 个参数；据我们所知，这是发表时从头开始用强化学习训练的最大模型（Ota 等，2021）。我们的方法概述见图2。

我们的主要贡献如下：

- 我们提出了 AdA，这是一种能够在广泛且具有挑战性的任务中实现人类时间尺度适应的智能体。
- 我们在开放式任务空间中，使用自动化课程通过大规模元强化学习训练 AdA。
- 我们表明，适应受到记忆架构、课程以及训练任务分布的规模和复杂性的影响。
- 我们得出了模型规模和记忆方面的缩放定律，并证明 AdA 通过零样本第一人称提示提升了性能。

2. 自适应智能体（AdA）

为了在广阔多样的任务空间中实现人类时间尺度的适应，我们提出了一种通用且可扩展的基于记忆的元强化学习方法，训练出一个自适应智能体 (AdA)。我们在 XLand 2.0 中训练并测试 AdA。该环境支持程序化生成多样的三维世界和多人游戏，其丰富的动力学特性要求智能体能够适应。我们的训练方法结合了三个关键组成部分：用于引导智能体学习的课程、用于训练具有大规模基于注意力的记忆的智能体的基于模型的强化学习算法，以及支持扩展的蒸馏。图2展示了我们方法的概览。在接下来的章节中，我们将介绍各个组成部分及其如何促进高效的少样本适应。

2.1. 开放式任务空间：XLand 2.0

为展示在开放式任务空间中的快速适应能力，我们扩展了程序生成的三维环境 XLand（OEL Team 等人，2021年），并在此将其称为 XLand 1.0。在 XLand 中，一个任务由一场游戏、一个世界和一系列协同玩家策略（如果有）组成。游戏为每位玩家设定一个目标，该目标定义为关于环境状态的布尔函数（谓词）。当且仅当目标达成时，智能体才会获得奖励。目标以合成语言定义，智能体会接收相应的编码。世界指定了静态地面拓扑、玩家可以交互的物体，以及玩家的生成位置。智能体通过第一人称像素观测来观察世界及其中的任何协同玩家。游戏、世界和协同玩家系统的所有基本细节均

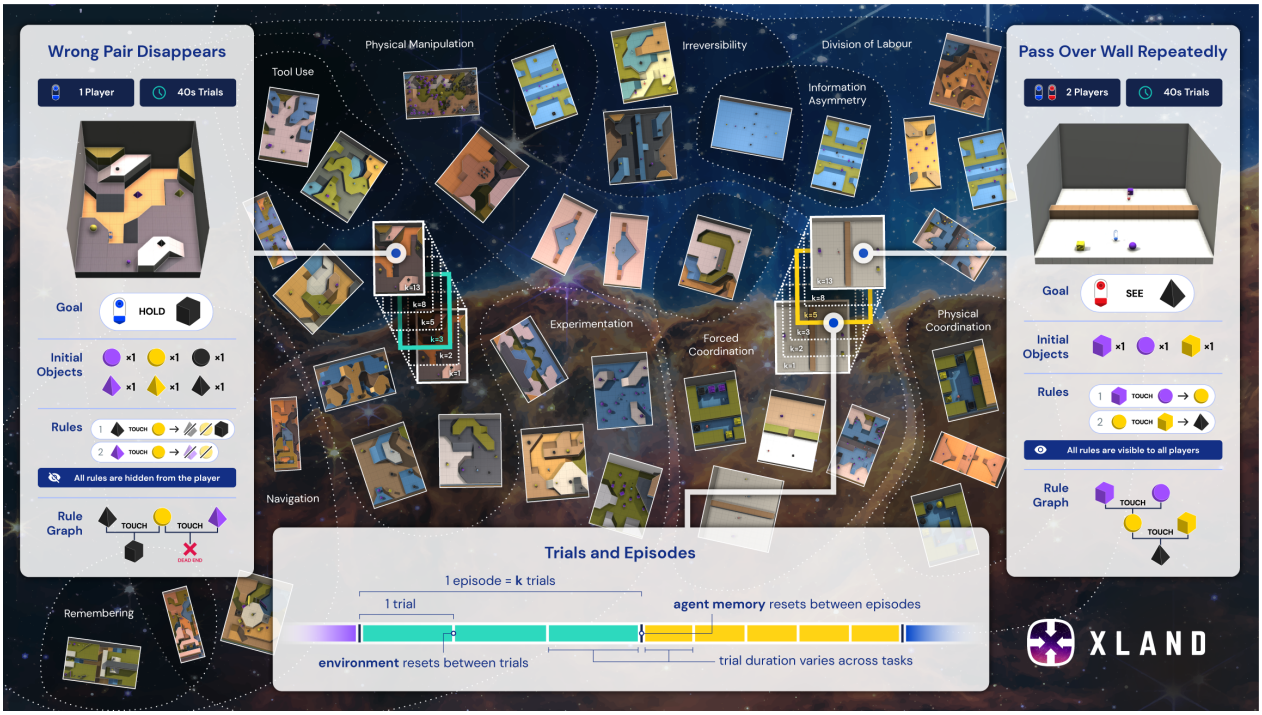


图 3 | **XLand 2.0：一个广阔、连续且多样的适应问题任务空间**。不同任务的适应要求各不相同，例如实验、工具使用或劳动分工。例如，在需要实验的任务中，玩家可能需要识别哪些物体可以有效组合，避开死胡同，然后优化物体的组合方式，就像玩具版的实验化学。每项任务可以进行一次或多次试次；试次之间环境会重置，但智能体记忆不会重置。图中突出显示了两个示例任务，`Wrong Pair Disappears` 和 `Pass Over Wall Repeatedly`，展示了目标、初始物体、生成规则（图中称为“规则”），以及智能体需要如何与它们交互才能完成任务。完整的任务描述请参阅附录F.1。

继承自原始 XLand；详见OEL Team 等人（2021年），以获取完整描述；新增功能的详情请参阅附录A.1。

XLand 2.0 在 XLand 1.0 的基础上增加了一个名为生成规则的系统。每条生成规则都表示一种额外的环境动力学，从而产生比 XLand 1.0 丰富得多、也多样得多的转移函数。生成规则系统可以被视为一种用于表达这些多样动力学的领域专用语言（DSL）。每条生成规则由以下部分组成：

1. 一个 `condition`，即谓词，例如 `near(yellow sphere,black cube)`；2. 一个可能为空的 `spawn`列表，它们是物体，例如 `purple cube`， `black cube`。

当满足 `condition` 时，存在于 `condition` 中的物体会从环境中移除，而 `spawns` 中的物体会出现。每个游戏可以包含多个产生式规则。根据任务配置的不同，产生式规则可能对玩家可见，也可能部分或完全隐藏。更确切地说，有三种不同的机制可用于向玩家隐藏产生式规则信息：

1. 隐藏一条完整的生成规则，玩家只能获知规则存在，但不知道其条件，也不知道会生成什么。

2. 隐藏一个物体，即从所有生成规则中隐藏某个特定物体。隐藏的物体会编号，以便智能体在多个物体被隐藏时能够区分它们。 3. 隐藏条件的谓词，智能体可以知道需要满足某个 谓词的物体，但不知道具体是哪个谓词。隐藏的谓词也会编号。

我们不在运行时按程序生成任务，而是预先采样一个大型任务池。有关我们用于预采样任务的具体机制的更多详情，请参见附录 A.2。我们在图3中将 XLand 2.0 任务空间可视化。

2.2. 元强化学习

我们采用黑盒元强化学习问题设定（Duan 等人，2017年；Wang 等人，2016年）。我们将任务空间 M 定义为一组部分可观测马尔可夫决策过程（POMDPs）。对于给定任务 $m \in \mathrm{M}$，我们将试验 定义为从初始状态 s_0 到终止状态 s_T的任意转移序列。[1] 在 XLand 中，当某个按任务指定的时间段 $T \in [10\mathrm{s}、40\mathrm{s}]$ 经过后，任务才会终止。环境以每秒 30 帧的频率运行，智能体每 4^{th}帧进行一次观测，因此以时间步为单位的任务长度范围为 $[75, 300]$。

一个回合由给定任务的一系列 k 试次组成 m。在试次边界处，任务会重置到一个初始状态。在我们的领域中，除智能体的朝向外，初始状态都是确定性的；智能体的朝向则是从均匀随机分布中采样的。试次和回合的结构如图3所示。

在黑盒元强化学习训练中，智能体利用与广泛任务分布交互的经验来更新其神经网络的参数；该网络根据状态观测为智能体的动作参数化策略分布。如果智能体具有动态内部状态（记忆），那么元强化学习训练便会利用重复试次的结构，为这种记忆赋予一种隐式的在线学习算法（Mikulik 等人，2020年）。

在测试时，这种在线学习算法使智能体无需进一步更新神经网络权重即可调整其策略。因此，智能体的记忆不会在试次边界处重置，而是在回合边界处重置。要生成一个回合，我们采样一对（m，k），其中 $k \in \{1，2，\ldots\ldots 6\}$。正如我们稍后将讨论的，在测试时，AdA 会在多种 k 取值下对未见过的留出任务进行评估，其中也包括训练期间未见过的留出 k 。关于 AdA 元强化学习方法的完整细节，请参阅附录D.1。

2.3. 自动课程学习

鉴于我们预先采样的任务池规模庞大且种类繁多，智能体很难通过均匀采样有效学习。大多数随机采样的任务可能过于困难（或过于简单），无法促进智能体的学习进度。为此，我们采用自动化方法，选择处于智能体能力前沿的“有趣”任务，这与人类认知发展中的“最近发展区”类似（Vygotsky，1978年）。我们提出了对两种现有方法的扩展，二者均显著提升了智能体的性能和样本效率（见第 3.3 节），并形成涌现课程，随着时间推移选择复杂度不断增加的任务。

空操作过滤。 我们将 OEL Team 等人 (2021年，第5.2节）提出的动态任务生成方法扩展到了我们的设置中。从池中采样出新任务后，首先会对其进行评估，以判断

[1]请注意，我们采用了与 Duan 等人（2017年）相反的命名约定。按照我们的约定，“试验”一词与人类行为学文献中的相关概念十分契合（Barbosa 等人，2022年）。

AdA 是否能够从中学习。我们对 AdA 的策略和“空操作”控制策略（在环境中不采取任何动作）进行若干回合的评估。仅当这两种策略的得分满足若干条件时，该任务才会用于训练。我们扩展了原始空操作过滤中的条件列表，并采用归一化阈值，以适应不同的试验时长。详情请参阅附录D.5 。

优先级关卡回放（PLR）。我们对“鲁棒 PLR”（此处称为*PLR*，Jiang 等人(2021a）进行修改，以适配我们的设置。与空操作过滤不同，PLR 使用适应度分数(Schmidhuber，1991 年) 来近似智能体在给定任务上的遗憾值。我们考察了若干种估计智能体遗憾值的方法，从Jiang 等人(2021b使用的时序差分误差，到采用 AdA 动力学模型误差的新方法（见附录D.5 和图D.1）。

PLR 通过维护一个固定大小的存档来运行，其中保存适应度最高的任务。我们只在从存档中抽取的任务上训练 AdA，抽取概率为 p。以$1-p$的概率，随机抽取并评估一个新任务，并将其适应度与存档中的最低值进行比较。如果新任务的适应度更高，就将其加入存档，并移除适应度最低的任务。因此，PLR 也可视为一种过滤形式，其采用动态标准（存档中的最低适应度值）。与空操作过滤不同，只要任务保持较高适应度，就可以反复从存档中抽取。为了在异构任务空间中应用 PLR，我们使用滚动均值和方差对每个试验索引的适应度进行归一化，并采用每时间步适应度的均值而非总和，以考虑试验时长的差异。最后，由于我们关注跨试验适应后处于智能体能力前沿的任务，因此只使用最后一次试验的适应度。详情见附录D.5。

2.4. 强化学习智能体

学习算法。 我们使用 Muesli（Hessel等人，2021年）作为强化学习算法。这里简要介绍该算法，详情请参阅原始论文。AdA以依赖历史的编码作为输入（在我们的案例中，是循环神经网络或 Transformer 的输出），并学习一个序列模型（长短期记忆网络），以预测接下来 I 步的价值 $\hat{v}_i$、动作分布 $\hat{\pi}_i$ 和奖励 $\hat{r}_i$。这里，$i=0,\ldots,I$ 表示提前 i 步的预测。I 通常较小，在我们的案例中为 $I=4$。对于每个观测到的步骤 t，模型展开 I 步，并根据相应的目标进行更新：

$$\mathcal{L}_r^t=\sum_{i=0}^{I}\left(\hat{r}_i^t-r_{t+i}\right)^2,\ \mathcal{L}_v^t=\sum_{i=0}^{I}\left(\hat{v}_i^t-G_{t+i}\right)^2,\ \mathcal{L}_\pi^t=\sum_{i=0}^{I}\mathrm{KL}\left(\pi_{\mathrm{CMPO}}^{t+i}\,\middle\|\,\hat{\pi}_i^t\right). \tag{1}$$

这里，r_{t+i} 指观测奖励。G_{t+i} 指使用 Retrace（Munos等人，2016年）计算得到的价值目标，其依据是模型单步预测所得的 Q 值。

动作目标 π_{CMPO}^t 通过对当前策略[2] 重新加权得到，所依据的是经过裁剪、归一化并进行指数变换的优势函数。此外，Muesli 还加入了一项基于这些优势函数的辅助策略梯度损失，以帮助优化对动作概率的即时预测。最后，Muesli 维护一个滞后于序列模型的目标网络，用于执行动作以及计算 Retrace 目标和优势函数。

记忆架构。 记忆是适应的关键组成部分，因为它能让智能体存储并回忆过去学到和经历过的信息。为了让智能体能够有效地适应

[2]先验分布实际上是当前策略估计值、用于生成样本的（过时）策略，以及均匀分布的混合；后两者混合后作为正则化项。

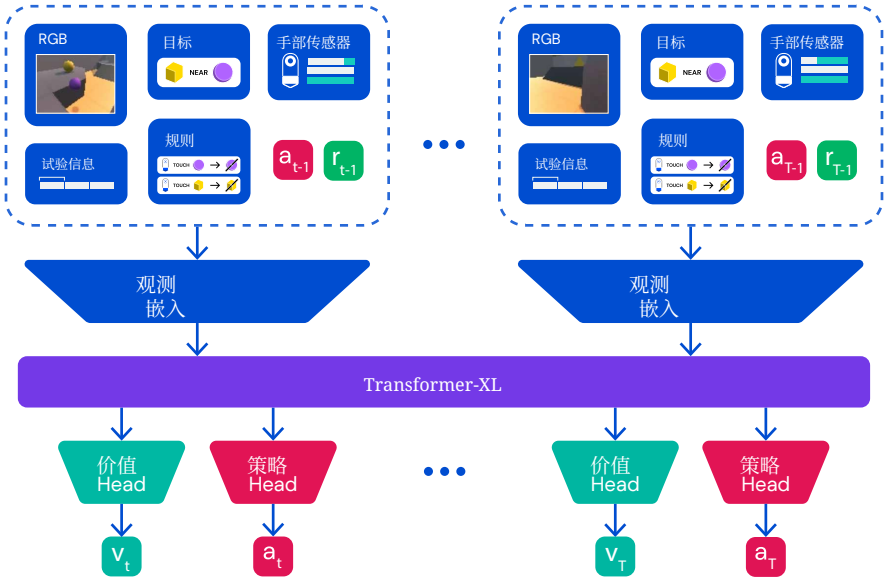


图 4 | **智能体架构**。在每个时间步，我们将像素观测、目标、手部、试验和时间信息、生成规则、先前动作以及先前奖励嵌入并合并为一个向量。这些观测嵌入依次输入 Transformer-XL，其输出嵌入再输入多层感知机价值头、多层感知机策略头，以及 Muesli LSTM 模型步骤（为简洁起见，图中省略）。有关智能体架构的更多详情，请参见附录C.1。

任务要求的变化，记忆应能让智能体回忆起近期和更久远的信息。虽然缓慢的基于梯度的更新能够捕捉后者，却往往来不及捕捉前者，即快速适应。大多数基于记忆的元强化学习研究都采用循环神经网络作为快速适应机制（Parisotto，2021年）。在这项工作中，我们表明，循环神经网络无法在我们这个具有挑战性的部分可观测具身三维任务空间中实现适应。我们尝试了两种记忆架构来解决这一问题：

1. 带注意力机制的循环神经网络会将若干过去的激活值（本例中为 64 个）存入情景记忆，并以当前隐藏状态作为查询，对其进行注意力计算。注意力模块的输出随后与隐藏状态拼接，并输入循环神经网络。我们通过仅将每 8th 个激活值存入情景记忆，来增加智能体的有效记忆长度。[3]

2. *Transformer-XL*（*TXL*）(Dai 等人，2019年）是 Transformer 架构（Vaswani 等人，2017年）的一种变体，支持使用更长且长度可变的上下文窗口，以增强模型捕捉长期依赖的能力。为提高使用强化学习训练 Transformer 的稳定性，我们按照Parisotto 等人（2020年）的做法，在每个层之前进行归一化，并采用Shazeer (2020年）中的方法对前馈层使用门控。

两个记忆模块都以一系列学习得到的时间步嵌入为输入，并生成一系列输出嵌入，输入Muesli架构，如图所示：4中的 Transformer-XL模块。在第 3.2 节中，我们表明，在需要适应的任务中，这两种基于注意力的记忆模块的表现都显著优于原始循环神经网络。Transformer-XL 的表现最佳，因此，除非另有说明，我们在所有实验中都将其用作默认记忆架构。

[3]这些数值是在性能与速度之间折中后得出的。请注意，最终得到的架构比同等规模的 Transformer 更慢。

超越少样本。我们提出了一种对 Transformer-XL架构的简单修改，可在不增加计算成本的情况下延长有效记忆长度。由于视觉强化学习环境中的观测往往具有很强的时间相关性，我们提出对序列进行子采样，方法如带注意力机制的循环神经网络中所述，使智能体能够关注多达 4 倍数量的试次。为确保仍能关注位于子采样点之间的观测，我们首先使用循环神经网络对整个轨迹进行编码，旨在每个步骤总结近期历史。我们表明，额外的循环神经网络编码不会影响我们所用 Transformer-XL 变体的性能，却能实现更长范围的记忆（见第 3.7节）。

2.5. 蒸馏

在训练的前四十亿步中，我们使用额外的蒸馏损失（Czarnecki 等人，2019年；Schmidhuber，1992；Schmitt 等人，2018年），通过预训练教师模型的策略来引导 AdA 学习，这一过程称为启动训练；反复进行这一过程会形成代际训练机制（OEL Team 等人，2021年；Wang 等人，2021年）。教师模型通过强化学习从头进行预训练，采用与 AdA 相同的训练流程和超参数，区别在于不进行初始蒸馏，且模型规模较小（教师模型有 2300 万个 Transformer 参数，多智能体 AdA 有 2.65 亿个）。与上述先前研究不同，我们在较早的代际中不采用奖励塑形或基于种群的训练（PBT，Jaderberg 等人（2017年）。蒸馏期间，AdA 根据自身策略行动，而教师模型则根据 AdA 观察到的轨迹提供目标逻辑值。蒸馏使我们能够摊销原本代价高昂的初始训练阶段，并帮助智能体克服训练初期阶段习得的有害表征；见第3.6节。

为了将蒸馏损失与 Muesli 结合，我们从学生观测到的每个转移开始展开模型。我们最小化模型预测的所有动作概率与教师策略在相应时间步预测的动作概率之间的 KL 散度。类似于 Muesli 的策略损失 $\mathcal{L}_\pi$ （定义于（1）），我们定义

$$\mathcal{L}_{\text{dist}} = \sum_{i=0}^{I} \text{KL}\left(\tilde{\pi}_0^{t+i} \,\middle\|\, \hat{\pi}_i^t \right), \tag{2}$$

其中，$\tilde{\pi}$对应于教师根据相同的观测历史预测得到的动作逻辑值。此外，我们发现，在蒸馏过程中加入额外的 L^2 正则化很有用。

3. 实验与结果

我们在两种截然不同的情境下评估智能体：一组从与训练任务相同的分布中采样的 1000 个测试任务，以及一组由 30 个单智能体和 28 个人工编写的多智能体探测任务。拒绝采样过程确保程序生成的测试任务和探测任务不在训练集中。探测任务呈现的情境对人类而言尤其直观，并且有意涵盖了各种在性质上迥异的适应行为。附录中的图B.1 至B.3展示了探测任务示例，而附录 F中提供了每个探测任务的完整描述。

每项任务可获得的总奖励各不相同，因此，每当我们展示测试集或手工编写的任务集上的汇总结果时，都会将每项任务的每次试验总奖励，与在相应任务集上对 AdA 进行微调所获得的奖励进行归一化。我们将归一化后的奖励称为得分。我们规定，自适应智能体必须具备两种能力：零样本泛化和少样本适应。零样本泛化 通过以下情况下的得分来评估：智能体仅与一个留出任务进行 1 次试验的交互。少样本适应 则通过得分的提升来评估，

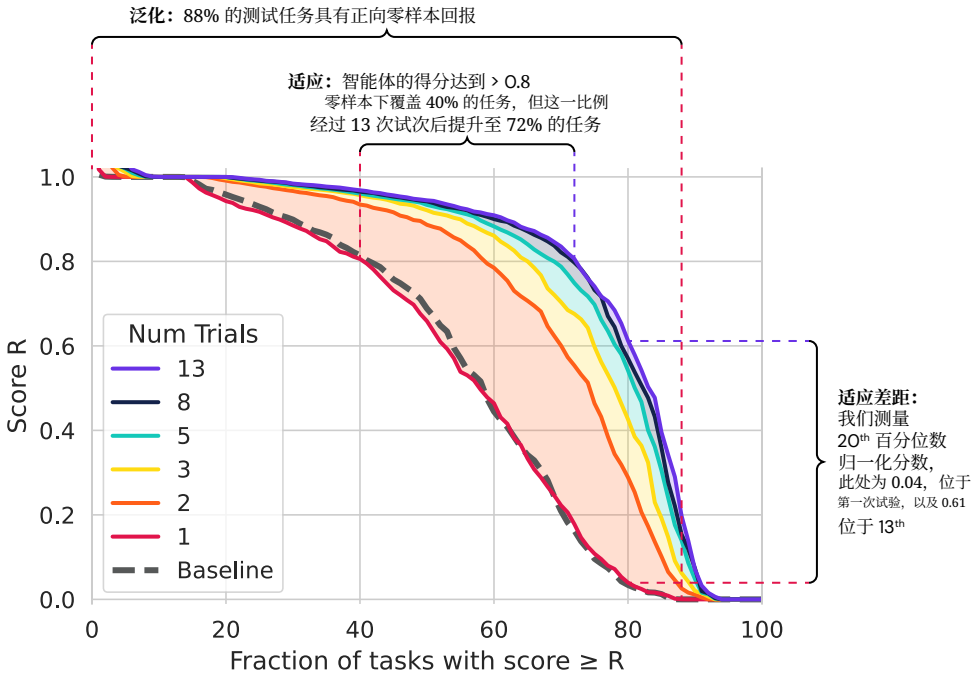


图 5 | **零样本泛化与少样本适应**。我们报告了在不同试验次数下评估时，单智能体测试集上归一化任务分数的分布。 y轴表示相对于在测试任务上微调的智能体（近似“无限次试验”性能）的最后一次试验总奖励。曲线越向右上角移动，表示性能越好。给予更多试验后，智能体在最后一次试验中获得了更高分数，表明其在任务分布的大部分范围内实现了测试时适应（阴影区域）。虚线表示在每个回合仅包含单次试验的训练机制下训练的智能体的零样本性能。

也就是随着智能体与任务进行的交互试次数逐步增加（k），得分如何变化。更准确地说，对于每个 k，我们报告最后一次试验的得分，以此展示智能体能否即时利用额外经验来提升表现，即衡量其适应能力。

我们使用一个或多个百分位数汇总任务集上的得分。展示单个探测任务时，我们报告每项任务中智能体以及适用情况下人类玩家的未归一化最后一次试验总奖励。关于评估方法的完整详情，请参阅附录 B。

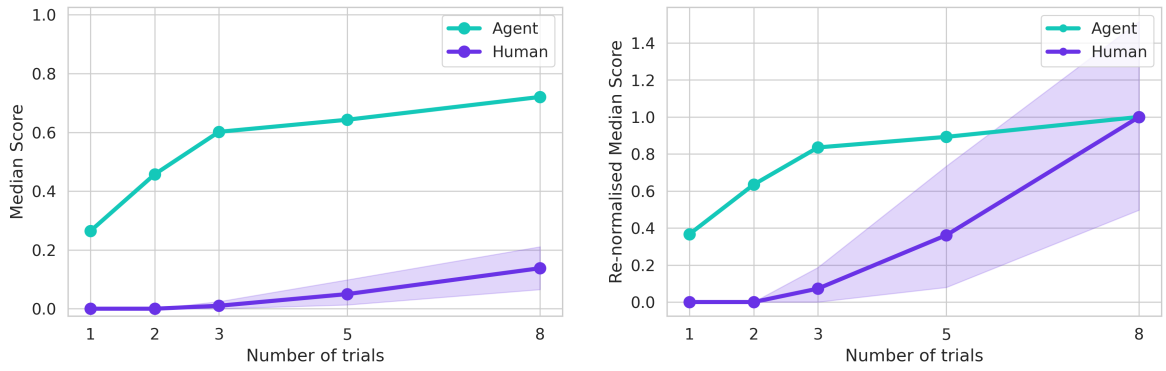
AdA 的训练配置空间很大，涵盖模型规模、自动课程、记忆架构、记忆长度、XLand 任务池中的任务数量、单智能体与多智能体任务、蒸馏教师以及训练步数。每次实验比较中，我们采用一致的训练配置，但不同的实验比较之间配置有所不同。因此，提醒读者不要直接比较不同章节中的结果。为方便起见，所有实验配置均列于附录 D。

3.1. AdA 展现了人类时间尺度适应

单智能体。在图 5中，我们展示了 AdA 在表 1所述的单智能体设置下训练后的性能。首先来看 AdA 的零样本性能（$k = 1$，红线）。其表现与基线智能体相当；该基线智能体仅在每个回合都由单次试验组成的设置下训练。换言之，尽管 AdA 是在试验次数分布为

表 1 | 第3.1节中智能体实验的实验设置。

玩家数量	模型参数	记忆	任务池	课程	教师	步数
1	169M TXL / 总计 353M	1800	25B	PLR D.5	D.1	100B
2	265M TXL / 总计 533M	1800	参见附录 D.3	PLR D.5	D.2	70B



(a) (b)

图 6 | **人类时间尺度适应**。 我们报告了 AdA 和人类玩家在 30 项人工编写任务中的中位数归一化末次试验得分，并将其作为试次数量的函数。随着试次数量增加，AdA 和人类玩家的性能均有所提升，这表明 AdA 能够实现人类时间尺度适应。**(a)** 展示了采用标准逐任务归一化方案得到的结果。**(b)** 则按照每种玩家类型的最高分对结果重新归一化，以考虑智能体与人类玩家之间的系统性差异。具体而言，人类玩家表示游玩时存在延迟，这可能导致得分较低。

$k \in \{1、2、\ldots\ldots、6\}$ 上训练的，AdA 的零样本性能并未因此下降。现在请看 AdA 的少样本性能（$k \in \{2、3、5、8、13\}$，从橙色到紫色的线）。随着试次数增加，AdA 在超过 80% 的任务集上提升了性能，清楚地展现了其在测试时的适应能力。与零样本性能相比，两次试验设置下的提升尤为明显；而随着试次数进一步增加，AdA 的性能仍持续提升。

我们比较了 AdA 与一组人类玩家在 30 项留出的人工编写探测任务上的性能，以评估 AdA 是否能以与人类相同的时间尺度进行适应。图6a 展示了 AdA 和人类玩家的中位数得分随试次数量变化的情况。随着体验到更多任务试次，AdA 和人类玩家的得分都能够提高，这表明在这组探测任务中，AdA 展现出了人类时间尺度适应。图 F.1提供了各项任务得分的更多细节。图中显示，有少数任务人类可以解决而 AdA 无法解决，例如 `Spacer Tool` 任务：在这项任务中，必须将一个物体用作工具来移动另一个物体，而这种情况在 XLand 中极为罕见。还有一些任务，例如 `Small Workstation`，`All Rules Visible` AdA 可以解决而人类无法解决，这可能是因为控制要求复杂。不过，大多数任务都显示人类和 AdA 均能适应；AdA 得分曲线的斜率与人类玩家一样陡峭，甚至更陡，尤其是在试次数量较少时。有关我们的人类实验设计的完整详情，请参阅附录 B.4。

图 7 更详细地分析了 AdA 在某个特定留出任务上的行为。试次数增加时得分随之提高，这表明试次数越多，任务就能越稳定、更快地解决。考察不同试次数下的轨迹，

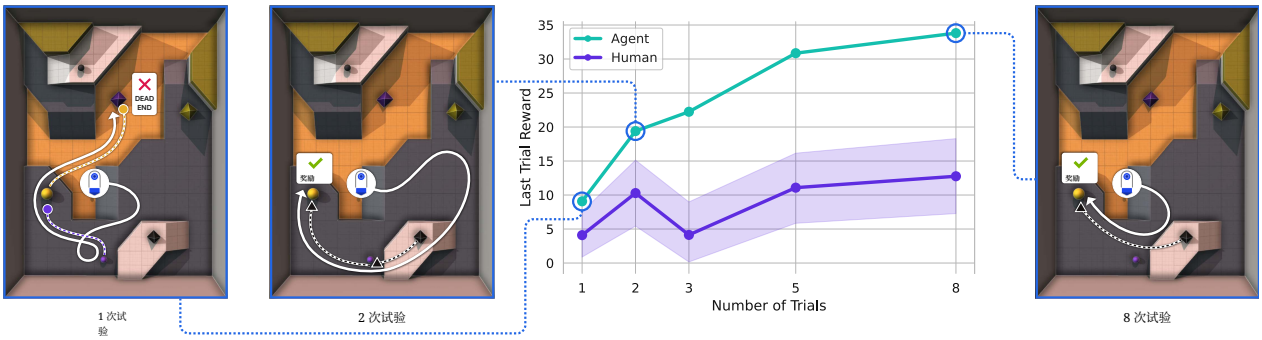


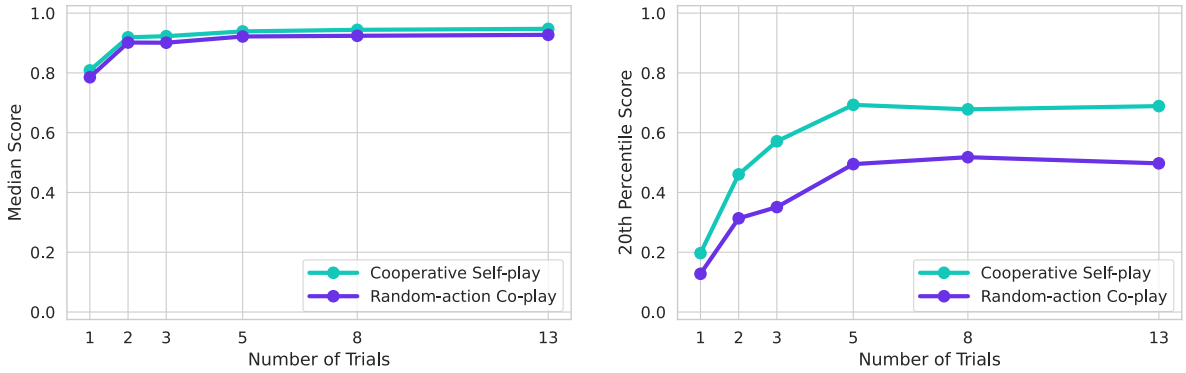
图 7 | **实验、成功与改进**。我们报告 AdA 在探测任务上的平均性能和代表性行为 `Wrong Pair Disappears`，评估时采用不同数量的试次。提供更多试次时，AdA 的性能会提升，展现出测试时适应能力。俯视图图像展示了总试次数不同时，最后一次试验中具有代表性的轨迹。对应的视频展示了 $k = 3$ 一个回合中所有试次的行为。

我们可以从 AdA 的行为来解释这一现象。当提供 1 或 2 次试验时，AdA 会进行结构化的假设驱动探索：尝试物体的不同组合，并找到解决方案或陷入死胡同。找到解决方案后，AdA 会在后续试次中改进策略，更高效地收集正确的物体，并以正确的方式将它们组合起来。因此，提供更多试次进行改进时，AdA 能够获得更高的最后一次试验得分。提供 8 次试验时，其最后一次试验的性能接近微调智能体。我们在许多留出探测任务中都持续观察到这种行为模式；视频请见我们的 微型网站。

多智能体。我们在完全合作的多智能体任务和单智能体任务的混合任务集上训练一个独立智能体，以探究多智能体设置中的适应。在完全合作的多智能体任务中，两名玩家拥有相同的目标。这类任务通常存在多个纳什均衡（Dafoe 等人，2020年）。面对新问题时，智能体必须即时适应，达成一致，选择一个能使双方利益最大化的均衡（Christianos 等人，2022年；Hu 等人，2020年；Stone 等人，2010年）。由此产生了纯单智能体设置中所没有的各种有趣策略新现象，包括涌现式劳动分工和实体协调。这两种行为在多智能体强化学习文献中都已得到广泛研究（例如，Gronauer 和 Diepold（2022年）；Strouse 等人（2021年）；Wang 等人（2020年b）；Yang 等人（2020年））；据我们所知，我们在此首次展示，这些行为可以在留出任务上通过少样本方式于测试时涌现。我们使用虚拟自博弈生成训练任务的协作玩家（Heinrich 等人，2015年），再按照 Samvelyan 等人（2022年）的方法，使用 PLR 对其进行筛选。更多详情见表 1 和附录 D.3。

与单智能体设置类似，我们发现，在留出测试任务空间中，几乎90% 的任务都存在强有力的适应证据（图 E.1）。此外，我们通过两种方式评估由此得到的智能体在合作多智能体任务留出测试集上的表现：自博弈，以及与随机动作策略协同博弈。如图 8所示，无论是在零样本还是少样本设置下，自博弈的表现都远优于与随机动作策略协同博弈。这表明智能体正在分担解决任务所需的劳动，从而更快地（或最终成功地）完成任务，并提升共同性能。

自博弈中涌现出的社会行为示例见图 9 和 E.2。试次较少时，智能体会探索各种可能解决方案的空间，有时独立行动



(a) (b)

图 8 | **两个注意力头胜过一个**。在双人合作式留出任务的测试集上，合作式自博弈的表现优于单智能体表现。对于这项评估，我们仅考察目标和生成规则均不涉及玩家，且单个玩家即可解决的任务（1000 个测试任务中的 216 个）。为生成紫色曲线，我们针对每项任务，在与随机动作策略的协作玩家共同游戏时对 AdA 评估两次：一次作为第一位玩家，另一次作为第二位玩家；在跨任务汇总前，取两次评估中的最高分。这考虑到了在某项任务中，作为一位玩家而非另一位玩家可能具有的优势。（**a**）中位数分数。（**b**） 20th 百分位数得分。

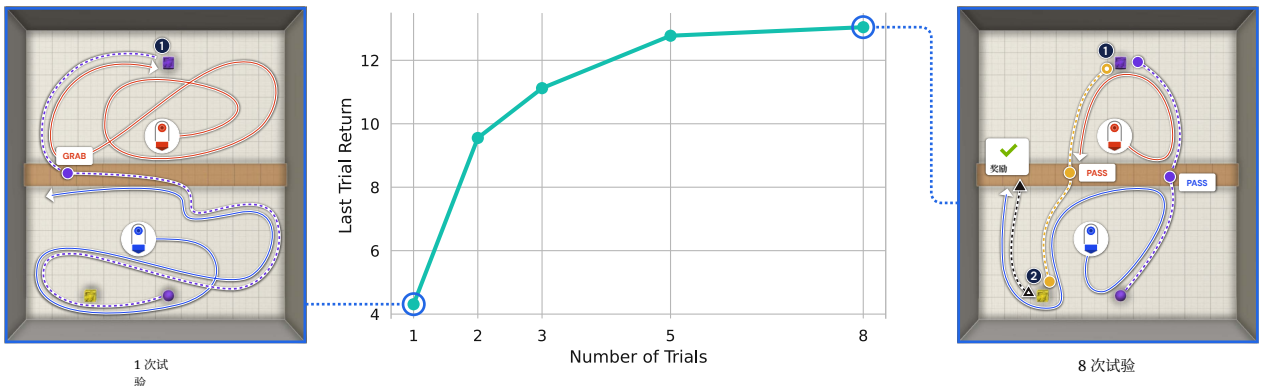
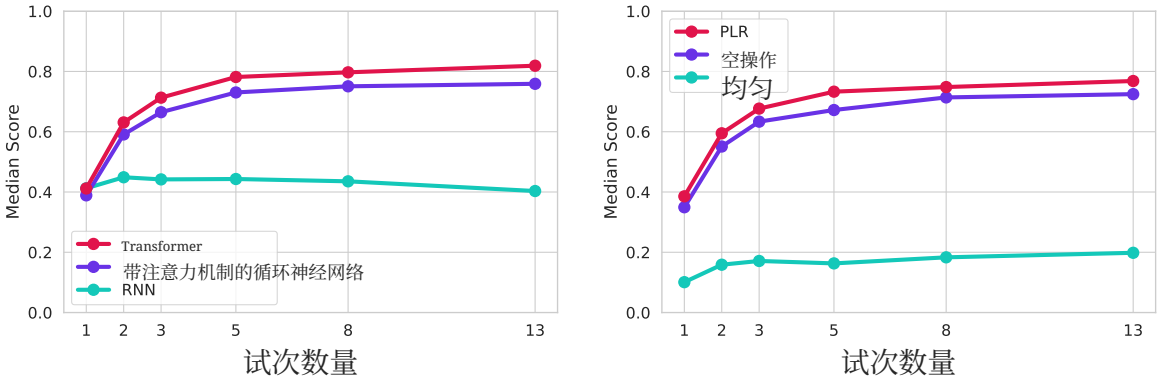


图 9 | **多智能体协同**。我们报告了 AdA 在探测任务 `Pass Over the Wall Repeatedly` 中的平均性能和代表性行为，评估采用自博弈并设置不同数量的试次。提供更多试次时，AdA 的性能会提高，展现出测试时适应能力。俯视图图像展示了总试次数不同时具有代表性的最后一次试验轨迹。相应的视频展示了情形 $k = 5$ 下一个回合内所有试次中的行为。

，有时则协同合作。试次增多后，一旦智能体找到解决方案，便会通过身体协调和分工来优化路径，从而高效地完成任务。这种行为是在测试时适应过程中涌现的，在训练期间并未受到明确激励，唯一的激励来自高层次的完全合作奖励函数。我们在 微型网站 上提供了这类行为在各种任务中的视频。



(a) 架构的影响　　(b) 课程的影响

图 10 | **(a)**不同架构选择下，随着试次数增加而产生的适应。纳入注意力模块对于实现适应至关重要，其中 Transformer-XL 架构表现最佳。**(b)** 不同课程选择下，随着试次数增加而产生的适应。与均匀采样基线相比，空操作过滤和 PLR 大幅提升了零样本泛化和少样本适应。

3.2. 架构影响性能

现在，我们通过一系列消融研究，深入了解方法中的哪些组成部分至关重要。在这些研究中，我们使用单个初始化种子，因为训练 AdA 时不同种子之间的方差较低（参见附录 F.3）。除非另有说明，所有消融实验均在单智能体设置下进行。

首先，我们通过实证比较不同架构的选择：Transformer-XL、循环神经网络和带注意力机制的循环神经网络。为实现循环神经网络，我们使用门控循环单元（Cho 等人，2014）。为便于比较，我们使所有架构的网络总规模保持一致。表 D.3展示了实验设置的详细信息。图 10a 显示，在这项比较中，Transformer-XL 是性能最佳的架构；而在循环神经网络中加入多头注意力模块，则能恢复 Transformer 的大部分性能，这凸显了注意力模块的有效性。

3.3. 自动课程学习提升性能

为确立自动课程学习的重要性，我们比较了采用第 2.3 节所述课程方法进行训练时的适应情况：空操作过滤和 PLR。图 10b展示了采用不同课程训练的智能体最后一次试验得分的中位数。空操作过滤和 PLR 课程的表现都显著优于使用均匀采样任务训练的基线。此外，PLR 优于空操作过滤，尤其是在试次数较多时；这表明，基于遗憾的课程特别有助于学习长期适应。附录 D.5详细介绍了训练配置，并比较了我们方法的样本效率。结果显示，无论以学习步数还是浮点运算次数衡量，两种自动课程方法的样本效率都高于均匀采样。

在图 11 中，我们展示了两种方法下任务复杂度的演变。在这两种情况下，起初都会优先处理较简单的任务，并逐渐形成清晰的课程。尽管两种方法都没有明确地优化这些指标以使其提升，但随着智能体能力不断增强，任务复杂度也随之增加。更多任务复杂度指标见图 D.3 。

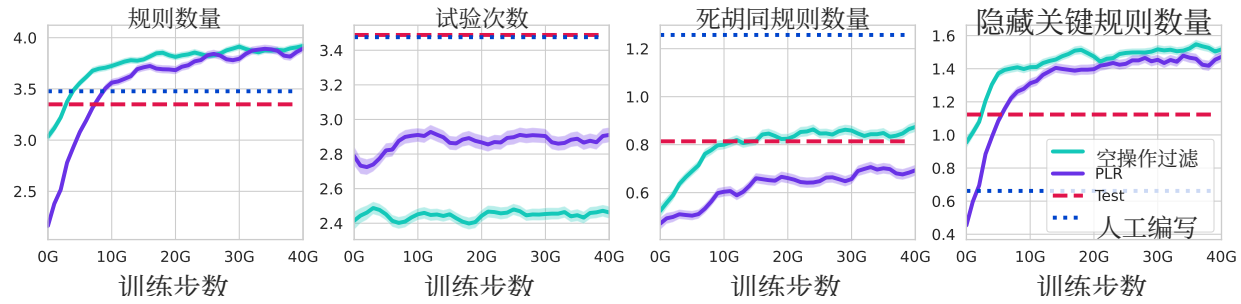
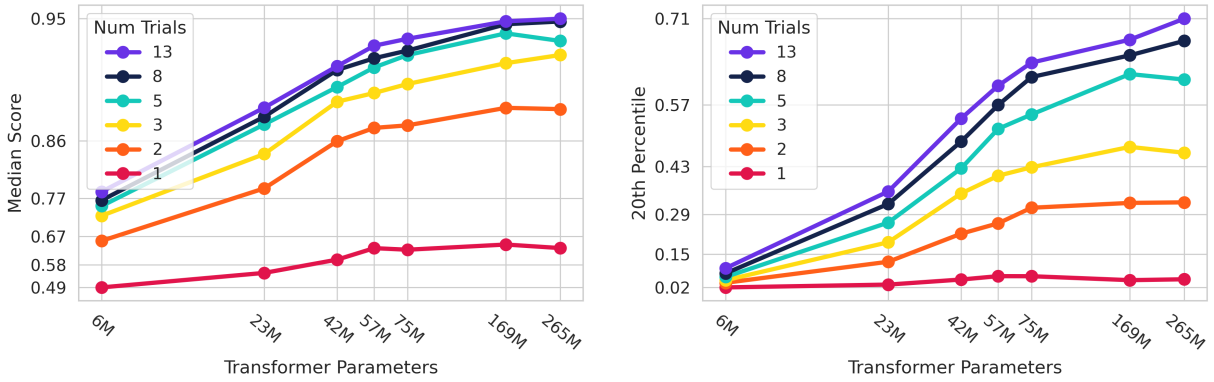


图 11 | 空操作过滤和 PLR 的涌现课程。图中展示了动态训练集的一些任务指标，这些指标是对集合中所有任务取平均所得，并以阴影表示标准误。在所有图中，指标值越高表示任务难度越大。例如，规则数量更多的任务需要更多试错才能找到要触发的正确规则。水平线表示在测试和人工编写的评估任务集上取平均后的相同指标值，分别以虚线和点线表示。



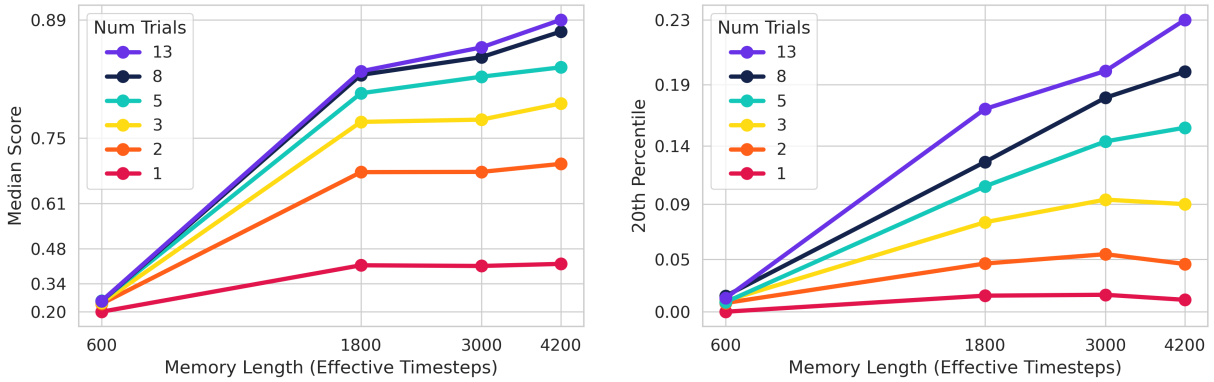
(a) (b)

图 12 | 扩展 Transformer 参数可同时提高中位数**(a)** 和第20百分位数**(b)** 测试得分。两个坐标轴均按对数尺度缩放，分别依据函数 $\log(x)$ 和 $-\log(1-y)$；在此尺度下，模型规模与性能之间的关系看起来大致呈线性。评估的试次数越多，斜率越陡，这表明扩展模型对于促进更强的适应特别有效，而非增强零样本泛化。

3.4. 扩展智能体可提升性能

能够良好扩展的方法对于机器学习的持续进步至关重要，而了解方法如何扩展，对于决定未来应将时间和计算资源投入何处也很重要。许多基础模型都已确定了缩放定律（见第 4节）：性能与模型规模及其他因素之间呈幂律关系，在双对数图上表现为线性关系。受此类分析启发，我们研究适应如何随 Transformer 模型规模和记忆长度扩展。

扩展网络规模。我们展示了 AdA 的 Transformer 模型性能如何随模型规模扩展，并对表D.8所示的模型规模进行了实验。在研究模型规模的缩放定律时，我们遵循Kaplan 等人(2020年的做法，只测量 Transformer（即非嵌入）参数，其数量跨越3 个数量级，从6M 到265M 个 Transformer 参数（即总参数从41M 到533M）。完整的超参数列表见表D.9。



(a) (b)

图 13 | 扩展 Transformer-XL 的记忆长度会同时提高中位数（**a**）和第20百分位数（**b**）测试得分。两个坐标轴均采用对数尺度，分别依据函数 $\log(x)$ 和 $-\log(1-y)$ 绘制；在此尺度下，记忆长度与性能之间的关系大致呈线性。评估的试次数越多，斜率越陡，表明扩展记忆尤其有助于促进更强的少样本适应。

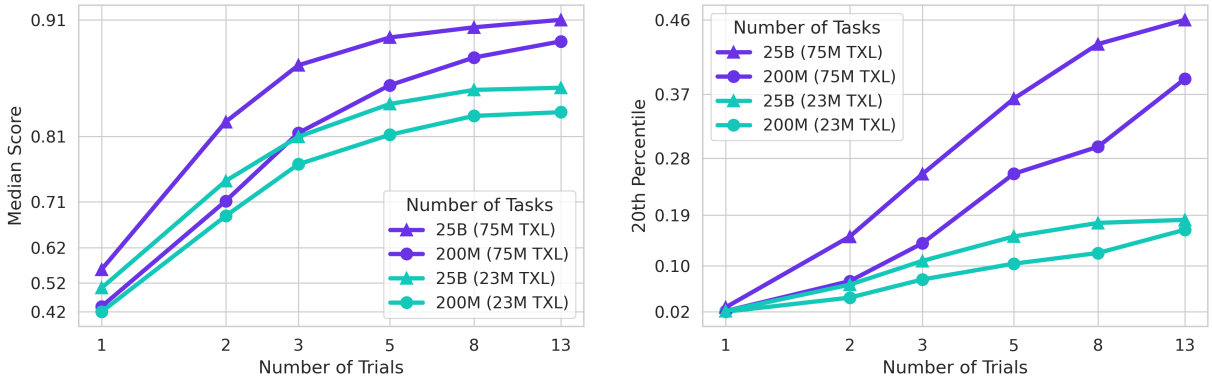
图12 显示，网络规模越大，性能越好，尤其是在测试时提供更多试次以进行适应时。虽然较大的模型似乎有助于提高测试集得分的中位数（图12a），但模型规模对测试集较低百分位数的影响尤为显著（图12b）。这表明，较大的模型能让智能体将其适应能力泛化到更广泛的任务。双对数图中模型规模与性能之间大致呈线性关系，这体现了幂律缩放关系，尽管结果只涵盖了两个到三个数量级。曲线并非完全线性，可能有几个原因：我们尚未训练至收敛（不过所有模型的性能提升都已放缓），而且在所有模型规模的实验中，我们都使用了一个含 23M 个参数的蒸馏教师。

附录 D.7 详述了各种模型规模的计算成本，并展示了经浮点运算次数调整后的结果。对于相同的训练步数，大模型的零样本得分和适应能力确实优于小模型，且样本效率更高；但考虑计算成本后，最大模型未必总是最佳选择。

扩展记忆长度。 性能也会随着 AdA 记忆长度的增加而提升。实验设置见表D.10，我们考察缓存的先前网络激活值数量，取值范围为 100 至700；采用 6 个 Transformer-XL 模块时，有效时间步范围为 600 至 4200个时间步。[4]

图 13 显示，与模型规模一样，扩展记忆长度有助于提升性能，尤其是在测试百分位数较低时，能够提高分布尾部的性能。对于我们的任何任务，试验时长上限都是 300 个时间步，因此值得注意的是，例如，5 次试验（1500 个时间步）上的性能仍会随着“有效记忆长度”从 1800 增至 4200 而不断提升。这表明，Transformer-XL 更容易利用明确提供的记忆激活值，而不是依赖这些激活值中隐含的、理论上范围更长的信息。

[4]Transformer-XL 通过在每次前向传播时，将缓存的先前注意力层输入记忆拼接到键和值上，从而支持使用更长、长度可变的上下文窗口。由于中间层的输入是来自前一层的激活值，而这些激活值本身包含有关过去的信息，因此从理论上讲，缓存 M 激活值可实现 $M \times L$ 的有效记忆范围，其中 L 是网络中的注意力层数量。



(a) (b)

图 14 | 中位数（**a**）和 20th 百分位数（**b**）适应尺度随任务池大小的变化。对于较大的模型，这种影响尤为显著。与其他扩展实验一样，我们以对数刻度显示 y 轴。这里，我们在 x 轴上绘制试次数，并考察两种任务分布曲线之间的差距（三角形标记与圆形标记）。

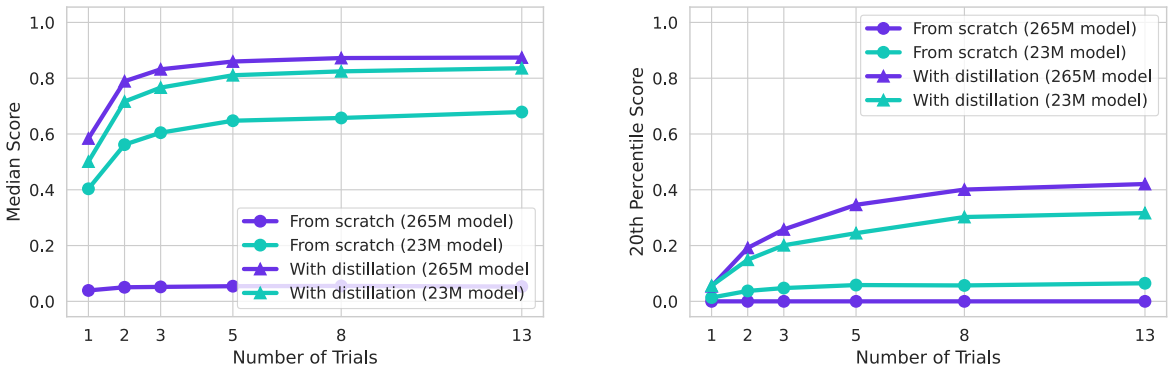
3.5. 扩展任务池可提升性能

另一个重要的扩展因素是模型训练所用的数据量。例如，Hoffmann 等人 (2022年) 表明，要充分发挥扩展语言模型的效益，训练数据量必须与参数数量以相同的速率扩展。在我们的研究中，相关数据来自与不同任务的交互，因此我们考察了扩展 XLand 池中不同任务的数量和复杂度所带来的影响。

任务池规模的扩展。这里，我们考察自动课程可从中采样的训练任务数量变化所产生的影响。回顾一下，在 XLand 中，任务是一个世界（地形和物体的物理布局）与一个游戏（指定目标和生成规则）的组合。我们研究了从由 2 亿个不同任务组成的小型池中采样任务进行训练的效果（4,000 个世界 $\times$ 50,000 个游戏），并将其与从由 250 亿个不同任务组成的大型池中采样任务进行训练的效果进行比较（50,000 个世界 $\times$ 500,000 个游戏）。表D.11展示了这些比较的完整实验设置。

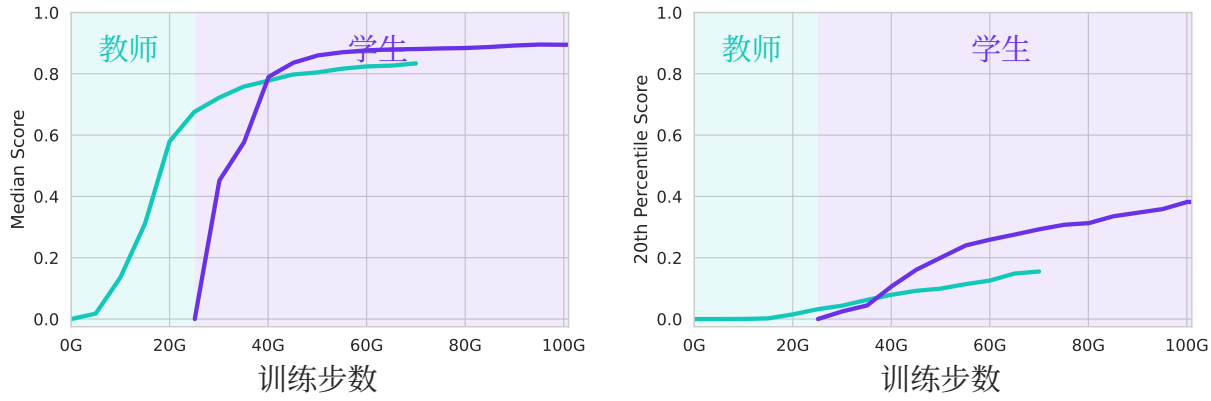
图 14 显示，在更大的任务池上，规模相同的模型取得了更高的测试得分。与其他扩展实验一样，我们尤其看到 20th 百分位数上的性能有所提升。结果展示了两种不同规模的模型；扩展任务池规模时，较大的 Transformer 带来的差距更大。这表明，大模型尤其容易在较小的任务池上过拟合。

任务池复杂度的扩展。我们还可以沿着另一个维度扩展方法，即任务分布的整体复杂度。例如，平均而言，在平坦地形上的任务比在地形变化的环境中的任务更容易解决。在图E.3中，我们通过比较在两个规模相同但复杂度不同的分布上训练的智能体的模型扩展效果，并分别在相应的测试集上进行评估，展示了较低的环境复杂度可能成为扩展的瓶颈。因此，环境复杂度无上限的开放式设置（例如多智能体系统）对于扩展自适应智能体可能尤为重要。



(a) (b)

图 15 | 从零开始训练或通过蒸馏热启动时，随着试次数增加，模型的适应情况。模型分别具有 23M 和 265M 个 Transformer 参数。圆形标记表示从零开始训练，三角形标记表示通过 40 亿帧的蒸馏热启动训练。在这项消融实验中，智能体按照第 3.1 节所述的多智能体设置进行训练，并在总计 220 亿训练帧后于多智能体测试集上进行评估。 **(a)** 中位数得分。 **(b)**20th 百分位数得分。



(a) (b)

图 16 | 使用 23M 参数的 Transformer-XL 时的 $k = 13$ 标准化最后一次试验得分。教师从零开始训练，而其他方面完全相同的学生则通过教师的快照进行蒸馏；该快照是在训练 250 亿步后截取的。 x轴统计经验总量，其中包括用于训练教师的经验。对比结果表明，即使经验总量和更新次数相同，蒸馏也能大幅提升学生的性能。这一点在中位数得分 **(a)** 上成立，而在第 20th 百分位数上 **(b)** 更为明显。

3.6. 蒸馏提升性能并助力智能体扩展 s

上一节展示的所有规模比较实验，在训练开始阶段的前若干帧中都使用了相同的蒸馏教师，详见附录 D.6。现在，我们来探讨蒸馏在扩展中的作用。简而言之，我们发现，在扩大模型规模时，先通过一段时间的蒸馏来启动训练至关重要。如图 15所示，不使用蒸馏训练一个265M 参数的 Transformer 模型，其性能远不如以相同方式训练、规模小得多的 23M 参数 Transformer。然而，如果在最初的 40 亿训练帧中，使用一个 23M 参数教师进行蒸馏训练，那么 265M 模型的表现就会明显优于23M 版本。参见

附录中的实验细节D.11。

此外，我们发现，即使学生和教师的模型规模相同，在总帧数预算固定的情况下，蒸馏也能带来显著提升（图16）。我们推测，这是因为学生智能体在早期学到了较差的表征（Cetin 等人，2022年；Nikishin 等人，2022年），而蒸馏可以避免这一问题。这也与离线强化学习中的发现一致：要有效扩展模型，往往需要额外数据（Reid 等人，2022年）。蒸馏的效果在第一轮最大，后续轮次的收益则递减（图E.5）。

3.7. 使用跳跃记忆进行更多试次的训练可实现多样本适应

到目前为止，我们考察的是少样本场景：训练使用 1 至 6 个试次，评估最多使用 13 个试次。本节将评估 AdA 在更长时间跨度内的适应能力。我们发现，使用 $k \in \{1,2,\ldots 6\}$训练时，智能体在超过 13 个试次后不再继续适应；然而，将训练期间的最大试次数提高到 24，并相应增加记忆的有效长度，可显著提升这种长期适应能力。这些结果表明，我们的方法能够自然扩展至多样本时间尺度，回合时长超过30分钟。[5] 本节分别对这两个因素进行消融实验，并证明二者对于长程适应都至关重要。训练配置（除训练步数外，与记忆扩展实验相同）详见表D.14。

正如我们在第 3.4 节中所指出的，增加记忆长度会提升容量，即使整个回合都能装入记忆，这也有利于智能体，但同时也会增加计算成本。为了区分这些因素，我们对第 2.4 节所述的记忆架构做了一项简单改动，在不增加计算成本的情况下延长有效记忆长度。我们使用门控循环单元（GRU）对轨迹进行编码，然后再将其输入 Transformer-XL。这样便可以对编码后的轨迹进行时间步下采样，使智能体能够在无需额外计算的情况下关注多达 4 倍数量的试次。我们发现，单独增加 GRU 并不会显著影响智能体的性能。

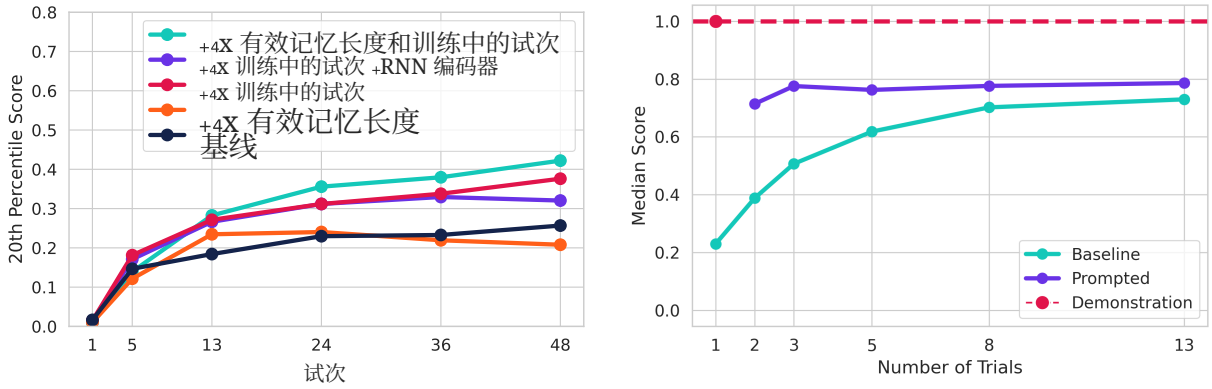
如图 17a 所示，增加训练分布中的试次数量，能显著提升智能体在后续试次中的性能；当相应地增加记忆长度时，效果尤为明显。换言之，AdA 学到的适应策略受益于经历大量试次，而非仅仅经历最近的试次。因此我们可以得出结论：AdA 能够基于跨多个试次整合到记忆中的长期知识进行适应，而不仅仅是编码一种只依赖前一个试次轨迹的简单元策略。

即使记忆没有增加，增加训练中的试次也能带来更好的适应。这表明，给予智能体更长的训练回合（其中包含更多试次），它就能够学会更好的探索和改进策略。请注意，如果不增加训练试次的数量，仅增加有效记忆并不能提升性能，因为智能体并未接受过如何利用额外记忆容量的训练。

3.8. AdA 可以利用第一人称示范进行提示

为了确定 AdA 是否能够从第一人称示范中进行零样本学习，我们向它提供了一段由微调后的教师完成的第一人称演示作为提示，具体如下。教师在第一次试验中控制化身，而 AdA 继续像往常一样接收观测，并以此作为其

[5] 48 次 40秒任务持续32分钟。相比之下，星际争霸2游戏的平均时长为 10 至 15分钟，而且 AlphaStar 每秒采取行动的频率低于 AdA（Vinyals 等人，2019年）。



(a) (b)

图 17 | **（a）**消融实验：改变训练试验的最大数量（从 $k = 6$ 基线到 $k = 24$），并通过子采样增加有效记忆大小（从 1800 步增至 7200 步）时，测试得分的 20th 百分位数。综合这些因素，智能体能够在更多试次中进行适应（持续时间超过 30 分钟）。增加训练试验数量的影响最大，也是子采样发挥作用的前提。此外，该图还显示，添加 RNN 编码器以促进子采样，本身并不会显著影响性能。 **（b）**在每个回合的第一次试验中，通过第一人称演示提示 AdA 后，其人工编写任务得分的中位数，与无提示基线的对比。提示后的得分严格高于基线，这表明 AdA 能够利用演示提示中的信息来提升性能。然而，其得分低于演示本身的得分，这说明它无法充分利用演示。

Transformer 记忆的条件。随后，AdA 得以在剩余试次中独立继续运行，并按常规方式记录其得分。

图 17b 展示了在手工编写测试集上，经过提示的 AdA 与无提示基线的中位数得分对比。在中位数任务的第二次试验中，经过提示的 AdA 无法完全模仿教师演示，得分因此下降。不过，在所有试次数量下，它的表现都优于无提示基线，这表明它能够有效地将演示中的信息融入其策略。这一过程类似于对大语言模型进行提示：智能体的记忆会以期望行为的示例作为引导，并据此继续行动。值得注意的是，AdA 从未接受过此类离策略第一人称示范的训练，但其上下文学习算法仍能够泛化至这些示范。

在图 F.4 中，我们展示了所有单智能体手工编写任务的提示结果，并讨论提示在哪些情况下有效。在附录 F.4中，我们还展示了早期结果，探究在部分任务中使用人类演示进行提示的效果。这些结果表明，提示在某些情况下能取得令人瞩目的成功，但也证实，人类演示无法克服 AdA 任务分布本身的局限性。两个视频对比了在任务 `Object permanence: yellow cube`上进行提示和不进行提示时的行为。

4. 相关工作

在这项工作中，我们利用基于注意力的模型方面的进展，在开放式任务空间中进行元学习。我们的智能体学习一种上下文学习强化学习算法，同时自动构建训练任务分布；因此，我们结合了*AI* 生成算法(AI-GA, Clune

(2019年））。与我们最相似的工作是OEL Team 等人(2021年），该研究同样探讨了在庞大的多智能体任务空间中，利用自动课程和代际学习进行训练。我们工作的一个关键区别在于，我们关注适应（而非零样本性能），并使用大型 Transformer 模型。Akkaya 等人(2019年）也展示了自适应课程在元学习用于控制机器人手的策略时的有效性；不过，他们聚焦于特定的仿真到现实设置，而非能力更为通用的智能体。现在，我们将依次总结与本工作各个组成部分相关的文献。

程序化环境生成。我们利用程序化内容生成（PCG）来生成规模庞大且多样化的任务分布。游戏社区多年来一直在研究 PCG（Risi 和 Togelius，2020；Togelius 和 Schmidhuber，2008），近来它也被用于为强化学习智能体创建测试平台（Cobbe 等人，2018；Justesen 等人，2018；Raileanu 和 Rocktäschel，2020）。事实上，过去几年中，人们提出了一系列具有挑战性的程序化内容生成环境（Chevalier-Boisvert 等人，2018；Cobbe 等人，2020；Deitke 等人，2022；Hafner，2022；Juliani 等人，2019；Küttler 等人，2020；Samvelyan 等人，2021），其重点大多在于测试和提升强化学习中的泛化能力（Bhatt 等人，2022；Fontaine 等人，2021；Kirk 等人，2021）。近来，人们愈发重视开放式世界：Albrecht 等人（2022）提出了 Avalon，这是一个支持复杂任务的 3D 世界；而 Minecraft（Johnson 等人，2016）则被提议作为开放式演化与强化学习的挑战（Fan 等人，2022；Grbic 等人，2021；Kanervisto 等人，2022）。然而，与 XLand 不同，Minecraft 无法对整个仿真栈进行控制，因此限制了任务空间的平滑性。

开放式学习。近期一系列研究展示了智能体与环境在任务分布上的共同适应的有效性（Parker-Holder 等，2022；Wang 等人，2019，2020a）。我们的方法与无监督环境设计（UED，Dennis 等人（2020））范式相似，因为我们旨在训练通才智能体，而不让其预先了解测试任务。这一领域的先驱方法之一是 PAIRED（Dennis 等人，2020），它尝试通过强化学习训练的对抗者生成任务。我们以优先级关卡回放（Jiang 等，2021a,b）为基础；该方法会筛选出随机采样且遗憾值较高的环境。我们的工作也与课程学习相关（Campero 等人，2021；Fang 等人，2021；Matiisen 等人，2020；Mu 等人，2022；OpenAI 等人，2021；Portelas 等人，2019；Sukhbaatar 等人，2018），关键区别在于，这些方法通常预设了特定的下游目标或任务。也有研究通过围绕协同玩家构建自动课程来训练智能体，不过这类研究通常聚焦于单例环境（Berner 等人，2019；Vinyals 等，2019）或均匀采样任务（Baker 等人，2020；文化通用智能团队等人，2022；Jaderberg 等人，2019；Liu 等人，2019）。与 XLand 2.0 的产生式规则系统相似，Zhong 等人（2020）训练智能体，使其能够泛化到未观测到的环境动态。不过，他们研究的是零样本泛化：智能体必须从语言描述中推断环境底层动态；而 AdA 智能体则通过在多个试次中即时进行假设驱动探索，在测试时发现这些规则。

适应。 本研究聚焦于控制问题中的少样本适应，通常将其表述为 元强化学习。我们聚焦于 基于记忆的元强化学习，并以 Duan 等人(2017年) 和 Wang 等人(2016年) 的研究为基础。他们指出，如果智能体能够观测到奖励和终止信号，且记忆不会重置，那么基于记忆的策略便可以实现一种学习算法。事实证明，这是一种有效的方法，能够学习贝叶斯最优策略（Mikulik 等人，2020年；Ortega 等人，2019年），并且可能具有神经学对应机制（Wang 等人，2018年）。事实上，我们的智能体能够学习概念性探索策略；若采用元梯度方法，则外层学习器需要估计内层学习器的回报才能实现这一点（Stadie 等人，2018年）。这一领域中的解决方案

要么依赖方差较高的蒙特卡洛回报（Garcia 和 Thomas，2019；Stadie 等人，2018；Vuorio 等人，2021），要么依赖历史相关的估计器（Zheng 等人，2020）。我们的工作也受到 Alchemy（Wang 等人，2021）的启发，这是一个元强化学习基准领域，其机制启发了我们工作中的生成规则。该研究的作者采用小型 Transformer 进行基于记忆的元强化学习，但发现智能体的性能仅略优于随机启发式。研究还表明，Transformer 在简单领域的元强化学习（Melo，2022）以及利用离线数据学习强化学习算法（Laskin 等人，2022）方面也很有效。其他元强化学习方法包括元梯度（Andrychowicz 等人，2016；Finn 等人，2017；Flennerhag 等人，2022；Xu 等人，2018），这种方法效率较高，但往往存在不稳定性和短视性问题（Flennerhag 等人，2022；Metz 等人，2021；Vuorio 等人，2021）；以及基于潜变量的方法（Finn 等人，2018；Humplik 等人，2019；Rakelly 等人，2019；Zintgraf 等人，2019）。适应在机器人学中也发挥着关键作用，智能体经过训练后能够适应不同的地形（Clavera 等人，2019；Kumar 等人，2021）或受损关节（Cully 等人，2015）。

强化学习中的 Transformer 及其他领域。Transformer 架构近期已被证明对离线强化学习非常有效（Chen 等人，2021年；Janner 等人，2021年；Reed 等人，2022年），但在在线环境中的成功仍然有限。少数成功训练基于Transformer的策略的工作之一是 Parisotto 等人（2020年），他们在一个更简单、规模更小的环境中引入了多种启发式方法，以稳定训练过程。事实上，尽管我们采用了类似的 Transformer-XL架构（Dai 等人，2019年；Vaswani 等人，2017年），但我们展示了在线元强化学习中的缩放定律，这些定律与其他领域中观察到的规律相似，例如语言领域（Brown 等，2020年；Devlin 等人，2019年；Kaplan 等人，2020年；Rae 等人，2021年）。类似地，Melo（2022年）在规模更小的元强化学习环境中使用 Transformer 进行快速适应，将自注意力机制解释为一种通过递归应用 Transformer层、利用时间步嵌入构建情景记忆的方式。Transformer 架构也被用于强化学习之外的元学习，例如学习通用算法（Kirsch 等人，2022年）或超参数优化器（Chen 等人，2022年）。Transformer 还普遍存在于现代大语言模型中，而研究已表明，这些模型是少样本学习器（Brown 等，2020年）。

5. 结论

在不同时间尺度上适应新信息，是通用智能体的一项关键能力。基础模型尤其展现出了获取大量知识信息并迅速将其应用于新场景的能力。迄今为止，它们主要依赖监督学习和自监督学习，因此需要获取大型数据集。收集数据集的一种替代方案，是让智能体通过强化学习从自身经验中学习，前提是能够提供足够丰富的物理世界或开放式模拟。这引出了一个问题：能否通过强化学习训练出大规模、具备普遍适应能力的模型?

据我们所知，本文首次展示了一个经强化学习训练的智能体，它能够在广阔的开放式任务空间中进行快速的上下文内适应，其适应时间尺度与人类玩家相近。这个自适应智能体 (AdA)以结构化的方式探索留出任务，仅通过与任务进行少量交互，就能不断优化其策略，使之趋近最优行为。此外，AdA 支持情境化第一人称提示，从而增强其少样本性能，这与大语言模型中的提示类似。AdA 的性能可随参数数量、上下文长度以及训练任务分布的丰富程度而扩展。

我们的训练方法基于黑盒元强化学习，这种方法此前被认为难以扩展。我们表明，最先进的自动课程技术可以塑造数据分布，从而提供

足够的信号，使智能体能够在开放式任务空间中学会学习。此外，我们证明，基于注意力的架构能够比纯循环网络更有效地利用这些信号，这表明数据分布与智能体架构的协同适应对于促进快速适应至关重要。最后，蒸馏使我们得以实现大规模 Transformer 架构的潜力。

人工智能研究的未来必然涉及训练规模日益庞大、能力日益通用且适应性不断增强的模型。为此，我们提供了训练一个5亿参数模型的方法，希望能为强化学习与基础模型交汇领域的进一步发展铺平道路。AdA 展现出快速且可扩展的多种适应能力，涵盖工具使用与实验、劳动分工与导航。鉴于扩展定律的趋势，这类模型未来或将成为基础模型，用于在现实世界中针对实用控制问题进行少样本适应和微调。

6. 作者与贡献

作者按姓氏字母顺序排列。如需联系，请致信 Feryal Behbahani（feryal@deepmind.com）和 Edward Hughes（edwardhughes@deepmind.com）。

6.1. 核心贡献者

- **Jakob Bauer**：技术领导、课程研究、基础设施工程、任务撰写、论文写作

- **Kate Baumli**：智能体研究、扩展、智能体分析、任务编写、论文撰写
- **Feryal Behbahani**：研究愿景、团队领导、智能体研究、论文撰写
- **Avishkar Bhoopchand**：技术领导、评估研究、基础设施工程、任务编写、论文撰写

- **Michael Chang**：可视化、智能体分析、人类实验
- **Adrian Collister**：XLand 开发、人类实验
- **Edward Hughes**：研究愿景、团队领导、评估研究、论文撰写
- **Sheleem Kashem**：基础设施工程、课程研究、人类实验
- **Jack Parker-Holder**：课程研究、论文撰写
- **Yannick Schroecker**：智能体研究、扩展、任务编写、智能体分析、论文撰写
- **Jakub Sygnowski**：基础设施工程、课程研究、智能体分析、论文撰写

- **Alexander Zacherl**：设计领导、智能体分析、任务编写、可视化、人类实验

- **Lei Zhang**：课程研究、智能体分析、论文撰写

6.2. 部分贡献者

- **Nathalie Bradley-Schmieg**：项目管理
- **Natalie Clay**：质量保证测试、人类实验
- **Vibhavari Dasagi**：评估研究
- **Lucy Gonzalez**：项目管理
- **Karol Gregor**：智能体研究
- **Maria Loks-Thompson**：XLand 开发、人类实验
- **Hannah Openshaw**：项目管理
- **Shreya Pathak**：智能体分析

- **Nicolas Perez-Nieves**：智能体分析、任务编写
- **Nemanja Rakicevic**：课程研究、智能体分析
- **Tim Rocktäschel**：战略建议、论文撰写
- **Sarah York**：质量保证测试、人类实验

6.3. 赞助方

- **Satinder Baveja**：战略建议
- **Karl Tuyls**：战略建议

7. 致谢

我们感谢 Max Jaderberg 在项目愿景方面提供的早期指导。我们感谢 Wojciech Marian Czarnecki 提供的早期版本产生式规则形式主义，以及 Catarina Barros 提供的原型实现。我们感谢 Dawid Górny 在实现可视化工具方面给予的支持。我们感谢 Alex Platonov 为图示及配套视频制作的艺术化呈现。我们感谢 Nathaniel Wong、Tom Hudson 和 Worlds Team 提供的工程支持。此外，我们感谢 Andrew Bolt、Max Cant、Valentin Dalibard、Richard Everett、Nik Hemmings、Shaobo Hou、Jony Hudson、Errol King、George-Cristian Muraru、Alexander Neitz、Valeria Oliveira、Doina Precup、Drew Purves、Daniel Tanis、Roma Patel 和 Marcus Wainwright 提供的宝贵讨论和支持。我们感谢 Sebastian Flennerhag 和 Raia Hadsell 审阅论文草稿。

参考文献

R. Agarwal、M. Schwarzer、P. S. Castro、A. C. Courville 和 M. G. Bellemare。深度强化学习处于统计悬崖边缘。 *CoRR*，abs/2108.13264，2021年。I. Akkaya、M. Andrychowicz、M. Chociej、M. Litwin、B. McGrew、A. Petron、A. Paino、M. Plappert、G. Powell、R. Ribas 等人。用机器人手解魔方。 *arXiv* 预印本 *arXiv:1910.07113*，2019年。J. Albrecht、A. J. Fetterman、B. Fogelman、E. Kitanidis、B. Wróblewski、N. Seo、M. Rosenthal、M. Knutins、Z. Polizzi、J. B. Simon 和 K. Qiu。Avalon：一个使用程序生成世界评估强化学习泛化能力的基准。在 第三十六届神经信息处理系统大会数据集与基准赛道，2022年。M. Andrychowicz、M. Denil、S. Gomez、M. W. Hoffman、D. Pfau、T. Schaul、B. Shillingford 和 N. De Freitas。通过梯度下降来学习梯度下降。 神经信息处理系统进展，第29卷，2016年。I. Babuschkin、K. Baumli、A. Bell、S. Bhupatiraju、J. Bruce、P. Buchlovsky、D. Budden、T. Cai、A. Clark、I. Danihelka、C. Fantacci、J. Godwin、C. Jones、R. Hemsley、T. Hennigan、M. Hessel、S. Hou、S. Kapturowski、T. Keck、I. Kemaev、M. King、M. Kunesch、L. Martens、H. Merzic、V. Mikulik、T. Norman、J. Quan、G. Papamakarios、R. Ring、F. Ruiz、A. Sanchez、R. Schneider、E. Sezener、S. Spencer、S. Srinivasan、L. Wang、W. Stokowiec 和 F. Viola。DeepMind JAX 生态系统，2020年。网址 http://github.com/deepmind。B. Baker、I. Kanitscheider、T. Markov、Y. Wu、G. Powell、B. McGrew 和 I. Mordatch。多智能体自动课程中涌现的工具使用。在 国际学习表征会议，2020年。

D. Balduzzi、K. Tuyls、J. Perolat 和 T. Graepel。重新评估评估方法。 神经信息处理系统进展，第31卷，2018年。J. Barbosa、H. Stein、S. Zorowitz、Y. Niv、C. Summerfield、S. Soto-Faraco 和 A. Hyafil。在实验室研究人类行为的实用指南。 行为研究方法，第1–19页，2022年。C. Berner、G. Brockman、B. Chan、V. Cheung、P. Debiak、C. Dennison、D. Farhi、Q. Fischer、S. Hashme、C. Hesse、R. Józefowicz、S. Gray、C. Olsson、J. Pachocki、M. Petrov、H. P. de Oliveira Pinto、J. Raiman、T. Salimans、J. Schlatter、J. Schneider、S. Sidor、I. Sutskever、J. Tang、F. Wolski 和 S. Zhang。使用大规模深度强化学习玩 Dota 2。*CoRR*，abs/1912.06680，2019年。V. Bhatt、B. Tjanaka、M. C. Fontaine 和 S. Nikolaidis。深度代理辅助环境生成。在 神经信息处理系统进展，2022年。R. Bommasani、D. A. Hudson、E. Adeli、R. Altman、S. Arora、S. von Arx、M. S. Bernstein、J. Bohg、A. Bosselut、E. Brunskill、E. Brynjolfsson、S. Buch、D. Card、R. Castellon、N. S. Chatterji、A. S. Chen、K. Creel、J. Q. Davis、D. Demszky、C. Donahue、M. Doumbouya、E. Durmus、S. Ermon、J. Etchemendy、K. Ethayarajh、L. Fei-Fei、C. Finn、T. Gale、L. Gillespie、K. Goel、N. D. Goodman、S. Grossman、N. Guha、T. Hashimoto、P. Henderson、J. Hewitt、D. E. Ho、J. Hong、K. Hsu、J. Huang、T. Icard、S. Jain、D. Jurafsky、P. Kalluri、S. Karamcheti、G. Keeling、F. Khani、O. Khattab、P. W. Koh、M. S. Krass、R. Krishna、R. Kuditipudi 等人。基础模型的机遇与风险。*CoRR*，abs/2108.07258，2021年。J. Bradbury、R. Frostig、P. Hawkins、M. J. Johnson、C. Leary、D. Maclaurin、G. Necula、A. Paszke、J. Van- derPlas、S. Wanderman-Milne 和 Q. Zhang。JAX：Python 的可组合变换+NumPy 程序，2018年。网址 http://github.com/google/jax。T. Brown、B. Mann、N. Ryder、M. Subbiah、J. D. Kaplan、P. Dhariwal、A. Neelakantan、P. Shyam、G. Sastry、A. Askell、S. Agarwal、A. Herbert-Voss、G. Krueger、T. Henighan、R. Child、A. Ramesh、D. Ziegler、J. Wu、C. Winter、C. Hesse、M. Chen、E. Sigler、M. Litwin、S. Gray、B. Chess、J. Clark、C. Berner、S. McCandlish、A. Radford、I. Sutskever 和 D. Amodei。语言模型是少样本学习器。收录于 H. Larochelle、M. Ranzato、R. Hadsell、M. Balcan 和 H. Lin 编，神经信息处理系统进展，第33卷，第1877–1901页。Curran Associates, Inc.，2020年。A. Campero、R. Raileanu、H. Kuttler、J. B. Tenenbaum、T. Rocktäschel 和 E. Grefenstette。使用 AMIGo 学习：对抗性驱动的内在目标。收录于 国际学习表征会议，2021年。M. Carroll、R. Shah、M. K. Ho、T. L. Griffiths、S. A. Seshia、P. Abbeel 和 A. Dragan。探讨学习人类对于人类与人工智能协调的效用，2019年。E. Cetin、P. J. Ball、S. Roberts 和 O. Celiktutan。稳定基于像素的离策略深度强化学习。收录于 K. Chaudhuri、S. Jegelka、L. Song、C. Szepesvari、G. Niu 和 S. Sabato 编，第*39*届国际机器学习会议论文集，第162卷，机器学习研究论文集，第2784–2810页。PMLR，2022年7月17日至23日。L. Chen、K. Lu、A. Rajeswaran、K. Lee、A. Grover、M. Laskin、P. Abbeel、A. Srinivas 和 I. Mordatch。决策变换器：通过序列建模实现强化学习。收录于 M. Ranzato、A. Beygelz- imer、Y. Dauphin、P. Liang 和 J. W. Vaughan 编，神经信息处理系统进展，第34卷，2021年。Y. Chen、X. Song、C. Lee、Z. Wang、Q. Zhang、D. Dohan、K. Kawakami、G. Kochanski、A. Doucet、M. Ranzato、S. Perel 和 N. de Freitas。探索借助 Transformer 学习通用超参数优化器。收录于*2022*年神经信息处理系统大会（*NeurIPS*），2022年。

M. Chevalier-Boisvert、L. Willems 和 S. Pal。用于 OpenAI Gym 的极简网格世界环境。https://github.com/maximecb/gym-minigrid，2018年。K. Cho、B. Van Merriënboer、D. Bahdanau 和 Y. Bengio。神经机器翻译的属性：编码器-解码器方法。*arXiv* 预印本 *arXiv:1409.1259*，2014年。F. Christianos、G. Papoudakis 和 S. V. Albrecht。用于多智能体强化学习均衡选择的帕累托演员-评论家。*arXiv*，2022年。doi：10.48550/ARXIV.2209.14344。I. Clavera、A. Nagabandi、S. Liu、R. S. Fearing、P. Abbeel、S. Levine 和 C. Finn。通过元强化学习，学习在动态真实世界环境中适应。载于国际学习表征会议，2019年。J. Clune。AI-GAs：人工智能生成算法——一种用于产生通用人工智能的替代范式。*CoRR*，abs/1905.10985，2019年。K. Cobbe、O. Klimov、C. Hesse、T. Kim 和 J. Schulman。量化强化学习中的泛化。*CoRR*，abs/1812.02341，2018年。K. Cobbe、C. Hesse、J. Hilton 和 J. Schulman。利用程序化生成对强化学习进行基准测试。载于第*37*届国际机器学习会议论文集，第2048–2056页，2020年。A. Cully、J. Clune、D. Tarapore 和 J.-B. Mouret。能够像动物一样适应的机器人。《自然》，521：503–507，2015年。文化通用智能团队、A. Bhoopchand、B. Brownfield、A. Collister、A. D. Lago、A. Edwards、R. Everett、A. Frechette、Y. G. Oliveira、E. Hughes、K. W. Mathewson、P. Mendolicchio、J. Pawar、M. Pislar、A. Platonov、E. Senter、S. Singh、A. Zacherl 和 L. M. Zhang。在不使用人类数据的情况下学习稳健的实时文化传播，2022年。W. M. Czarnecki、R. Pascanu、S. Osindero、S. Jayakumar、G. Swirszcz 和 M. Jaderberg。蒸馏策略蒸馏。载于第二十二届国际人工智能与统计会议，第1331–1340页。PMLR，2019年。A. Dafoe、E. Hughes、Y. Bachrach、T. Collins、K. R. McKee、J. Z. Leibo、K. Larson 和 T. Graepel。协作式人工智能中的开放问题。*CoRR*，abs/2012.08630，2020年。Z. Dai、Z. Yang、Y. Yang、J. Carbonell、Q. Le 和 R. Salakhutdinov。Transformer-XL：超越固定长度上下文的注意力语言模型。载于计算语言学协会第*57*届年会论文集，第2978–2988页，意大利佛罗伦萨，2019年7月。计算语言学协会。doi：10.18653/v1/P19-1285。M. Deitke、E. VanderBilt、A. Herrasti、L. Weihs、J. Salvador、K. Ehsani、W. Han、E. Kolve、A. Farhadi、A. Kembhavi 和 R. Mottaghi。ProcTHOR：利用程序化生成实现大规模具身人工智能。载于神经信息处理系统进展，2022年。doi：10.48550/ARXIV.2206.06994。M. Dennis、N. Jaques、E. Vinitsky、A. Bayen、S. Russell、A. Critch 和 S. Levine。通过无监督环境设计实现涌现复杂性与零样本迁移。载于神经信息处理系统进展，第33卷，2020年。J. Devlin、M. Chang、K. Lee 和 K. Toutanova。BERT：用于语言理解的深度双向Transformer预训练。载于 J. Burstein、C. Doran 和 T. Solorio 主编的*2019*年计算语言学协会北美分会会议论文集：人类语言技术，*NAACL-HLT 2019*，美国明尼苏达州明尼阿波利斯，*2019*年*6*月*2*日至*7*日，第*1*卷（长篇与短篇论文）。计算语言学协会，2019年。

Y. Duan、J. Schulman、X. Chen、P. L. Bartlett、I. Sutskever 和 P. Abbeel。RL^2：通过慢速强化学习实现快速强化学习，2017年。L. Fan、G. Wang、Y. Jiang、A. Mandlekar、Y. Yang、H. Zhu、A. Tang、D.-A. Huang、Y. Zhu 和 A. Anandkumar。MineDojo：利用互联网规模的知识构建开放式具身智能体。载于第三十六届神经信息处理系统大会数据集与基准赛道，2022年。K. Fang、Y. Zhu、S. Savarese 和 F.-F. Li。针对困难探索问题的自适应程序化任务生成。载于国际学习表征会议，2021年。G. Farquhar、K. Baumli、Z. Marinho、A. Filos、M. Hessel、H. P. van Hasselt 和 D. Silver。自洽模型与价值。载于 M. Ranzato、A. Beygelzimer、Y. Dauphin、P. Liang 和 J. W. Vaughan 主编的神经信息处理系统进展，第34卷，第1111–1125页，2021年。A. Filos、E. Vértes、Z. Marinho、G. Farquhar、D. Borsa、A. L. Friesen、F. M. P. Behbahani、T. Schaul、A. Barreto 和 S. Osindero。模型-价值不一致性作为认知不确定性的信号。载于 K. Chaudhuri、S. Jegelka、L. Song、C. Szepesvári、G. Niu 和 S. Sabato 主编的国际机器学习会议（*ICML 2022*），*2022*年*7*月*17*日至*23*日，美国马里兰州巴尔的摩，机器学习研究论文集第162卷，第6474–6498页。PMLR，2022年。C. Finn、P. Abbeel 和 S. Levine。用于深度网络快速适应的模型无关元学习。载于第*34*届国际机器学习会议（*ICML*），澳大利亚新南威尔士州悉尼，*8*月*6*日至*11*日，机器学习研究论文集第70卷。PMLR，2017年。C. Finn、K. Xu 和 S. Levine。概率模型无关元学习。神经信息处理系统进展，第31卷，2018年。S. Flennerhag、Y. Schroecker、T. Zahavy、H. van Hasselt、D. Silver 和 S. Singh。自举元学习。载于国际学习表征会议，2022年。M. Fontaine、Y.-C. Hsu、Y. Zhang、B. Tjanaka 和 S. Nikolaidis。环境在人机协调中的重要性。2021年7月。F. Garcia 和 P. S. Thomas。用于终身强化学习探索的元马尔可夫决策过程方法。神经信息处理系统进展，第32卷，2019年。D. Grbic、R. Palm、E. Najarro、C. Glanois 和 S. Risi。*EvoCraft*：开放式演化的新挑战，第325–340页。2021年4月。S. Gronauer 和 K. Diepold。多智能体深度强化学习：综述。《人工智能评论》，55(2)：895–943，2022年。D. Hafner。智能体能力谱系的基准测试。载于国际学习表征会议，2022年。K. He、X. Zhang、S. Ren 和 J. Sun。用于图像识别的深度残差学习。载于*2016*年*IEEE*计算机视觉与模式识别会议（*CVPR*），第770–778页，2016年。doi：10.1109/ CVPR.2016.90。J. Heinrich、M. Lanctot 和 D. Silver。扩展式游戏中的虚拟自博弈。载于国际机器学习会议，第805–813页。PMLR，2015年。D. Hendrycks 和 K. Gimpel。高斯误差线性单元（GELU）。*arXiv*：学习，2016年。

M. Hessel, I. Danihelka, F. Viola, A. Guez, S. Schmitt, L. Sifre, T. Weber, D. Silver 和 H. Van Hasselt。Muesli：结合策略优化方面的改进。载于国际机器学习会议。PMLR，2021年。J. Hoffmann, S. Borgeaud, A. Mensch, E. Buchatskaya, T. Cai, E. Rutherford, D. d. L. Casas, L. A. Hendricks, J. Welbl, A. Clark 等。训练计算资源最优的大语言模型。*arXiv* 预印本 *arXiv:2203.15556*，2022年。H. Hu, A. Peysakhovich, A. Lerer 和 J. Foerster。“other-play”用于零样本协作。载于机器学习与系统会议论文集 *2020*，第 9396–9407 页，2020年。J. Humplik, A. Galashov, L. Hasenclever, P. A. Ortega, Y. W. Teh 和 N. Heess。元强化学习即任务推断。*arXiv* 预印本 *arXiv:1905.06424*，2019年。M. Jaderberg, V. Dalibard, S. Osindero, W. M. Czarnecki, J. Donahue, A. Razavi, O. Vinyals, T. Green, I. Dunning, K. Simonyan 等。神经网络的基于种群的训练。*arXiv* 预印本 *arXiv:1711.09846*，2017年。M. Jaderberg, W. M. Czarnecki, I. Dunning, L. Marris, G. Lever, A. G. Castañeda, C. Beattie, N. C. Rabinowitz, A. S. Morcos, A. Ruderman, N. Sonnerat, T. Green, L. Deason, J. Z. Leibo, D. Silver, D. Hassabis, K. Kavukcuoglu 和 T. Graepel。基于群体强化学习的 3D 多人游戏人类水平性能。《科学》，364(6443):859–865，2019年。M. Janner, Q. Li 和 S. Levine。将离线强化学习视为一个大型序列建模问题。载于神经信息处理系统进展，2021年。M. Jiang, M. Dennis, J. Parker-Holder, J. Foerster, E. Grefenstette 和 T. Rocktäschel。回放引导的对抗性环境设计。载于神经信息处理系统进展，2021a。M. Jiang, E. Grefenstette 和 T. Rocktäschel。优先级关卡回放。载于国际机器学习会议，2021b。M. Johnson, K. Hofmann, T. Hutton 和 D. Bignell。用于人工智能实验的 Malmo 平台。载于第二十五届国际人工智能联合会议论文集。AAAI Press，2016年。A. Juliani, A. Khalifa, V. Berges, J. Harper, E. Teng, H. Henry, A. Crespi, J. Togelius 和 D. Lange。Obstacle Tower：视觉、控制与规划领域的泛化挑战。载于*IJCAI*，2019年。N. Justesen, R. R. Torrado, P. Bontrager, A. Khalifa, J. Togelius 和 S. Risi。程序化关卡生成提升了深度强化学习的泛化能力。*CoRR*，abs/1806.10729，2018年。A. Kanervisto, S. Milani, K. Ramanauskas, N. Topin, Z. Lin, J. Li, J. Shi, D. Ye, Q. Fu, W. Yang, W. Hong, Z. Huang, H. Chen, G. Zeng, Y. Lin, V. Micheli, E. Alonso, F. Fleuret, A. Nikulin, Y. Belousov, O. Svidchenko 和 A. Shpilman。Minerl 钻石 2021 竞赛：概述、结果与经验教训，2022年。J. Kaplan, S. McCandlish, T. Henighan, T. B. Brown, B. Chess, R. Child, S. Gray, A. Radford, J. Wu 和 D. Amodei。神经语言模型的缩放定律。*CoRR*，abs/2001.08361，2020年。R. Kirk, A. Zhang, E. Grefenstette 和 T. Rocktäschel。深度强化学习泛化研究综述。*CoRR*，abs/2111.09794，2021年。L. Kirsch, J. Harrison, J. Sohl-Dickstein 和 L. Metz。通过元学习 Transformer 实现通用上下文学习。*arXiv*，2022年。

A. Kumar, Z. Fu, D. Pathak 和 J. Malik。RMA：足式机器人的快速运动适应。载于《机器人学：科学与系统》，2021年。H. Küttler, N. Nardelli, A. H. Miller, R. Raileanu, M. Selvatici, E. Grefenstette 和 T. Rocktäschel。NetHack 学习环境。载于神经信息处理系统会议论文集（*NeurIPS*），2020年。M. Laskin, L. Wang, J. Oh, E. Parisotto, S. Spencer, R. Steigerwald, D. Strouse, S. Hansen, A. Filos, E. Brooks, M. Gazeau, H. Sahni, S. Singh 和 V. Mnih。通过算法蒸馏实现上下文强化学习，2022年。K.-H. Lee, O. Nachum, S. Yang, L. Lee, C. D. Freeman, S. Guadarrama, I. Fischer, W. Xu, E. Jang, H. Michalewski 和 I. Mordatch。多游戏决策变换器。载于 A. H. Oh, A. Agarwal, D. Belgrave 和 K. Cho 编，神经信息处理系统进展，2022年。S. Liu, G. Lever, N. Heess, J. Merel, S. Tunyasuvunakool 和 T. Graepel。通过竞争实现涌现协调。载于国际学习表征会议，2019年。D. Mahajan, R. Girshick, V. Ramanathan, K. He, M. Paluri, Y. Li, A. Bharambe 和 L. van der Maaten。探索弱监督预训练的局限。载于《计算机视觉——*ECCV 2018*：第十五届欧洲会议，德国慕尼黑，*2018*年*9*月*8*日至*14*日，论文集，第二部分》，第185–201 页，柏林，海德堡，2018年。Springer-Verlag。ISBN 978-3-030-01215-1。T. Matiisen, A. Oliver, T. Cohen 和 J. Schulman。师生课程学习。*IEEE* 神经网络与学习系统汇刊，31(9):3732–3740，2020年。L. C. Melo。Transformer 是元强化学习器。载于国际机器学习会议，第 15340–15359 页。PMLR，2022年。L. Metz, C. D. Freeman, S. S. Schoenholz 和 T. Kachman。梯度并非你所需的一切。*arXiv* 预印本 *arXiv:2111.05803*，2021年。V. Mikulik, G. Delétang, T. McGrath, T. Genewein, M. Martic, S. Legg 和 P. Ortega。元训练智能体实现贝叶斯最优智能体。载于 H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan 和 H. Lin 编，神经信息处理系统进展，第 33 卷，第 18691–18703 页。Curran Associates, Inc.，2020年。J. Mu, V. Zhong, R. Raileanu, M. Jiang, N. Goodman, T. Rocktäschel 和 E. Grefenstette。利用语言抽象改进内在探索。载于神经信息处理系统进展，2022年。R. Munos, T. Stepleton, A. Harutyunyan 和 M. Bellemare。安全高效的离策略强化学习。神经信息处理系统进展，29，2016年。E. Nikishin, M. Schwarzer, P. D'Oro, P.-L. Bacon 和 A. Courville。深度强化学习中的首因偏差。载于国际机器学习会议，第 16828–16847 页。PMLR，2022年。OEL Team, A. Stooke, A. Mahajan, C. Barros, C. Deck, J. Bauer, J. Sygnowski, M. Trebacz, M. Jaderberg, M. Mathieu, N. McAleese, N. Bradley-Schmieg, N. Wong, N. Porcel, R. Raileanu, S. Hughes-Fitt, V. Dalibard 和 W. M. Czarnecki。开放式学习孕育通用能力智能体。*CoRR*，abs/2107.12808，2021年。

OpenAI、M. Plappert、R. Sampedro、T. Xu、I. Akkaya、V. Kosaraju、P. Welinder、R. D' Sa、A. Petron、H. P. de Oliveira Pinto、A. Paino、H. Noh、L. Weng、Q. Yuan、C. Chu 和 W. Zaremba。用于机器人操作中自动发现目标的非对称自我博弈，2021年。P. A. Ortega、J. X. Wang、M. Rowland、T. Genewein、Z. Kurth-Nelson、R. Pascanu、N. Heess、J. Ve- ness、A. Pritzel、P. Sprechmann 等。序列策略的元学习。*arXiv* 预印本 *arXiv:1905.03030*，2019年。K. Ota、D. K. Jha 和 A. Kanezaki。训练更大的深度强化学习网络，2021年。E. Parisotto。通过记忆实现的元强化学习。博士论文，卡内基梅隆大学，宾夕法尼亚州匹兹堡，2021年。E. Parisotto、F. Song、J. Rae、R. Pascanu、C. Gulcehre、S. Jayakumar、M. Jaderberg、R. L. Kaufman、A. Clark、S. Noury 等。稳定用于强化学习的 Transformer。载于国际机器学习会议，第7487–7498页。PMLR，2020年。J. Parker-Holder、M. Jiang、M. Dennis、M. Samvelyan、J. Foerster、E. Grefenstette 和 T. Rocktäschel。通过基于遗憾的环境设计演进课程。载于国际机器学习大会，2022年。M. Pislar、D. Szepesvari、G. Ostrovski、D. L. Borsa 和 T. Schaul。智能体何时应该探索？载于国际学习表征会议，2022年。R. Portelas、C. Colas、K. Hofmann 和 P. Oudeyer。面向连续参数化环境中深度强化学习课程学习的教师算法。载于 L. P. Kaelbling、D. Kragic 和 K. Sugiura 编，《style id='11'>第三届年度机器人学习大会（CoRL 2019）论文集，日本大阪，2019年10月30日至11月1日，机器学习研究论文集第100卷，第835–853页。PMLR，2019年。J. W. Rae、S. Borgeaud、T. Cai、K. Millican、J. Hoffmann、F. Song、J. Aslanides、S. Henderson、R. Ring、S. Young 等。扩展语言模型：Gopher 训练的方法、分析与见解。*arXiv* 预印本 *arXiv:2112.11446*，2021年。R. Raileanu 和 T. Rocktäschel。Ride：为程序生成环境中的影响驱动探索提供奖励。载于国际学习表征会议，2020年。K. Rakelly、A. Zhou、C. Finn、S. Levine 和 D. Quillen。通过概率上下文变量实现高效的离策略元强化学习。载于国际机器学习会议，第5331–5340页。PMLR，2019年。S. Reed、K. Zolna、E. Parisotto、S. G. Colmenarejo、A. Novikov、G. Barth-maron、M. Giménez、Y. Sulsky、J. Kay、J. T. Springenberg、T. Eccles、J. Bruce、A. Razavi、A. Edwards、N. Heess、Y. Chen、R. Hadsell、O. Vinyals、M. Bordbar 和 N. de Freitas。通才智能体。机器学习研究汇刊，2022年。M. Reid、Y. Yamada 和 S. S. Gu。维基百科能帮助离线强化学习吗？*CoRR*，2022年。S. Risi 和 J. Togelius。通过程序化内容生成提高机器学习的通用性。《自然·机器智能》，2，2020年8月。doi：10.1038/s42256-020-0208-z。M. Samvelyan、R. Kirk、V. Kurin、J. Parker-Holder、M. Jiang、E. Hambro、F. Petroni、H. Kuttler、E. Grefenstette 和 T. Rocktäschel。MiniHack the Planet：开放式强化学习研究的沙盒。载于第三十五届神经信息处理系统大会数据集与基准赛道，2021年。

M. Samvelyan、A. Khan、M. D. Dennis、M. Jiang、J. Parker-Holder、J. N. Foerster、R. Raileanu 和 T. Rocktäschel。MAESTRO：面向多智能体强化学习的开放式环境设计。载于 *NeurIPS 2022* 深度强化学习研讨会，2022年。T. Schaul、J. Quan、I. Antonoglou 和 D. Silver。优先经验回放。载于国际学习表征会议，2015年。J. Schmidhuber。好奇心驱动的模型构建控制系统。载于 *[论文集] 1991 IEEE* 国际神经网络联合会议，第1458–1463页，第2卷，1991 年。doi：10.1109/IJCNN.1991.170605。J. Schmidhuber。运用历史压缩原理学习复杂的长序列。《神经计算》，4(2):234–242，1992年。doi：10.1162/neco.1992.4.2.234。S. Schmitt、J. J. Hudson、A. Zidek、S. Osindero、C. Doersch、W. M. Czarnecki、J. Z. Leibo、H. Kuttler、A. Zisserman、K. Simonyan 等。启动训练深度强化学习。*arXiv* 预印本 *arXiv:1803.03835*，2018年。C. Schuhmann、R. Beaumont、R. Vencu、C. W. Gordon、R. Wightman、M. Cherti、T. Coombes、A. Katta、C. Mullis、M. Wortsman、P. Schramowski、S. R. Kundurthy、K. Crowson、L. Schmidt、R. Kaczmarczyk 和 J. Jitsev。LAION-5b：用于训练新一代图文模型的开放式大规模数据集。载于第三十六届神经信息处理系统大会数据集与基准赛道，2022年。N. Shazeer。GLU 变体改进 Transformer。*arXiv* 预印本 *arXiv:2002.05202*，2020年。B. C. Stadie、G. Yang、R. Houthooft、X. Chen、Y. Duan、Y. Wu、P. Abbeel 和 I. Sutskever。关于通过元强化学习学习探索的一些思考。*arXiv* 预印本 *arXiv:1803.01118*，2018年。P. Stone、G. A. Kaminka、S. Kraus 和 J. S. Rosenschein。临时自主智能体团队：无需预先协调的协作。载于 M. Fox 和 D. Poole 编，《style id='21'>第二十四届 AAAI 人工智能大会论文集（AAAI 2010），美国佐治亚州亚特兰大，2010年7月11日至15日。AAAI Press，2010年。D. Strouse、K. McKee、M. Botvinick、E. Hughes 和 R. Everett。无需人类数据与人类协作。神经信息处理系统进展，34:14502–14515，2021年。S. Sukhbaatar、Z. Lin、I. Kostrikov、G. Synnaeve、A. Szlam 和 R. Fergus。通过非对称自我博弈实现内在动机与自动课程。载于国际学习表征会议，2018年。C. Sun、A. Shrivastava、S. Singh 和 A. Gupta。重新审视深度学习时代数据的惊人效能。载于*2017 IEEE* 国际计算机视觉会议（*ICCV*），第843–852页，2017年。Y. Tay、M. Dehghani、S. Abnar、H. W. Chung、W. Fedus、J. Rao、S. Narang、V. Q. Tran、D. Yogatama 和 D. Metzler。缩放定律与模型架构：归纳偏置如何影响缩放？，2022年。J. Togelius 和 J. Schmidhuber。自动游戏设计实验。载于*2008 IEEE* 计算智能与游戏研讨会，第111–118页，2008年。doi：10.1109/CIG.2008.5035629。A. Vaswani、N. Shazeer、N. Parmar、J. Uszkoreit、L. Jones、A. N. Gomez、Ł. Kaiser 和 I. Polosukhin。注意力就是你所需要的一切。神经信息处理系统进展，30，2017年。

O. Vinyals, I. Babuschkin, W. M. Czarnecki, M. Mathieu, A. Dudzik, J. Chung, D. H. Choi, R. Powell, T. Ewalds, P. Georgiev, J. Oh, . D. Horgan, M. Kroiss, I. Danihelka, A. Huang, L. Sifre, T. Cai, J. P. Agapiou, M. Jaderberg, A. S. Vezhnevets, R. Leblond, T. Pohlen, V. Dalibard, D. Budden, Y. Sulsky, J. Molloy, T. L. Paine, Ç. Gülçehre, Z. Wang, T. Pfaff, Y. Wu, R. Ring, D. Yogatama, D. Wünsch, K. McKinney, O. Smith, T. Schaul, T. P. Lillicrap, K. Kavukcuoglu, D. Hassabis, C. Apps, and D. Silver. 使用多智能体强化学习达到《星际争霸II》宗师级水平。《自然》, 575(7782): 350–354, 2019. doi: 10.1038/s41586-019-1724-z. R. Vuorio, J. A. Beck, G. Farquhar, J. N. Foerster, and S. Whiteson. 不掷骰子：对元梯度中偏差-方差权衡的研究。发表于*NeurIPS 2021* 深度强化学习研讨会，2021年。L. Vygotsky. 学习与发展之间的交互。《儿童发展读本》，第34–40页，1978年。J. X. Wang, Z. Kurth-Nelson, D. Tirumala, H. Soyer, J. Z. Leibo, R. Munos, C. Blundell, D. Kumaran, and M. Botvinick. 学习如何进行强化学习。*arXiv*预印本 *arXiv:1611.05763*，2016年。J. X. Wang, Z. Kurth-Nelson, D. Kumaran, D. Tirumala, H. Soyer, J. Z. Leibo, D. Hassabis, and M. Botvinick. 前额叶皮层作为元强化学习系统。《自然-神经科学》，21 (6):860–868，2018年。J. X. Wang, M. King, N. Porcel, Z. Kurth-Nelson, T. Zhu, C. Deck, P. Choy, M. Cassin, M. Reynolds, F. Song, et al. Alchemy：一种用于元强化学习的结构化任务分布。*arXiv*，2021年。R. Wang, J. Lehman, J. Clune, and K. O. Stanley. 成对开放式开拓者（POET）：持续生成日益复杂且多样的学习环境及其解决方案。*CoRR*，abs/1901.01753，2019年。R. Wang, J. Lehman, A. Rawal, J. Zhi, Y. Li, J. Clune, and K. Stanley. 增强版 POET：通过不受限制地发明学习挑战及其解决方案，实现开放式强化学习。发表于 H. D. III and A. Singh（编），第*37*届国际机器学习会议论文集，第119卷，机器学习研究论文集，第9940–9951页。PMLR，2020年7月13日至18日。T. Wang, H. Dong, V. Lesser, and C. Zhang. ROMA：具有涌现角色的多智能体强化学习。*arXiv*预印本 *arXiv:2003.08039*，2020年。Z. Xu, J. Modayil, H. P. van Hasselt, A. Barreto, D. Silver, and T. Schaul. 自然价值近似器：学习何时信任过去的估计值。发表于 I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett（编），神经信息处理系统进展，第30卷。Curran Associates, Inc.，2017年。Z. Xu, H. P. van Hasselt, and D. Silver. 元梯度强化学习。神经信息处理系统进展，31，2018年。J. Yang, B. Petersen, H. Zha, and D. Faissol. 强化学习中的单回合策略迁移。发表于国际学习表征会议，2019年。J. Yang, A. Li, M. Farajtabar, P. Sunehag, E. Hughes, and H. Zha. 学习如何激励其他学习智能体。发表于神经信息处理系统进展。arXiv, 2020年。doi: 10.48550/ARXIV. 2006.06051. W. Yu, J. Tan, Y. Bai, E. Coumans, and S. Ha. 通过元策略优化学习快速适应。《*IEEE*机器人与自动化快报》，5(2):2950–2957，2020年。

X. Zhai, A. Kolesnikov, N. Houlsby, and L. Beyer. 扩展视觉变换器。发表于*2022 IEEE/CVF*计算机视觉与模式识别会议（*CVPR*），美国加利福尼亚州洛斯阿拉米托斯，2022年6月。IEEE计算机学会。Z. Zheng, J. Oh, M. Hessel, Z. Xu, M. Kroiss, H. Van Hasselt, D. Silver, and S. Singh. 学得的内在奖励能够捕捉什么？发表于国际机器学习会议，第11436–11446页。PMLR，2020年。V. Zhong, T. Rocktäschel, and E. Grefenstette. RTFM：通过阅读泛化到新的环境动态。发表于第八届国际学习表征会议，*ICLR 2020*，埃塞俄比亚亚的斯亚贝巴，*2020*年*4*月*26*日至*30*日，2020年。L. Zintgraf. 通过元强化学习实现快速适应。博士论文，牛津大学，2022年。L. Zintgraf, K. Shiarlis, M. Igl, S. Schulze, Y. Gal, K. Hofmann, and S. Whiteson. Varibad：一种通过元学习实现贝叶斯自适应深度强化学习的优良方法。*arXiv*预印本 *arXiv:1910.08348*，2019年。

附录

A. 环境详情

A.1. XLand 2.0

本节介绍 XLand 2.0 与OEL Team 等人（2021年）的原始 XLand 环境之间的差异。我们对配置空间作如下修改：

- 我们引入了一种新关系 `touching(a,b)`。当物体 `a` 和 `b` 接触时，该关系成立；接触情况由 Unity 的碰撞检测判定，距离阈值为 1 毫米。
- 我们排除了所有涉及地面的关系。作为一种更灵活的替代方案，我们引入了将物体以永久冻结状态生成的选项，使其无法移动。冻结物体可在环境中用作锚点，玩家必须通过将其纳入目标或生成规则来导航至这些锚点。
- 我们仅使用由单个关系或其否定构成的谓词，不采用多个关系的合取或析取。请注意，生成规则（第 2.1 节）允许我们指定要求玩家按顺序满足谓词的任务，或为玩家提供多种方式来达到期望状态。

为方便读者，表 A.1 和 A.2 分别列出了我们在 XLand 中用于物体的所有形状和颜色。表A.3 列出了我们用于目标和生成规则的所有谓词。

表 A.1 | 物体使用的形状。

形状名称
Wall
Cube
球体
金字塔

表 A.2 | 物体使用的颜色。

颜色名称
黑色
紫色
黄色

表 A.3 | 用于目标和生成规则的谓词。

谓词名称	**含义**
`touching(a,b)`	`a` 和 `b` 是否接触。
`near(a,b)`	`a` 和 `b` 之间的距离是否不超过 1 米。
`hold(a,b)`	玩家 `a` 是否持有 `b`。
`see(a,b)`	如果 `a` 是玩家，它是否能看见 `b`。如果不是，则判断这条线连接 `a` 和 `b` 的质心的路径未被另一个物体阻挡。
`not(p)`	谓词 `p` 是否不成立。

A.2. 训练用任务的预采样

目标和产生式规则的空间至少可以生成 10^{40} 种不同的任务，即使对物体和规则的数量施加了相当严格的限制。[6] 为方便起见，我们使用下文介绍的方法对该空间的一个子集进行预采样。在第 3.5 节中，我们评估采样集大小的影响。对于每个任务，我们使用 OEL Team 等人(2021年) 概述的流程采样一个世界，并将其与一个游戏和生成规则组合起来，如下所示。

单人任务。我们首先均匀采样玩家的目标，该目标由一个可选是否带有否定的谓词和两个物体构成。然后，对于固定数量的步骤（我们在 1 到 4 之间均匀采样），添加新的生成规则，使其必须按顺序触发，才能获得目标中指定的物体。我们初始化世界，使其包含第一条生成规则的条件中指定的物体，以及最多 10 个干扰物体，这些物体不会出现在该规则链中的任何生成规则（也不会出现在目标）中。

接下来，我们引入死胡同生成规则。我们对这些规则进行采样，使其条件包含干扰物体，或最终满足目标所必需的物体，但生成物始终是干扰物。因此，触发死胡同可能会使游戏进入无解状态。在训练游戏中加入这些规则，会促使智能体避免不加区分地触发所有生成规则。最后，我们采样一个隐藏掩码，用于指定玩家能够看到或被隐藏的生成规则部分。隐藏机制（见第2.1节所述）以及实际要隐藏部分的采样均采用均匀随机方式。

多人任务。 在本研究中，我们将多人游戏限制为完全合作的双人游戏。众所周知，这类游戏尤其具有挑战性，因为它们存在多个均衡，因此会带来均衡选择问题（Dafoe 等人，2020年）。为采样此类游戏，我们首先按照上述方法采样一个单人游戏，然后随机将目标或生成规则中对玩家一的引用替换为对玩家二的引用。我们复制目标，并为玩家二采样一个新的生成规则隐藏掩码，由此得到一个完全合作的双人游戏，其中提供的关于任务生成规则的信息可能并非对称。

B. 评估

B.1. 测试分数

我们使用从与训练游戏相同的分布中采样、留出的 1000 个测试任务，并采用留出的世界拓扑结构，对智能体进行评估。预先生成 XLand 任务池的流程详见附录 A.2。拒绝采样确保测试集中不包含训练集中的任何游戏（目标和生成规则）。

在 XLand 中，只要目标达成，每一帧都会获得奖励，因此不同任务的总奖励无法直接比较。为解决这一问题，我们在测试任务集上对 AdA 进行微调。我们计算微调智能体在最后一次试验中获得的最高总奖励（试次数最多为 13 次），并将其作为每项测试任务在单次试验中可能获得的最高奖励的估计值。我们将这一数值称为归一化器。我们将智能体 i 在任务 m 上进行 k 次试验时的测试得分 S_m^i 定义为第 k 次试验获得的总奖励除以归一化器。这样一来，奖励大致被归一化到

[6]这是一个数量级下界估计，假设有 4 种形状、3 种颜色、7 个谓词、最多 5 条产生式规则（每条规则的右侧最多包含 3 个物体）、7 个留空选项，并且场景中最多有 20 个物体。

区间 $[0, 1]$。请注意，受评估的智能体可能获得高于 1 的得分，这既可能是评估过程中的噪声所致，也可能是因为受评估的智能体优于用于创建归一化器的智能体。

报告智能体在包含 k 次试次的游戏中的得分时，我们始终使用最后一次试次的总奖励。给定试次数量 k，这是一项衡量智能体是否在新颖的马尔可夫决策过程中成功权衡探索与利用的良好指标。如果智能体具备适应能力，我们预期最后一次（第 k 次）试次中的性能会随着 k 增加而提升：也就是说，智能体能够即时利用额外经验来取得更好的表现。我们在 $k \in \{1, 2, 3, 5, 8, 13\}$ 上进行评估，其中 8 和 13 是训练期间未见过的 k 取值。

为了聚合智能体在各个游戏中的得分，我们采用固定的（通常为 20th）百分位数得分（Agarwal 等人，2021年）。这为智能体在大多数任务上的性能提供了下界保证：例如，如果某智能体的 20th 百分位数得分为 0.5，那么它在 80% 的游戏中至少能获得 0.5 分。采用这种聚合方法（而非平均值），可以让我们专注于覆盖众多任务，而不是把精力放在提升离群任务上的性能。根据实证结果，我们发现，在一系列不同的百分位数下，结果都很稳健。

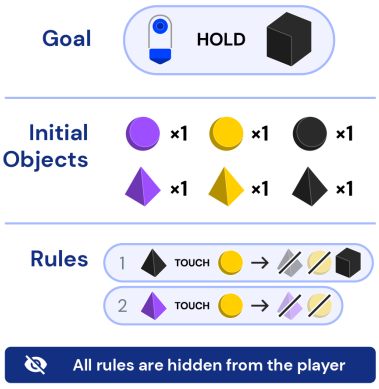
B.2. 人工编写的探测任务

使用测试任务进行评估，只能让我们了解附录 A.2 中预先采样的分布。尽管该分布确实范围广泛且多样，但从测试集随机抽取的任务并不一定容易为人类理解。因此，除了对 1000 个测试任务进行定量评估外，我们还通过两组探测任务来考察智能体具备的特定人类水平能力：其中包括 30 个单智能体探测任务和 28 个多智能体探测任务。这些任务基于人类容易理解的情境，要求智能体展现出在性质上截然不同的行为，因此我们可以“直接观察”并做更细致的检视。全部 58 个探测任务的完整说明见附录 F。图 B.1 至 B.3 对具有代表性的单智能体和多智能体探测任务进行了详细说明。

B.3. 适应性指标

我们提出一种适应性指标，用于根据智能体在人工编写的探测任务集上的少样本适应能力对其进行排名。对于所有 (m, k) 配对，我们收集最后一次试验的总奖励，其中m 是探测任务，$k \in$ 取值为$\{1,2,3,5,8,13\}$。我们像处理上述测试集一样对每个任务的得分进行归一化。然后，使用 50th 百分位数（中位数）对各任务得分进行汇总。最后，我们对k 进行汇总：当且仅当智能体 A 的任务汇总得分帕累托优于智能体 B 时，才认为智能体 A 的排名高于智能体 B。也就是说，我们希望智能体既能在可能的情况下实现高质量的零样本泛化（$k = 1$），又能利用额外的试验信息，在少数几次试验中高效改进其策略 $k > 1$；我们不会在这两者之间进行“权衡”。

使用帕累托改进准则来计算标量指标的一种便捷方法，是采用Balduzzi 等人（2018年提出的纳什平均法。我们构建了一个“智能体对战 k”的竞争性元博弈，并计算该游戏的最大熵纳什均衡。随后将纳什收益用作适应性指标，并据此为智能体排名。正如预期，该指标具有这样的性质：如果智能体 A 和智能体 B 均不帕累托支配对方，那么 A 和 B 获得相同的纳什收益，因此排名相同。我们在第3.1节中使用这一适应性指标来选择超参数，以训练出表现最佳的智能体。



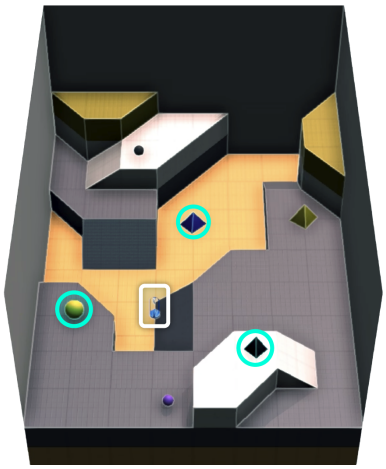
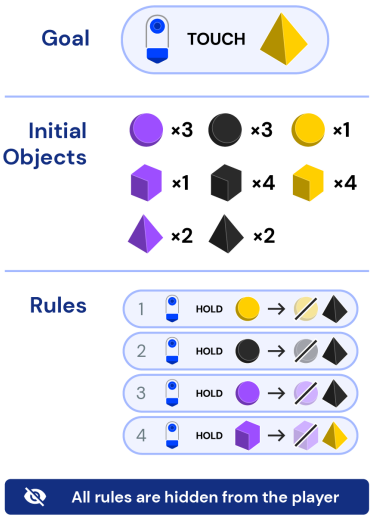


图 B.1 | **错误配对消失**：玩家的目标是拿着一个黑色立方体，而初始物体中并不存在这种物体。但这里有两条（隐藏的）生成规则。玩家需要找出能触发生成立方体规则的正确世界状态，而不是触发摧毁必要输入物体规则的状态。所有这些都发生在一个具有单向落差且视野有限、充满挑战的世界布局中。



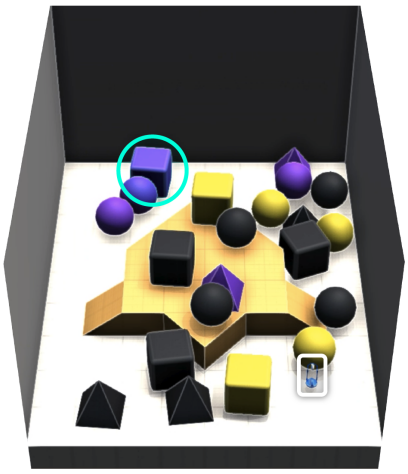


图 B.2 | **大海捞针式金字塔**：要创造所需的黄色金字塔，玩家需要找到并拿着紫色立方体。这个世界中有多个干扰物体和干扰规则，因此玩家不仅要应对艰难的探索挑战，还要面对非常嘈杂的环境。



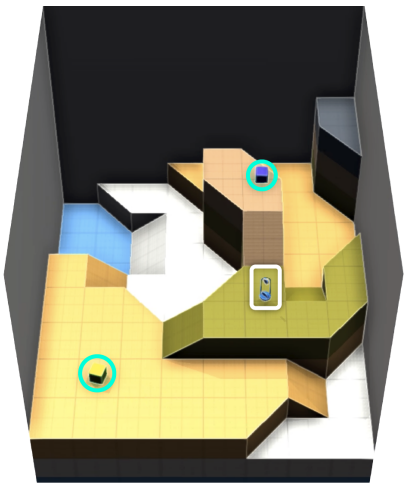


图 B.3 | **推而不举**：绝大多数训练和评估任务都要求举起物体。这里有两条隐藏规则，会在任何物体被举起时将其摧毁。要创造目标状态，需要一些“横向思维”：玩家需要发现，可以用身体推动立方体。



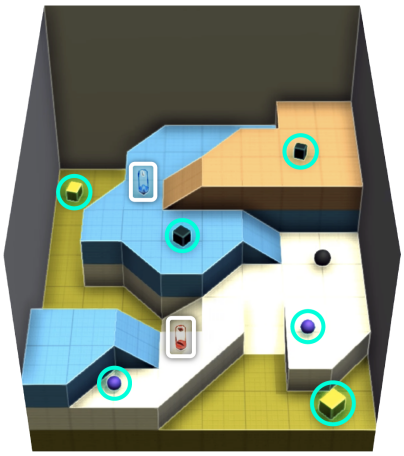
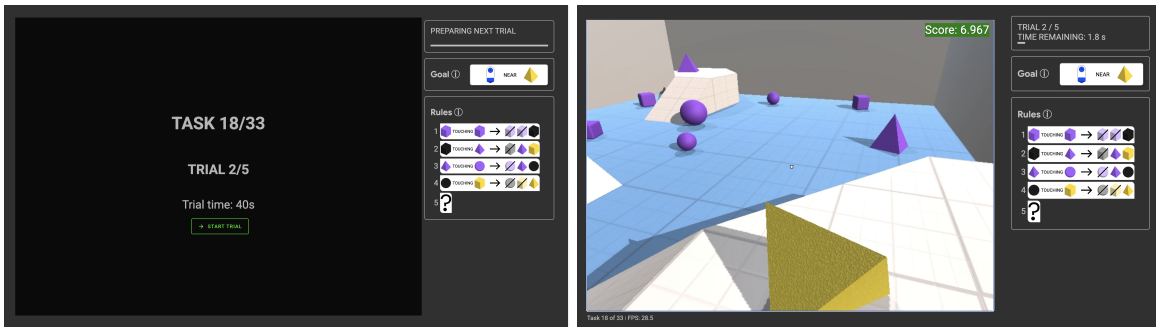


图 B.4 | **两方的不可逆生成**：第一位玩家持有一个黄色球体时，双方均可得分。初始状态中没有黄色球体，但按顺序执行第一、第二和第四条生成规则即可生成。另两条规则会走入死路，摧毁关键输入对象。请注意，某些输入对象在初始状态中存在多个，因此有多条解路径。



(a) (b)

图 B.5 | 人类玩家界面。（**a**）每次试次开始前，玩家会获知当前任务还剩多少次试次，并有无限时间阅读目标和生成规则（受任何隐藏机制的限制）。（**b**）试次进行期间，玩家观察到的第一人称摄像机视角与智能体相同，但 800×600 像素分辨率更高。目标、生成规则、当前得分以及试次剩余时间会通过用户界面元素显示。

B.4. 人类数据收集

为人类时间尺度适应提供基准，我们从一个由100 名人类玩家组成的池中收集了30 个单智能体探测任务的得分数据。在尝试探测任务之前，每名玩家完成了一个包含23 个任务的分级训练课程，以熟悉鼠标和键盘控制方案及用户界面（图B.5），以及 XLand 的特定游戏机制。由于人类无法在回合边界进行短期记忆“重置”，每名玩家仅针对每个探测任务进行一次 k，其中 $k \in \{1、2、3、5、8\}$。所有玩家在整个任务集中都体验了多种 k。我们为每名玩家分配了 30 个探测任务的一种独特顺序，以平均消除从较早任务向较晚任务进行的任何知识迁移；同时，在这些顺序中，我们对已知相似的任务之间设置了间隔。技术问题（例如网络中断）导致 3000 个回合中有 3.4% 未能完成，因此每个（任务，k）对平均留下 19.3 个样本，任何单个（任务，k）至少留下 17 个样本。

C. 智能体详情

在此，我们提供智能体架构、观测和动作规范的技术细节。

C.1. 智能体架构

此处概述智能体架构，并介绍智能体的重要超参数。需要指出的是，本研究仅进行了最低限度的超参数调优。

观测编码器。 第一人称视角的 RGB 观测（表 C.1）输入残差网络（He 等人，2016年），包含 $[16, 32, 32]$ 个卷积通道，每个通道由 2×2 个模块组成，最终输出大小为 256。除使用标量残差乘数外，还采用最大池化。 `relu` 全程使用激活值。

目标观测经过目标嵌入器处理，该嵌入器与 OEL Team 等人（2021年）中的相同。它将目标表示中的 6 个目标元素（否定、谓词、物体 1 的形状、物体 1 的颜色、物体 2 的形状、物体 2 的颜色）分别映射为大小为 8 的稠密嵌入向量。随后将这些向量拼接起来，并输入大小为 $[8, 8, 8]$，的 3 层多层感知机，最终得到大小为8的目标嵌入。

生成规则采用与目标相同的方式进行编码，但通过一个形状为 $[512, 256]$ 的更大的最终多层感知机进行映射，最终得到大小为 256 的产生式规则嵌入向量。

编码后的 RGB、目标和生成规则观测，与所有其余标量及向量观测（包括先前奖励、先前动作、本体感觉观测和试验计时信息）拼接在一起，并输入最终的多层感知机。由此得到的编码观测向量大小与 Transformer 记忆的隐藏维度相同。

Transformer 记忆。我们使用带因果掩码的 Transformer-XL（Dai 等人，2019年）。行动器步使用大小为 1 的上下文窗口，学习器步则使用大小为 80 的展开上下文窗口。除非另有说明，所有实验中 Transformer-XL 记忆均使用 300 个先前缓存激活（相当于 1800 个有效时间步）。与 Parisotto 等人（2020年）一样，我们在注意力操作之前应用层归一化；如Shazeer （2020年）所述，在前馈组件中使用门控；并如Dai 等人（2019年）所述应用相对位置嵌入。与 Transformer 中的常见做法一样，我们始终使用 `gelu` 激活函数（Hendrycks 和 Gimpel，2016年）。

Muesli序列模型和预测头。接下来，我们取Transformer-XL的输出嵌入，并使用两个宽度为 1000 的多层感知机，为Muesli模型长短期记忆网络的初始状态生成隐状态值和细胞值。在隐状态值之上，我们分别应用宽度为1000的策略和价值多层感知机头。策略多层感知机后接6个线性Softmax输出，对应于策略分解式动作空间中的6个动作组（如OEL Team等人(2021年所述）。价值多层感知机后接一个601单元的分箱logit预测头，如Hessel等人(2021年所述。最后，从LSTM状态嵌入开始，将Muesli序列模型继续展开4个步骤，并在每个LSTM步骤生成一个1000维输出向量。该向量输入另一个1000维多层感知机，后接一个601单元的分箱奖励预测头。

C.2. 观测

我们在表C.1中总结了智能体进行推理时接收到的所有观测。在描述中，“兼容旧版的原因”指的是从OEL Team等人(2021年)继承而来的观测格式。

表 C.1 | 智能体观测。

观测名称	形状	含义
RGB	72×96×3	智能体第一人称视角的 RGB 值。
正在持有	1	整数 {0,1}，表示智能体是否正在持有 物体。
手部距离	1	到持有物体的距离，占智能体.....的比例 最大触及范围（未持有物体时为 0）。
手部力量	1	智能体对持有物体施加的力量 占其最大握力的比例（未持有物体时为 0 物体）。
最后动作	10	智能体执行的最后动作。
目标原子	6 × 6	智能体可读的目标描述。出于兼容旧版的 原因，只有第一行非零。第一行 （6 个数字）通过以下元素描述目标： 1：目标是否取反， 2：二元谓词的索引， 3-4：第一个物体的形状和颜色， 5-6：第二个物体的形状和颜色。
目标SOP矩阵	6 × 6	始终相同，为兼容旧版而保留。
原子奖励	6	智能体在上一帧获得的奖励（1 个数字），为兼容旧版而填充了零。
选项奖励	6	与原子奖励相同，为兼容旧版而保留。
生成规则	16 × 26	最多 16 条生成规则的描述。单个 产生式规则描述为： 3 × 6 = 18 个数字 描述最多三个“物体-谓词-物体”触发 器，以及描述最多 4 个生成对象的 4×2 = 8 个数字 。本文所有任务始终只使用一个“物体-谓词- 物体”触发器。隐藏（第 节 2.1）通过添加额外谓词来实现 以及表示隐藏产生式规则含义的形状索引， 第一个隐藏物体，等等。
奖励	1	上一步获得的环境奖励 骤。
剩余试次	5	剩余试次数量的独热编码 回合中剩余试次数量，最多为 5 次。
剩余试验次数超过 5 次	1	一个取值为 {0,1} 的整数，表示是否还剩更多 回合中剩余试次超过 5 次。
距上次的时间 试验	1	距离当前回合中最后一次试验还剩（秒） 当前回合。
当前试验剩余时间 试验	1	距离结束还剩（秒） 当前试验。

上一次 试验时长	1	本回合最后一次试验的时长（以秒为单位） 。在我们的设定中，同一任务的所有试次 时长相同。
下一次 试验时长	1	当前回合中下一次试验的时长（秒） 。在我们的设定中，同一 任务的所有试验时长均相同。

D. 训练细节

D.1. 元强化学习

AdA 采用元强化学习设置进行训练，其中，智能体的经验回合包含与任务环境进行交互的多个试次，并在试次边界重置任务。在这种设置下，已知智能体的策略可以收敛至贝叶斯最优行为，在执行过程中不断尝试以降低认知不确定性，并重用已发现的信息，从而越来越高效地实现目标。重要的是，智能体的记忆不会在试次边界重置，只会在回合边界重置。同样，智能体在整个回合中使用固定的折扣因子进行训练，$\gamma \neq 0$ 包括试次边界处。训练时，我们按照因子化分布$(m,\ k)$ 采样$(\rho_{\mathcal{M}},\ \rho_K)$对，其参数由自动课程控制（第2.3节）。所有马尔可夫决策过程 m 均从一个程序化生成的领域 M（称为 XLand，第 2.1 节）中抽取。我们将试次空间设为 $K = \{1, 2, \ldots..6\}$，并向智能体提供 k 作为条件输入。训练完成后，智能体能够在广泛的马尔可夫决策过程中进行少样本适应，包括在留出的 $m \in \mathrm{M}$ 中（对于这些环境，$\rho_{\mathcal{M}}$ 赋予的概率质量为零），以及在 $k > 6$时。

D.2. 单智能体训练

我们的单智能体训练设置使用了一个按第2.1节所述生成的任务池。单智能体蒸馏教师的实验设置总结见表D.1。单智能体训练使用的 XLand 2.0 版本早于多智能体实验所用的版本，且不包含第A.1节所述的冻结物体。因此，测试集和人工编写的探测任务集中也未包含冻结物体。AdA 使用 JAX（Bradbury 等人，2018年）和 DeepMind JAX 生态系统（Babuschkin 等人，2020年）实现，并在 64 台 Google TPUv3 设备上训练。从头开始训练此版本 AdA 的实际耗时约为 5 周：1 周训练教师，4 周训练 AdA。即使经过这么长时间的训练，AdA 仍未收敛，这说明了开放式学习方法的优势。

表 D。1 | 第3.1节单智能体实验的蒸馏教师。

模型参数 2300万 TXL / 总计 7600 万	记忆	任务池	课程	教师	步数
	1800	25B	PLR D.5	None	25B

D.3. 多智能体训练

从用于单智能体训练的 250亿规模任务池出发，我们按照附录 A.2中描述的流程，生成一个规模相同的双人任务池。对于所有多智能体实验，我们采用单人任务和双人任务各占一半的混合任务池，如第 2.1节所述。对于每项任务，我们以一定概率将部分初始物体永久冻结（见附录A.1）

概率为 50%。对于包含冻结物体的任务，我们遍历最初生成的对象类型，并以 20% 的概率冻结该类型的所有已生成实例，同时确保任务仍然可解。

训练期间，我们从使用虚拟自博弈生成的池中均匀采样一个协作玩家策略。协作玩家池以随机动作策略初始化。每经过 500M 个训练帧，我们就将学习智能体的一个快照加入池中，从而随着时间推移不断增加能力更强的协同玩家。最后，我们应用 PLR 自动课程方法（参见第 2.3 节），根据智能体基于时序差分误差的适应度筛选任务（世界、游戏和协同玩家）。多智能体蒸馏教师的实验设置汇总于表 D.2。

表 D.2 | 第 3.1 节多智能体实验所用的蒸馏教师。

模型参数	记忆	任务池	课程	教师	步数
2300万 TXL / 总计 7600 万	1800	参见第 D.3	PLR D.5	None	22B

D.4. 架构实验

表 D.3 展示了第3.2节中用于比较不同记忆架构的实验设置。

表 D.3 | 用于比较不同记忆架构的实验设置。

架构	参数	记忆	任务池	课程	教师	步数
Transformer-XL 带注意力机制的门控循环单元	总计 7600 万	1800 -	25B	空操作	None	50B
GRU		-				

D.5. 自动课程学习

空操作过滤详情。 此处提供空操作过滤的更多详情。对于来自 XLand 训练池的每个任务，我们分别在该任务上评估学习智能体（不向学习器发送任何经验）和空操作策略，各进行 10 个独立回合，每个回合长度为 1 次试验，分别得到得分 $\{R_0, \ldots, R_9\}$、$\{R'_0, \ldots, R'_9\}$。如果候选任务满足以下标准，我们就将其纳入训练：

1. $\max R'_i \leq \epsilon_1$ （空操作表现不能太好。） 2. $|\{i : R_i \geq \epsilon_2\}| \leq \epsilon_3$ （智能体表现不能太好。） 3. $|\{i : R_i \geq \max R'_i + \epsilon_0\}| \geq \epsilon_4$ 或 $|\{i : R_i \leq \min R'_i - \epsilon_0\}| \geq \epsilon_5$ （智能体与空操作的表现差异足够大。） 4. $\max R_i - \min R_i \geq \epsilon_6$ （智能体得分具有足够的方差。）

The ϵ_i是阈值，并在我们的训练设置中作为超参数。由于不同任务的持续时间通常不同，我们对 ϵ_0, ϵ_1, ϵ_2, ϵ_6 使用相对阈值，即试验时长的一定比例；其余任务则使用绝对阈值。一旦某项任务获准参与训练，就会按照该任务指定的完整试次数运行，并进行30个回合。这些运行产生的所有经验都会发送给学习器。所用超参数见表D.4。

表 D.4 | 空操作过滤超参数。

参数	价值	相对于试验时长
ϵ_0	0.01	Y
ϵ_1	1.1	Y
ϵ_2	0.4	Y
ϵ_3	5	N
ϵ_4	1	N
ϵ_5	3	N
ϵ_6	0.01	Y

PLR 详情。这里我们补充介绍用于训练 AdA 的优先级关卡回放（PLR）。PLR 使用一个 适应度分数 来近似智能体在给定任务上的遗憾值（Jiang 等人，2021a、b）。PLR 会维护一个固定最大容量的任务回放存档 $\mathcal{P}$。以概率 p（称为回放概率），系统会从 $\mathcal{P}$ 中抽取一项任务，同时考虑每项任务的适应度分数和陈旧度（见Jiang 等人 (2021b)，第 3 节），以训练学习智能体。陈旧度表示自上次抽取该任务以来经过的时间，并确保 $\mathcal{P}$中的所有任务都有准确的得分。最终概率通过结合适应度分数和陈旧度分数计算得出，其中陈旧度使用参数 $s \in [0, 1]$加权。

首先以概率 $1 - p$ 从训练任务集中采样一个候选任务，并评估其适应度分数，从而将任务添加到 P 中。如果该适应度分数高于 P 中所有任务的最低适应度分数，则将该候选任务添加到 P 中。如果 P 的新大小超过固定的最大值，则移除适应度最低的任务。请注意，在 PLR 中，如果一个任务始终未离开 P，它就可能被无限期地用于训练。

我们发现，使用最后一次试验的适应度，相较于使用首次试验或平均试验适应度，能带来更好的实证性能和样本效率。这很可能是因为在较早的试验中，智能体正在进行探索行为，因此基于误差的适应度较高；这不应被视为策略次优的迹象。然而，在经过一段时间的适应后，智能体在解决任务的最后一次试验中若基于误差的适应度仍然较高，则很可能表明策略次优，这类似于原始单试验优先级关卡重放中的遗憾近似。

为了将最后一次试验的适应度用作目标，我们需要对原始优先级关卡重放框架进行若干修改。该框架是为更同质领域中的单次试验设计的。我们将试验 k 的第 i 步的逐步适应度分数记为 $f_{i,\,k}$。首先，为避免偏向试验时长较长的情况，我们使用每次试验的平均适应度分数 $\tilde{f}_k \triangleq \sum_i \tilde{f}_{i,\,k}/N_k$，其中 N_k 是每次试验的步数。接下来，为确保我们不会优先选择 k 较低的值（这些值往往具有较高的最后一次试验平均适应度分数），我们再将 $\tilde{f}_k$ 按 $f_k \triangleq (\tilde{f}_k - \mu_k)/\sigma_k$ 进行归一化，其中 μ_k 和 σ_k 是针对每个试验索引、根据所有已评估任务计算得出的每次试验滚动均值和方差。最后，我们可以将 PLR 的适应度定义为 f_k，即归一化后的最后一次试验适应度分数。

如Jiang 等人 (2021a,b)所述，PLR 包含以下超参数：回放概率 p、最大重放规模 $N\text{max}$、最小重放规模 $N\text{min}$（我们将 $p = 0$ 设为 $|\mathcal{P}| < N\text{min}$）、$N\text{train}$（重新采样前运行训练任务的试次总数），以及 s 陈旧度系数。超参数设置见表 D.5 。我们对回放概率 $p \in \{0.1, 0.2, 0.5\}$、重放池规模 $N\text{max} \in \{1000, 10000, 50000\}$ 和陈旧度系数 $s \in \{0.1, 0.2\}$进行了网格搜索，并且在所有情况下都将 $N\text{min}$ 设为 $N\text{max}$ 的 90%。

表 D.5 | PLR 超参数。

参数	价值
p	0.2
$N_{\max}$	1000
$N_{\min}$	900
$N_{\text{训练}}$	30
s	0.2

PLR 适应度指标。接下来，我们需要定义逐步适应度分数 $f_{i,k}$。为此，我们采用最简单的类似遗憾的指标：1 步时序差分误差（Jiang 等，2021b；Schaul 等，2015）。具体而言，我们根据模型的即时价值预测来估计时序差分误差适应度：$|r_t + \gamma \hat{v}_0^{t+1} - \hat{v}_0^t|$。在某些情况下，这种做法可能并不理想，例如，智能体获得的奖励通常越高，时序差分误差也越大。因此，我们还提出基于 Muesli 动力学模型计算适应度指标。我们不单纯使用模型预测的准确度，而是考察预测对价值函数和动作预测的影响。我们将价值模型适应度定义为 $|\hat{v}_0^{t+1} - \hat{v}_1^t|$，即预测的下一状态与真实下一状态的价值估计之差。我们还定义了一个与价值无关的指标——动作模型适应度，如下所示：$JS(\hat{\pi}_0^{t+1}, \hat{\pi}_1^t)$，即预测的下一状态与实际下一状态下动作预测的差异，其中差异采用詹森 - 香农散度衡量（Farquhar 等，2021；Filos 等，2022；Pislar 等，2022；Xu 等，2017）。

在图D.1 中，我们展示了时序差分误差适应度、价值模型适应度和动作模型适应度的训练曲线。表D.6 展示了这些实验的实验设置。可以看到，时序差分误差和动作模型适应度指标都优于价值模型适应度。我们在 PLR 训练中选择时序差分误差，是因为它在零样本和少样本设置下的渐近性能都更好，而且先前研究也表明它表现良好（Jiang 等人，2021a）。

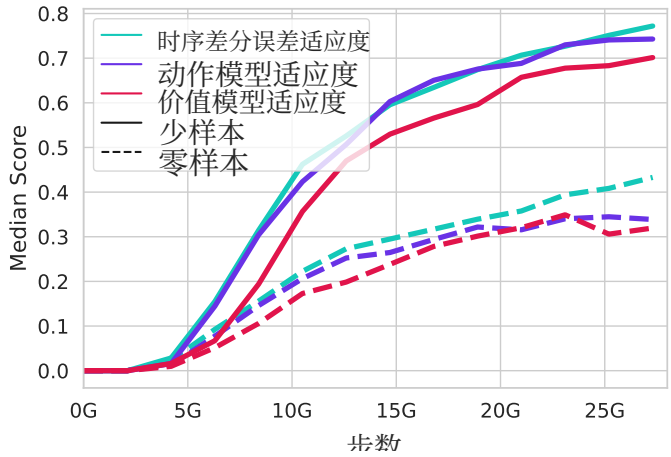
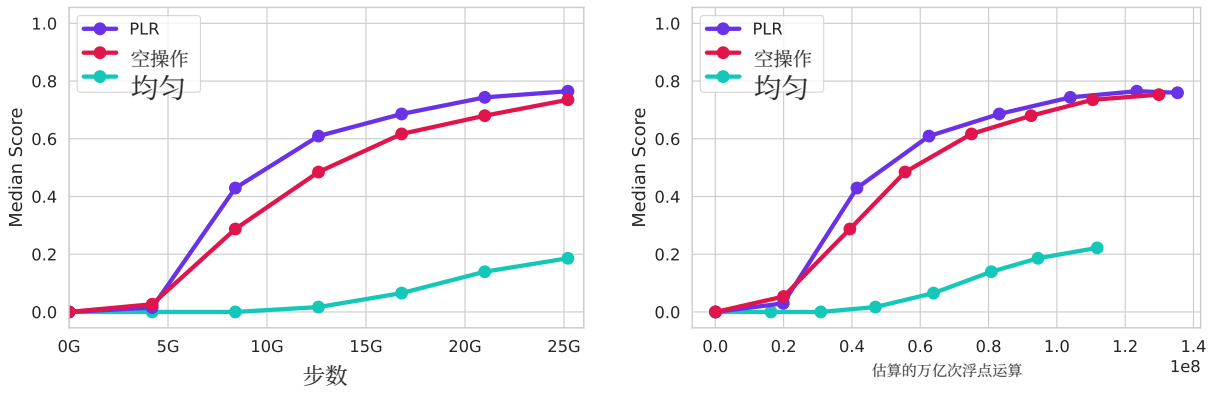


图 D.1 | 零样本泛化（$k = 1$）和少样本适应（$k = 13$）的 PLR 适应度指标比较。我们将主要智能体使用的时序差分误差适应度与两种使用 Muesli 动力学模型的方法进行比较。结果显示，动作模型适应度的少样本性能与时序差分误差适应度相当，但零样本性能较弱。

表 D.6 | 比较不同 PLR 适应度函数的实验设置。

模型参数	记忆	任务池	课程	教师	步数	适应度函数
2300万 TXL / 总计 7600 万	1800	25B	PLR	None	25B	时序差分误差 价值模型 动作模型



(a) (b)

图 D.2 | **（a）**不同课程选择下以步数衡量的样本效率。与任务均匀采样相比，空操作和 PLR 都显著提高了样本效率。少样本表示 $k = 13$得分，零样本表示 $k = 1$ 得分。**（b）**不同课程选择下以浮点运算次数衡量的样本效率。空操作初期优于 PLR，但在训练后期 PLR 的表现超过空操作。

课程效率。接下来，我们比较基线均匀采样和不同课程方法的训练样本效率，分别以学习器步数和浮点运算次数为单位。图D.2 展示了少样本（$k = 13$）和零样本评估任务的最后一次试验得分中位数随学习器步数的变化。可以看到，与均匀采样相比，空操作过滤和 PLR 课程都显著提升了训练样本效率，其中 PLR 在训练过程早期效率更高。将相同的少样本中位性能绘制为浮点运算次数的函数时，可以看到空操作在早期略占优势，但 PLR 在训练后期表现优于空操作。空操作最初的优势可能是因为它在任务评估中消耗了更多浮点运算次数（每个任务评估 10 次，而 PLR 为 1 次），从而在训练开始时找到了质量更高的训练任务。

涌现课程。在图D.3 中，我们展示了任务指标，并分析 PLR 和空操作过滤所选任务，从而扩展了图11中展示的结果。

D.6. 扩展实验的蒸馏教师

在所有扩展实验（第3.4 节和3.5节）中，我们都从同一个教师快照蒸馏策略，以确保实验具有可比性。教师的训练细节详见表D.7。该教师用于在训练最初的 4B 帧中启动我们的智能体。

D.7. 网络扩展

表D.8 展示了第 3.4 节中模型规模扩展实验的实验设置。有关各种模型规模对应参数数量的更多详情，请参见表 D.9。

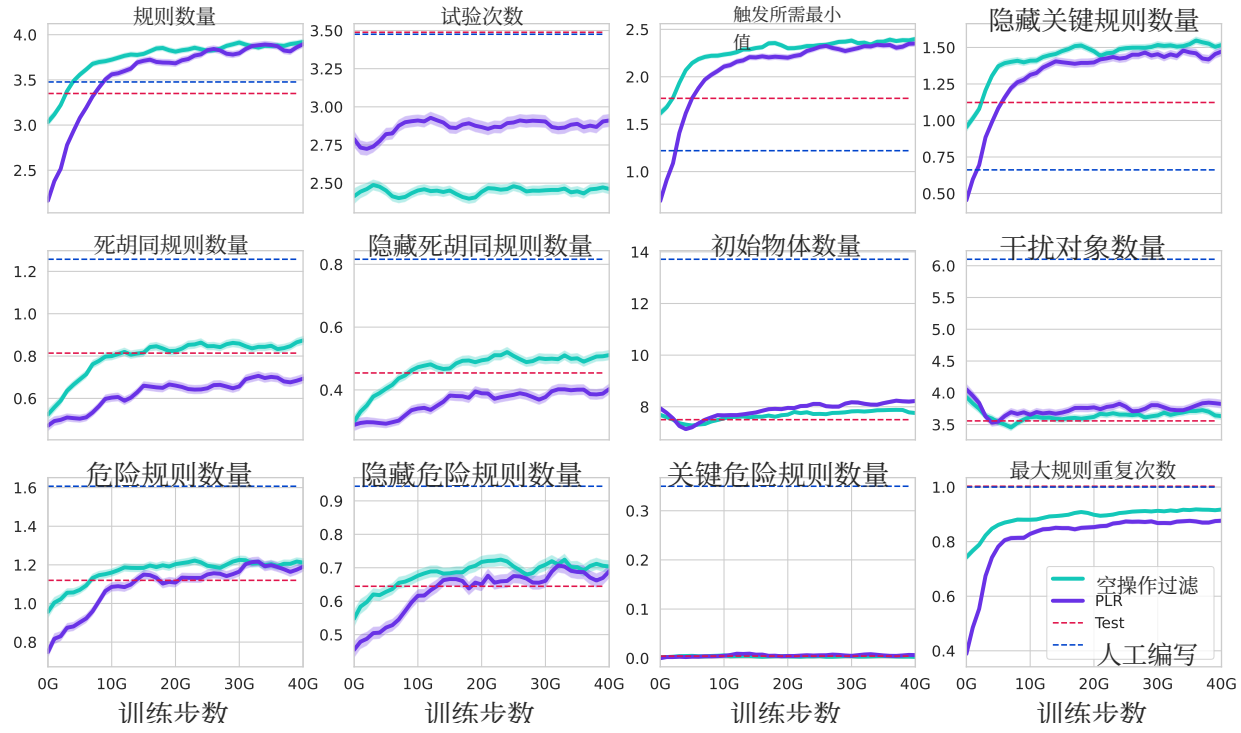


图 D.3 | 空操作过滤和 PLR 的涌现课程。图中展示了动态训练集的完整任务指标集，并对集合中的所有任务取平均值；阴影部分表示标准误。所有图中，指标值越高，表示任务难度越大。水平线表示相同指标值在测试集（虚线）和人工编写的评估任务集（点线）上的平均值。

表 D.7 | 用于扩展实验的蒸馏教师。

模型参数	记忆	任务池	课程	教师	步数
2300万 TXL / 总计 7600 万	1800	200M	空操作	None	23B

表 D.8 | 模型规模扩展的实验设置。

模型参数	记忆	任务池	课程	教师	步数
6M TXL / 总计 41M 2300万 TXL / 总计 7600 万 4200万 TXL / 总计 1.12 亿 5700万 TXL / 总计 1.41 亿 75M TXL / 总计 175M 169M TXL / 总计 353M 265M TXL / 总计 533M	1800	25B	空操作	表格 D.7	75B

Transformer-XL 的记忆是对先前注意力层输入的缓存，在每次前向传播时与键和值拼接。中间层的输入是前一层的激活值，而这些激活值本身包含有关过去的信息。以这种方式缓存 M 激活值，理论上可实现 $M \times L$ 的有效记忆范围，其中 L 是网络中的注意力层数。因此，为避免隐式扩展有效的 Transformer-XL 记忆长度，在我们的模型规模扩展实验中，我们固定 Transformer 的层数，并扩展

仅通过调整 Transformer 嵌入维度来改变参数（$d_{模型}$），同时将前馈层维度固定为 $4d_{模型}$，这符合 Transformer 架构的惯例（Vaswani 等人，2017年）。

表 D.9 | 不同模型规模的 Transformer 超参数。

模型参数	嵌入维度	模块	键维度	值维度	注意力头	前馈网络维度
6M TXL / 总计 41M	288	6	48	48	6	1152
2300万 TXL / 总计 7600 万	576	6	48	48	12	2304
4200万 TXL / 总计 1.12 亿	768	6	32	32	24	3072
5700万 TXL / 总计 1.41 亿	896	6	32	32	28	3584
7500万 TXL / 总计 1.75 亿	1024	6	32	32	32	4096
1.69 亿 TXL / 总计 3.53 亿	1536	6	48	48	32	6144
2.65 亿 TXL / 总计 5.33 亿	1920	6	48	48	40	7680

D.8. 扩展记忆长度

表 D.10展示了第 3.4 节中记忆长度扩展实验的实验设置详情。我们展示了每项实验的有效记忆时间步，其计算方法为缓存的网络激活值数量乘以 Transformer 模块的数量（6）。

表 D.10 | 记忆长度扩展实验的实验设置。

模型参数	记忆	训练任务池	课程	教师	训练步数
2300万 TXL / 总计 7600 万	600 1800 3000 4200	200M	空操作	表格 D.7	25B

D.9. 扩展任务池规模

表 D.11 展示了第 3.5 节中扩展任务池规模的实验设置详情。

表 D.11 | 扩展任务池规模的实验设置。

模型参数	记忆	训练任务池	课程	教师	训练步数
2300万 TXL / 总计 7600 万 7500 万 TXL / 总计 1.75 亿	1800	200M 25B 200M 25B	空操作	表格 D.7	25B

D.10. 扩展任务池的复杂度

表 D.12 展示了附录 E.3中扩展任务池任务复杂度的实验设置详情。在此实验中，我们比较的两个智能体使用了不同的蒸馏教师。因此，我们无法区分蒸馏和任务复杂度各自的影响。尽管如此，结果仍表明任务复杂度十分重要。用于该任务分布的教师

针对多种世界拓扑结构训练，其训练方式如表 D.7所示。单个房间任务分布的教师源自一脉相承的长链（> 6 代）蒸馏教师，其起点是使用 XLand 1.0 训练的智能体（OEL Team 等，2021年）。

表 D.12 | 扩展任务分布复杂度的实验设置。

模型参数	记忆	任务池	课程	步数
6M TXL / 总计 41M 2300万 TXL / 总计 7600 万 4200万 TXL / 总计 1.12 亿 5700 万 TXL / 总计 1.41 亿 7500 万 TXL / 总计 1.75 亿	1800	4000 个世界 × 50k 个游戏	空操作	23B
6M TXL / 总计 41M 2300 万 TXL / 总计 7600 万 4200万 TXL / 总计 1.12 亿 5700万 TXL / 总计 1.41 亿	1800	1 个世界 × 5k 次初始化 × 50k 个游戏	空操作	23B

D.11. 蒸馏使智能体扩展成为可能

TableD.13展示了第 3.6 节蒸馏实验的实验设置。

表 D.13 | 蒸馏实验的实验设置。

模型参数	记忆	训练任务池	课程	教师	步数
TXL 2300万 TXL / 总计 7600 万 TXL 265M TXL / 总计 533M	1800	参见第 D.3	PLR (D.5)	表 D.2 None 表 D.2 None	22B

D.12. 使用跳跃记忆进行更多试次的训练

表D.14 展示了第3.7节中记忆扩展实验的实验设置详情。该设置与表D.14中的设置相同，不同之处在于训练步数，以及正文中讨论的记忆架构和训练试验的变化。

表 D.14 | 采用跳跃记忆进行更多试次训练的实验设置。

模型参数	记忆	训练任务池	课程	教师	步数
2300万 TXL / 总计 7600 万	1800 1800 × 4 = 7200	200M	空操作	表格 D.7	50B

E. 补充实验

E.1. 多智能体适应

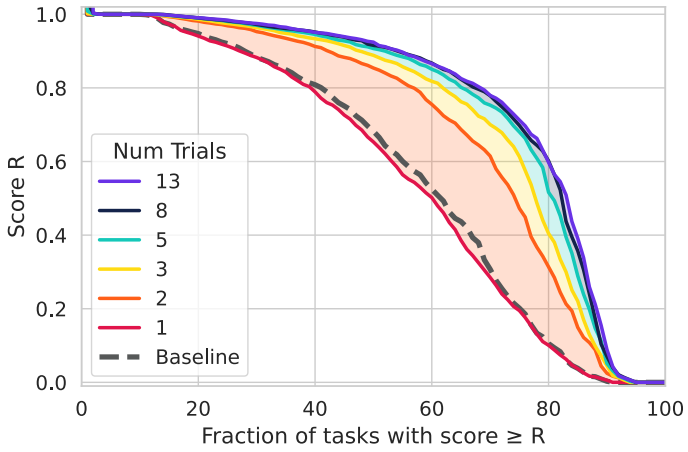


图 E.1 | 我们报告了在不同试次数下进行评估时，多智能体任务测试集上归一化任务分数的分布。所有任务均通过合作式自博弈进行评估。 y轴表示最后一次试验的总奖励，相对于在测试任务上微调的智能体所获得奖励的比例（近似于“无限试次”性能）。曲线越向右上角移动，表示性能越好。增加试次数后，智能体在最后一次试验中取得更高分数，表明其在任务分布的大部分范围内实现了测试时适应（阴影区域）。

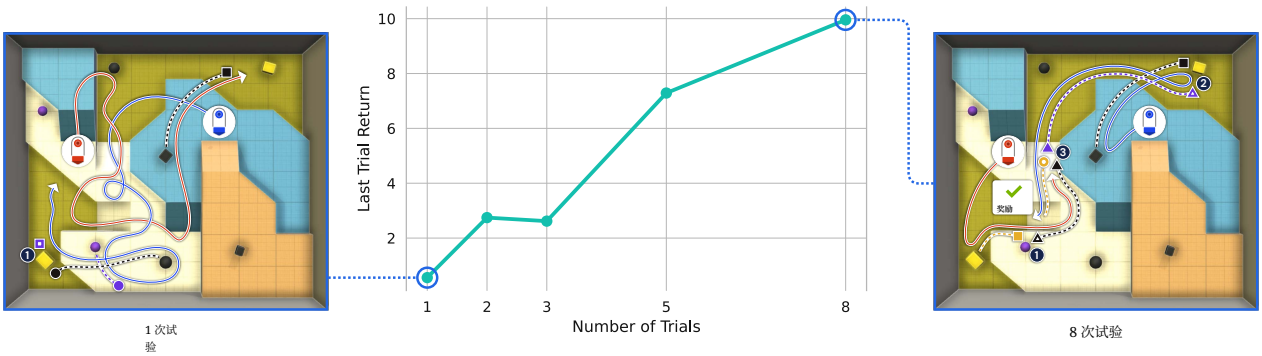


图 E.2 | AdA 在探测任务上的平均性能和代表性行为`Irreversible Production for Two` 在不同试次数下通过自博弈进行评估时的表现。增加试次数后，AdA 的性能有所提升，表明其实现了测试时适应。俯视图图像展示了总试次数不同时，最后一次试验的代表性轨迹。相应的 视频 展示了这种情况下 $k = 8$ 一个回合中所有试次的行为。

图E.1 展示了在多智能体任务的测试任务集上，适应能力在广泛的百分位数范围内的表现。图E.2 展示了 AdA 在某个特定探测任务中的最后一次试验性能。为生成这些图，AdA 按照第3.1 节（多智能体）所述进行训练。

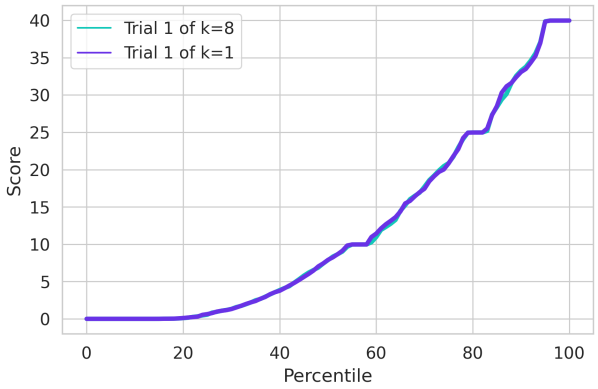


图 E.3 | 比较了包含 1 次试验的回合与包含 8 次试验的回合中的首次试验得分。两条线几乎完全重合，这表明我们的智能体不会利用试次数量条件信息，在可进行更多试验时，于早期试验中将其策略调整得更具探索性。

E.2. 以尝试次数为条件不会影响智能体表现

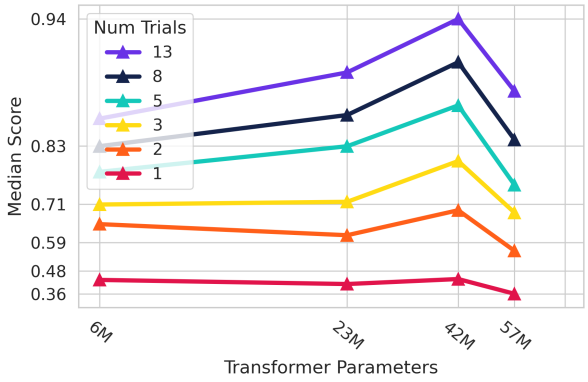
图 E.3 展示了在留出测试集中，对于每个百分位数，包含 1 次试验的回合（$k = 1$）和包含 8 次试验的回合（$k = 8$）的第 1 次试验中 AdA 获得的得分。这些线的重合表明，AdA 不会利用其观察到的试验条件信息，以任何会影响其得分的方式调整其行为。如果智能体在拥有更多试验次数时采用更具探索性的策略，我们可能会预期，使用 $k = 8$ 的第 1 次试验得分会低于使用 $k = 1$的第 1 次试验得分。

这可能是我们 XLand 2.0 任务的最优策略，也可能揭示了我们训练流程的局限性。可以设想这样一种情形：已知总共有 8 次试验，贝叶斯最优策略会在试验 1 中表现得奖励较少、探索性更强；而如果它被告知只有一次试验可以获取奖励，其行为就会有所不同。例如，智能体可能会放弃早期获得随机奖励的机会，以换取在发现某些关键信息后，确保稍后获得确定性奖励。我们在训练过程中并未直接鼓励这种行为。事实上，我们可能还抑制了这种行为，因为 AdA 会从一个回合中的所有奖励（而不只是最后一次试验的奖励）中学习，而且折扣因子小于 1，这可能导致短视行为。

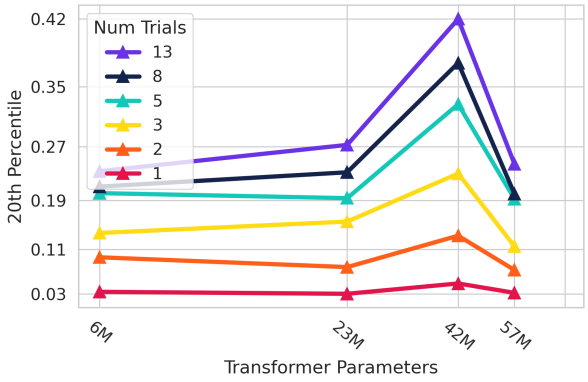
E.3. 扩展任务池的复杂度

我们的方法还可以沿着一个维度继续扩展，即任务分布的整体复杂度。我们将主要任务分布（如第 2.1 节和附录 A.2 所述）与一个子集进行比较。该子集保留了相同的目标、生成规则和物体，但通过采用单一世界拓扑结构（即空房间）消除了所有导航复杂度。请注意，任务数量按世界数量与游戏数量的乘积计算。为了区分复杂度扩展与单纯增加任务数量的影响，我们为空房间增加了 5,000 个独特的物体初始化点。这些点作为“4,000 个世界”的替代指标，而且有意设计得远不如主要训练池中的 4,000 个世界多样和复杂。[7] 有关实验设置的更多详情，请参阅附录 D.10和表 D.12。

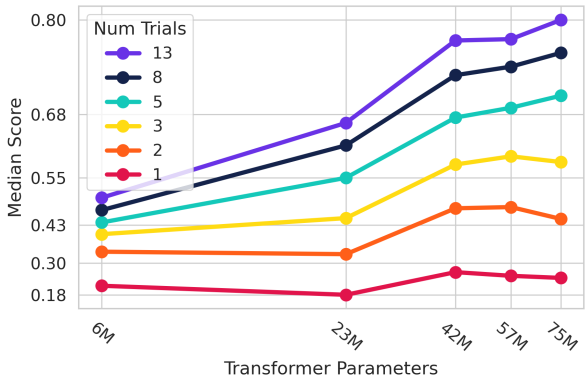
[7]请注意，我们在此使用的世界拓扑分布小于第3.4节模型规模实验中使用的分布，因此这些章节中的结果无法直接比较。



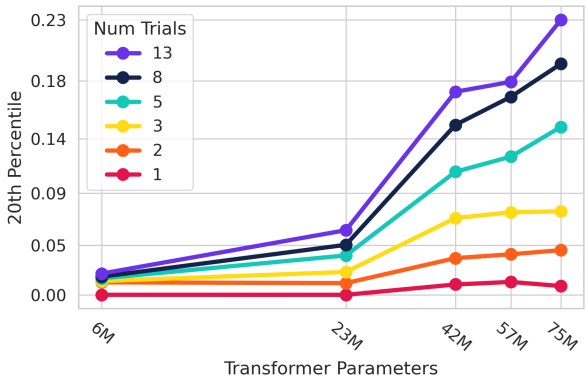
(a) 仅包含空房间的任务分布会阻碍扩展（中位数）。



(b) 仅包含空房间的任务分布会阻碍扩展（第20百分位数）。



(c) 包含多种世界拓扑结构的任务分布有助于扩展（中位数）。



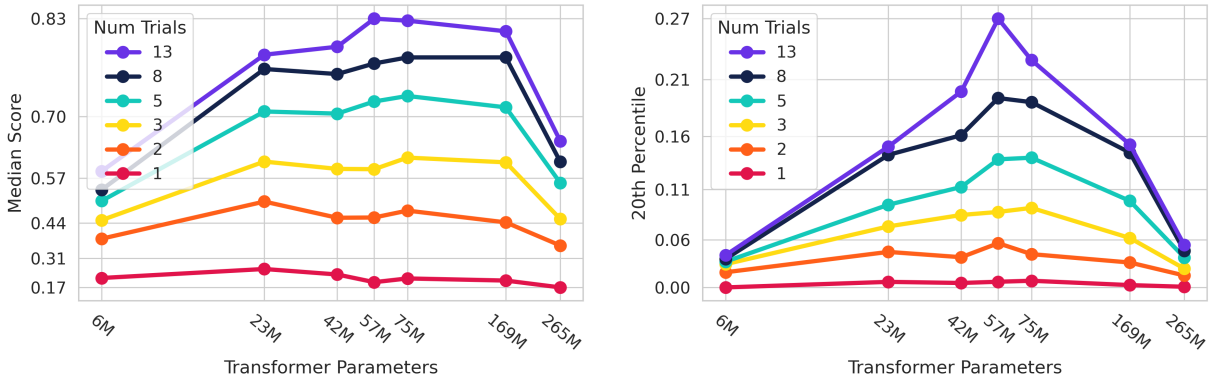
(d) 包含多种世界拓扑结构的任务分布有助于扩展（第20百分位数）。

图 E.4 | 如果分布不够复杂，即使考虑了任务总数，扩展模型规模的收益也会受到瓶颈限制。

在图 E.4中，我们通过比较在两种分布上训练的智能体扩展模型规模的效果，并分别在各自的测试集上进行评估，展示了较低的环境复杂度可能成为扩展的瓶颈。对于空房间中的中位数（图 E.4a）和 20th 百分位数（图 E.4b），我们发现，超过某个临界点（4,200 万个 Transformer 参数）后，扩展模型规模开始降低性能。相比之下，在包含多种世界拓扑结构的分布中（图 E.4c 和 E.4d），模型规模增大后，性能仍持续提升，至少在达到 7,500 万个 Transformer 参数时依然如此。因此，对于环境复杂度没有上限的开放式设置（例如多智能体系统），可能对扩展自适应智能体尤为重要。

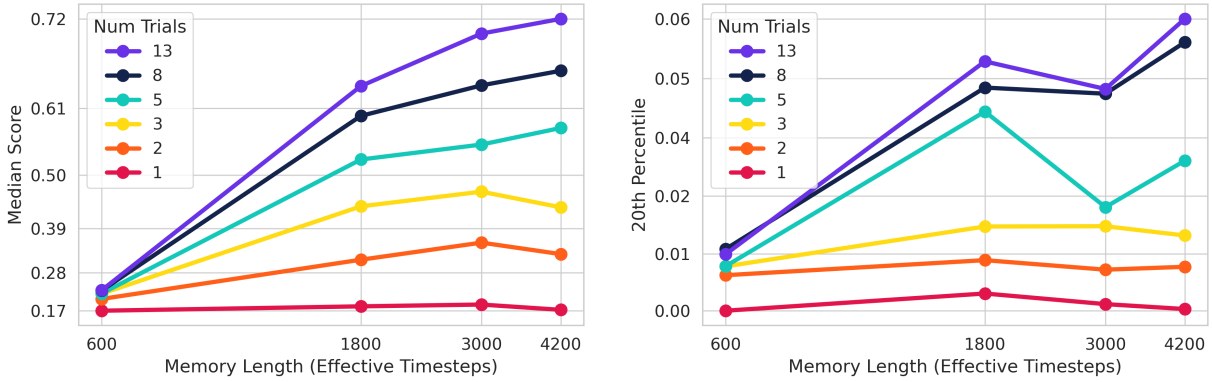
E.4. 计算成本

在扩展实验（第 3.4 节和 3.5节）中，我们比较的是训练步数相同的智能体。虽然这样可以控制模型的样本效率，但此处我们分析了各项实验的浮点运算次数（FLOPs）计算成本，并复现了部分扩展结果，同时控制计算成本。从这一角度来看，我们发现模型并非越大越好。对于每种模型规模和记忆长度，我们使用 JAX（Bradbury 等人，2018年）的成本分析，估算学习器步和行动器步每帧所需的浮点运算次数（表 E.1）。



(a) (b)

图 E.5 | 在控制学习器和执行器（包括自动课程评估执行器）的浮点运算总次数时，扩展 Transformer 模型规模。



(a) (b)

图 E.6 | 在控制学习器和执行器（包括自动课程评估执行器）的浮点运算总次数时，扩展 Transformer-XL 记忆。

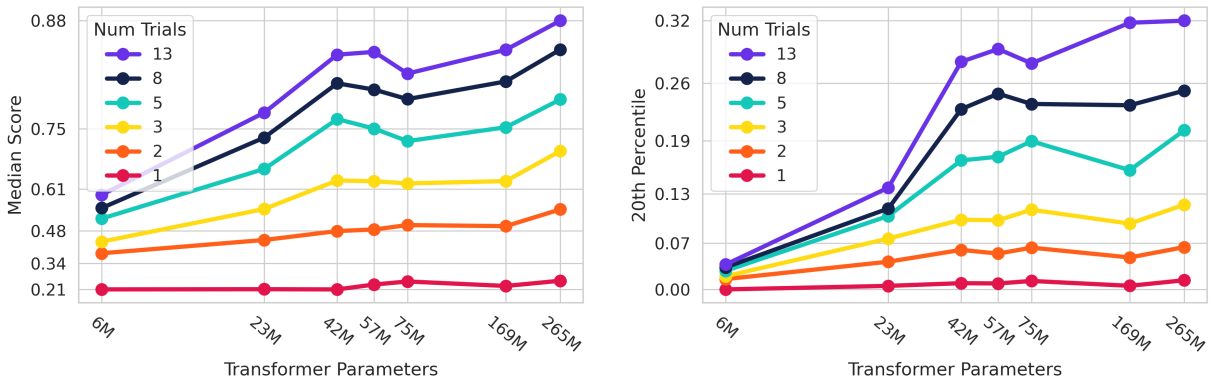
表 E.1 | 不同模型规模和记忆长度下每帧的浮点运算次数。

模型参数	记忆	学习器每帧浮点运算次数	执行器每帧浮点运算次数
6M TXL / 总计 41M		1,138,005,632	736,987,392
2300万 TXL / 总计 7600 万		1,380,255,078	2,580,445,440
4200万 TXL / 总计 1.12 亿		1,623,461,663	4,489,239,552
5700万 TXL / 总计 1.41 亿	1800	1,821,422,230	6,063,742,976
7500 万 TXL / 总计 1.75 亿		2,048,253,663	7,879,863,808
1.69 亿 TXL / 总计 3.53 亿		3,243,189,525	17,560,326,144
2.65 亿 TXL / 总计 5.33 亿		4,443,008,234	27,358,181,376
2300万 TXL / 总计 7600 万	600	1,278,491,404	963,739,136
	3000	1,481,009,258	4,181,042,688
	4200	1,582,270,213	5,789,702,144

我们将表 E.1 中的数值乘以各项实验的学习器/行动者步数；随后，对于给定的比较，我们取所有实验共有的最大值（通常

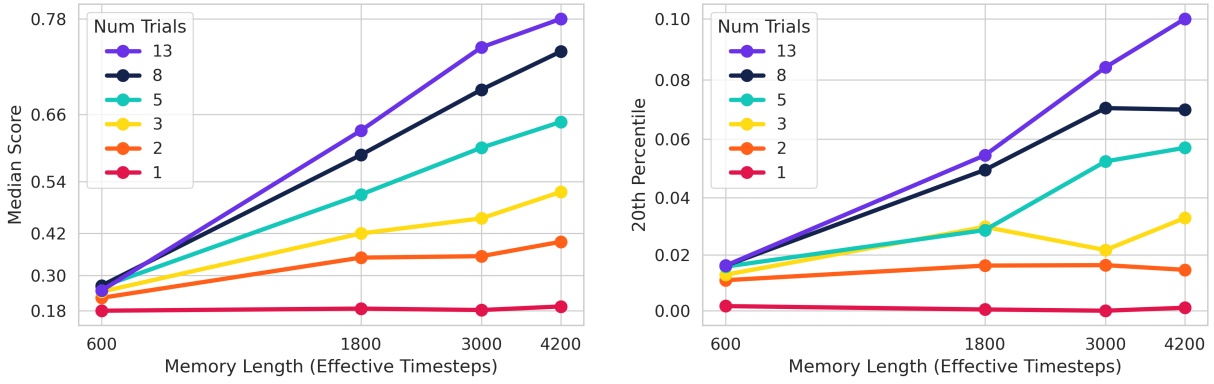
与最小模型相关联的总浮点运算次数，并在这一浮点运算次数下比较各个模型。浮点运算次数匹配的模型缩放实验结果如图 E.5 所示。我们看到，当模型规模超过约 57M 个 Transformer 参数（总计 141M 个参数）的“最佳点”后，性能会下降。图 E.6 中的浮点运算次数匹配的记忆缩放实验结果表明，在固定计算预算下，增加上下文长度仍然有益。这些实验所使用的计算资源详情见表 E.2（模型规模）和E.3 （记忆）。

我们注意到，表 E.1 显示，随着模型规模增大，行动器步数的浮点运算次数扩展性较差。我们怀疑，这可能是因为在 TPU 上对 Transformer 进行单步查询的优化效果不佳，相比之下，学习器采用展开长度为 80 的操作批处理。为了考虑这一点，以及因基于模型质量的课程评估差异而导致的行动器步数可能存在的偏差，我们还提供了仅统计各模型学习器步数浮点运算次数的图：图 E.7 （模型规模）和 E.8 （记忆长度）。这些图可能更具参考价值，并显示性能仍会随模型规模和记忆长度增加而提升，尽管提升幅度不及直接控制样本效率时那么明显。这些实验所用计算资源的详情见表 E.4（模型规模）和 E.5 （记忆长度）。



(a) (b)

图 E.7 | 在控制学习器浮点运算次数的情况下扩展 Transformer-XL 模型规模。



(a) (b)

图 E.8 | 在控制学习器浮点运算次数的情况下扩展 Transformer-XL 记忆长度。

表 E.2 | 各模型规模对应的总浮点运算次数及学习器步数。

模型参数	记忆	总浮点运算次数	学习器步数
6M TXL / 总计 41M	1800	$2。0 \times 10^{20}$	97B
2300万 TXL / 总计 7600 万			44B
4200万 TXL / 总计 1.12 亿			27B
5700万 TXL / 总计 1.41 亿			23B
7500万 TXL / 总计 1.75 亿			18B
169M TXL / 总计 353M			9B
265M TXL / 总计 533M			5B

表 E.3 | 每种记忆长度对应的总浮点运算次数所对应的学习器步数。

模型参数	记忆	总浮点运算次数	学习器步数
2300万 TXL / 总计 7600 万	600	$7。9 \times 10^{19}$	31B
	1800		17B
	3000		11B
	4200		8B

表 E .4 | 各模型规模在仅计算学习器步骤浮点运算次数时对应的学习器步数。

模型参数	记忆	学习器步数的浮点运算次数	学习器步数
6M TXL / 总计 41M	1800	$1。0 \times 10^{20}$	92B
2300万 TXL / 总计 7600 万			76B
4200万 TXL / 总计 1.12 亿			65B
5700万 TXL / 总计 1.41 亿			58B
7500万 TXL / 总计 1.75 亿			51B
1.69亿 TXL / 总计 3.53 亿			32B
2.65亿 TXL / 总计 5.33 亿			23B

表 E.5 | 每种记忆长度下，根据仅计算学习器步的浮点运算次数得出的相应学习器步数。

模型参数	记忆	学习器步数的浮点运算次数	学习器步数
2300万 TXL / 总计 7600 万	600	3.9×10^{19}	39B
	1800		36B
	3000		29B
	4200		25B

E.5. 重复蒸馏

在第 3.6 节中，我们展示了将智能体蒸馏到一个相同的学生模型中，可以大幅提升智能体的性能。在此，我们研究重复应用这一过程可能带来的益处。为此，我们继续图 16 所示的实验，并增加第三代：使用前一代学生在 250 亿帧后取得的快照；计入教师的经验后，这相当于总计 500 亿帧的经验。图 E.9 显示，重复应用这一过程确实能带来额外收益；但我们观察到，随着代数递增，收益逐渐递减。

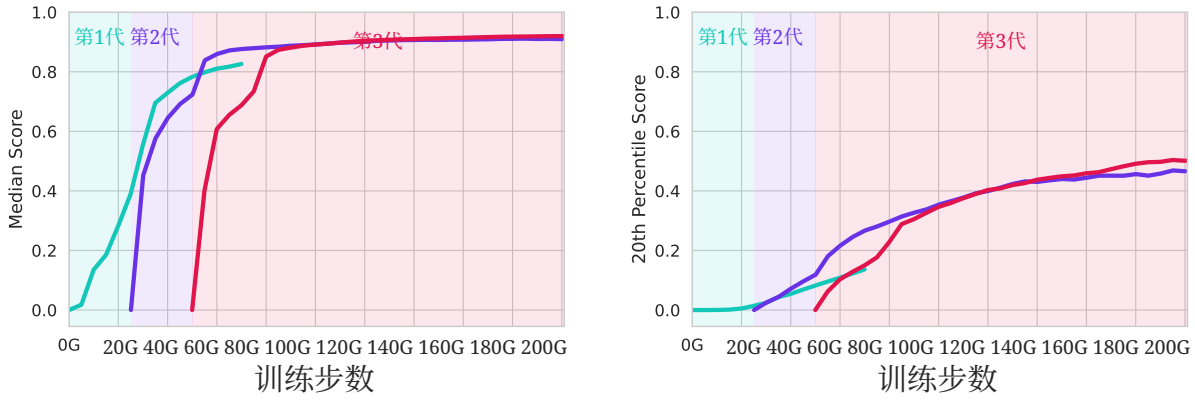


图 E。9 | 使用 2300 万参数的 Transformer- XL，在三代智能体中测得的归一化少样本得分。第一代和第二代对应于图 16. 中展示的智能体。第三代是在第二代经过 250 亿步训练后，由其蒸馏得到的。横轴表示经验总量，起点为原始教师收集的经验。第三代的表现优于第二代，但提升幅度小于第二代与第一代之间的差距。 *x*-轴

F. 人类时间尺度适应

在此　本节将进一步详细说明我们关于人类时间尺度适应的主张　　.

F.1. 探测任务

表格 F.1和 F.2 分别详细描述了单智能体和多智能体探测任务。除非另有说明，所有探测任务都设置在包含许多物体的复杂世界中，并使用多条对玩家完全隐藏的生成规则。

表 F.1 | 单智能体探测任务

Name	**描述**
错误配对消失	玩家的目标是拿到一个黑色立方体，而初始物体中并不存在这种物体。这里有两条（隐藏的）规则。玩家需要找出 能够触发创建立方体规则的正确世界状态，而不是触发摧毁必要输入物体规则的世界状态。 这一切都发生在一个充满挑战的 世界布局中，其中存在单向落差，且视野有限。
错误配对消失，部分隐藏	“错误配对消失”，但我们并未完全隐藏所有规则，而是只 隐藏两条规则的输入对象（输出和条件均完全 可见）。
不可逆生产	类似于“错误配对消失”，但此任务有多个死路 （会产生不可解状态的规则）。
不可逆生产，所有规则可见	“不可逆生产”，但所有规则都对玩家完全可见。

推而不举	绝大多数训练和评估任务都需要举起物体。 这里有两条隐藏规则：任何物体一旦被举起就会被摧毁。为了创造 目标状态，需要一些“横向思维”：玩家需要 发现可以用身体推立方体。
推而不举，带有 干扰物	与“推而不举”类似，但这里大量的干扰物体 使世界中的探索任务对任何 忽略目标中提及物体的玩家来说都更具挑战性。
间隔工具	需要将两个物体靠近，但抬起它们或接触 它们（用化身的身体）会摧毁它们。解决方案是使用另一个 世界中的物体作为工具，将它们推到一起。
变形以进行 运输	同样，需要将两个物体彼此靠近，但抬起它们并 接触它们会摧毁它们。这里的解决方案是利用一组规则 将其中一个物体变成可以安全携带的东西， 然后在它处于正确位置后，再将其变回原来的物体。
小型工作站	一个极其困难的物体处理任务。玩家出生位置附近的 8 个物体 需要以不同方式进行组合，并且需要遵循 5 条规则 被触发（有时会触发多次）以创建目标物体。此任务设置在 高原顶部，而且只要接触到某个物体，任务就很容易失败 过于用力并将其扔下高原边缘。
小型工作站，所有 规则可见	与“小型工作站”相同，但玩家可以看到所有规则。
合成金字塔	八个物体需要先递归地组合成四个，再组合成两个 然后最终组合成一个物体。这需要触发一连串共 7 条 规则。这对人类来说似乎很容易。但由于缺乏中间奖励 这对智能体而言成为一项困难的层级信用分配任务。
合成树，所有规则 可见	类似于 Minecraft 等电子游戏中的合成树，这个多 步骤任务需要依次触发不同规则来创建目标 物体。世界中每种物体都存在多个，因此有许多不同的 解决方案可行。
合成树，隐藏 捷径	与“制作树，所有规则可见”相同，但额外存在一条（隐藏的） 规则。这使玩家能够走捷径，从而比只执行可见规则更快地完成 比只执行可见规则更快地完成任务。
反物质	在一个满是黄色和黑色球体的世界中，目标是确保任意一对黑色 以及黄色球体彼此“看见”（没有直接视线）。除此之外 移动物体并用化身阻挡视线后，还存在 一条产生式规则，可摧毁任何接触的黄色球体与黑色球体配对 （其概念类似于物质和反物质粒子）。这一切 都置于一个需要高级导航技能的世界中。
带有反物质的 创建	与“反物质”类似，只是这里存在第三种物体（紫色金字塔） 它会复制任何与其接触的球体。此外，该任务设定在 两座高原上，更容易阻挡视线，并降低 导航难度。

大海捞针式金字塔	要创造所需的黄色金字塔，玩家需要找到并拿住 紫色立方体。这个世界中有多个干扰物体和干扰规则 ，玩家不仅需要应对艰难的探索 挑战，还要应对非常嘈杂的环境。
保护鸡蛋	玩家的任务是在世界中托住唯一的黄色球体。还有大量 其他球体存在于世界中。它们会摧毁黄色 球体。它们彼此接触或靠近玩家时， 就会复制。这可能导致物体数量不断增加， 最终填满整个世界。
三个混杂的球体	世界中存在三个球体。拿起其中一个会生成目标物体。 在10秒时间限制内只能触及一个球体，这意味着 第一次试验中的最优策略是均匀随机选择。
两扇门	目标物体隐藏在两个大型物体（即“门”）中的一个后面 位于世界的两端。只能到达其中一个 及时到达，因此玩家需要决定每次试验是探索其中一个 试验.
相同签名： 匹配颜色	所有“相同签名”任务的世界布局和目标都完全相同，而且 完全隐藏的规则数量。这意味着对于一位 刚开始的玩家来说，它们看起来完全相同。这个任务只要求将两个颜色相同的物体 移到彼此靠近的位置，以创造目标物体，只需使用三条产生式规则中的一条。 三个产生式规则之一。
相同特征：三步 步	这个“相同签名”任务变体要求玩家触发全部三条 （隐藏的）生成规则，以创造目标物体。
相同签名：两条 死路	在这个“相同签名”任务变体中，树的规则有两条是死路， 会导致世界状态无法解决。只有一条规则有用（事实上 完成任务所需）。
相同签名： 摧毁以保护	要完成这个“相同签名”任务变体，玩家首先需要摧毁 一个黑色球体（靠近它即可），然后再创建目标物体（一个黄色 立方体）。否则，当黑色球体“看到”黄色立方体时，两者 被摧毁。即使对 人类玩家来说，这也是一个极难应对的信用分配挑战（很难察觉正在发生什么）。
不要行动	这个任务已预先解决：玩家从一开始就能获得奖励—— 如果玩家抬起或触碰世界中的两个物体之一，该物体 就会被摧毁，之后便不可能再获得奖励。
别偷看	这个任务已“预先解决”：玩家从一开 始就能获得奖励。然而，玩家看到任何物体都会将其摧毁， 这时就不可能获得点数奖励。因此，最优策略是什么都不看， 只看天空。
导航：找到 cube	这个记忆任务不使用生成规则，而是采用一个大型世界， 目标物体（一个立方体）藏在一条非常蜿蜒的路径之后。

导航：寻找带诱饵的立方体	这项记忆任务没有使用生成规则。它设定在一个广阔的世界中目标物体（一个立方体）隐藏在一条非常蜿蜒的路径之后。该物体从生成点可以看到，但开始穿行后便会消失在视野之外地形。
导航：按住上键high	此记忆任务不使用生成规则。目标物体（一个金字塔—中）隐藏在高原顶部，只有在几乎那里。
物体恒存：黄色立方体	此记忆任务未使用生成规则。目标物体（一个黄色立方体）从生成点处可见。移动一段距离后，需要在两条路径之间做出选择，正确的路径是向右。此时，立方体已不再可见。
物体恒存：黑色立方体	此记忆任务不使用生成规则。其世界布局与上方的任务相同，只不过这里要求玩家找到黑色立方体，这需要向左走。

表 F.2 | 多智能体探测任务

Name	描述
传过墙	玩家被一堵不透明墙隔开。他们看不见彼此，只能看到世界的各自半边。解决方案要求将可用物体从一名玩家所在的一侧“传递”给另一名玩家，以便将它们组合成目标物体。然后，玩家必须确保由正确的玩家持有这个物体。
传过墙反复	与“传过墙”类似，但这里 3 个初始物体中有 2 个是“冻结”的（无法移动）。这规定了一种非常具体的解决策略，要求玩家按特定顺序将 3 个物体传过墙。
协调生产	此任务要求每位玩家靠近一个球体，将其变成金字塔，然后让两个金字塔彼此接触以创造目标物体。球体在复杂世界中略微隐蔽，因此需要进行一些探索。
协同带有死胡同	类似于「协同生产」，但在这里，每位玩家靠近某个初始物体时都会将其摧毁。这些死胡同规则使这成为一个难得多的探索问题。
协同交换	这要求每位玩家握住一个现有物体，创造出一个新物体，然后握住另一位玩家创造的物体，将其变成另一个中间物体，最后将两个物体组合成目标物体。尽管两位玩家都能完全访问这个世界，但只有两位玩家都积极参与，才能解决这一任务。
Overcooked：协调环	灵感来自视频游戏 Overcooked（Carroll 等人，2019年；Strouse 等人，2021年），在这项任务中，两名玩家都需要“为一位饥饿的顾客端上番茄汤”。这通过一条可重复执行的四步生产规则链实现，该规则链要求两名玩家谨慎地穿过各自的共享空间（一个环形布局），以免挡住另一名玩家。

Overcooked：协调环，所有规则均可见	在“Overcooked：协调环”中，所有生成规则对两位玩家均不可见；而在此处，它们完全可见（以匹配原版 Overcooked 游戏）。
Overcooked：狭窄room	类似于“Overcooked：协调环”，但布局共享空间和初始物体数量有所不同，并采用了不同的形状和颜色。
Overcooked：狭窄房间，所有规则均可见	在“Overcooked：狭窄房间”中，所有生成规则对玩家均不可见对两位玩家都是如此；而在这里，这些规则完全可见（以匹配原版 Overcooked 游戏的动力学）。
Overcooked：强制协同	与其他“Overcooked”任务变体类似，但在此，两个玩家都只能活动在世界的特定区域内，因此被迫协同完成此任务。没有玩家能独自完成任务，因为他们无法触及所有初始物体。
Overcooked：强制协同，所有规则可见	在“Overcooked：强制协作”中，所有生成规则对双方玩家都是隐藏的而在这里，所有规则都完全可见（以匹配原版 Overcooked 游戏的动力学）。
踢球	此任务发生在一个大型世界中，其中有两座冰冻金字塔，分别位于相对的世界的两侧。两名玩家都想把所有数量充足的紫色球体带到黄色金字塔那里。但举起它们会摧毁金字塔，所以他们需要让球体从化身身上弹开来“踢”它们。想象一下“足球训练”。
柠檬食客	第一名玩家会摧毁所有黄色球体（数量很多）碰到它们时也是如此。这也是两名玩家的目标。因此，他们需要合作，尽快将所有黄色球体带给第一名玩家。
谨慎的柠檬食者	与“柠檬食客”类似，但两个球体之间的任何碰撞都会使它们从黄色变成紫色。紫色球体“不可食用”，因此玩家需要要小心不要制造出这些东西，否则他们会失去奖励。
柠檬食客和制造者	与“柠檬食客”类似，但一开始只有紫色球体。只有第二个玩家才能通过举起紫色球体将其变成黄色球体，实际上这意味着必须为第一个玩家“制造食物”。
双人反物质	与单人“反物质”任务性质相同，但设置在不同的世界布局中，由两名玩家共享同一个目标。
反物质生成双人	与单人“反物质生成”任务性质相同，但设置在不同的世界布局中，由两名共享同一目标的玩家进行。
带有双人复制品的反物质	与“双人反物质”相同，但在此处，任意两个相同颜色的球体碰撞后，都会生成另一个相同颜色的球体。这可能会引发球体数量的失控增长，使得任务变得非常难以解决。
双人不可逆生产	在本质上与单人“不可逆生产”任务相同，但设置在不同的世界布局中，并由两名共享相同目标。

不可逆 双人生产， 所有规则均可见	类似于“双人不可逆生产”，但所有生成规则对 两位玩家均可见。
错误配对 双人版	与单人版“错误配对消失”任务的本质相同，但 设定在不同的世界布局中，由两名共享同一 目标。
错误配对 双人版错误配对消失， 部分隐藏	类似于“双人版错误配对消失”，但仅隐藏 生成规则中的输入对象，输出和条件则保持可见。
合成金字塔： two	与单人“合成金字塔”任务本质相同，但设定在一个 不同的世界布局，其中两名玩家拥有相同的目标。
信息 不对称	一个简单的世界，玩家需要在其中执行一条两步 产生式规则，同时存在多个死路。虽然第一名 玩家知道所有规则，但这些规则对第二名玩家完全隐藏 玩家。此任务旨在测量一种特定形式的第三人称 模仿。
信息 不对称与 重复	虽然“信息不对称”最多只允许完成四次， 这个任务变体（在时间不限的情况下）可以无限次重复， 从而提供更多模仿的机会。
组合输出	与“协同生产”类似，但设定在一个更大的世界中， （通过使用冻结物体）要求两名玩家反复 移动相当远的距离，以创建用于共同创作的输入对象。
两台机器	这个世界散落着许多颜色和形状各异的物体。冰冻的 世界一端的黄色金字塔会将这些物体变换成 一种中间物体。然后，玩家需要把这种中间物体带到 世界另一端冰冻的黑色金字塔那里，以便“兑现 它”并获得即时奖励，此时中间物体会 被摧毁。因此，要获得更多奖励，玩家必须重复这一 过程。
两台机器，具有 信息 不对称	与「两台机器」类似，但所有规则对一名玩家可见，而且所有规则 对另一名玩家不可见。这造成了信息不对称 因此也带来了进行第三人称模仿的机会。

F.2. 比较每个探测任务上的人类和智能体得分

在图 F.1 中，我们展示了人类和我们的智能体在每个30 个评估探测任务中，最后一次试验的总奖励随试次数量变化的原始数据。

F.3. 量化随机性

任务变异。 图 F.2a 显示，单个智能体在同一任务上重复 50 次所获得的得分存在相当高的方差。这是由于每次环境重置后化身的随机生成旋转，以及智能体策略中的随机性所导致的。为了解决这一问题，我们对每个（任务， k）组合重复 50 次，并对这些重复次数的得分取平均值。这将标准差降低到了图 F.2b所示的水平。在这里，平均任务的标准差

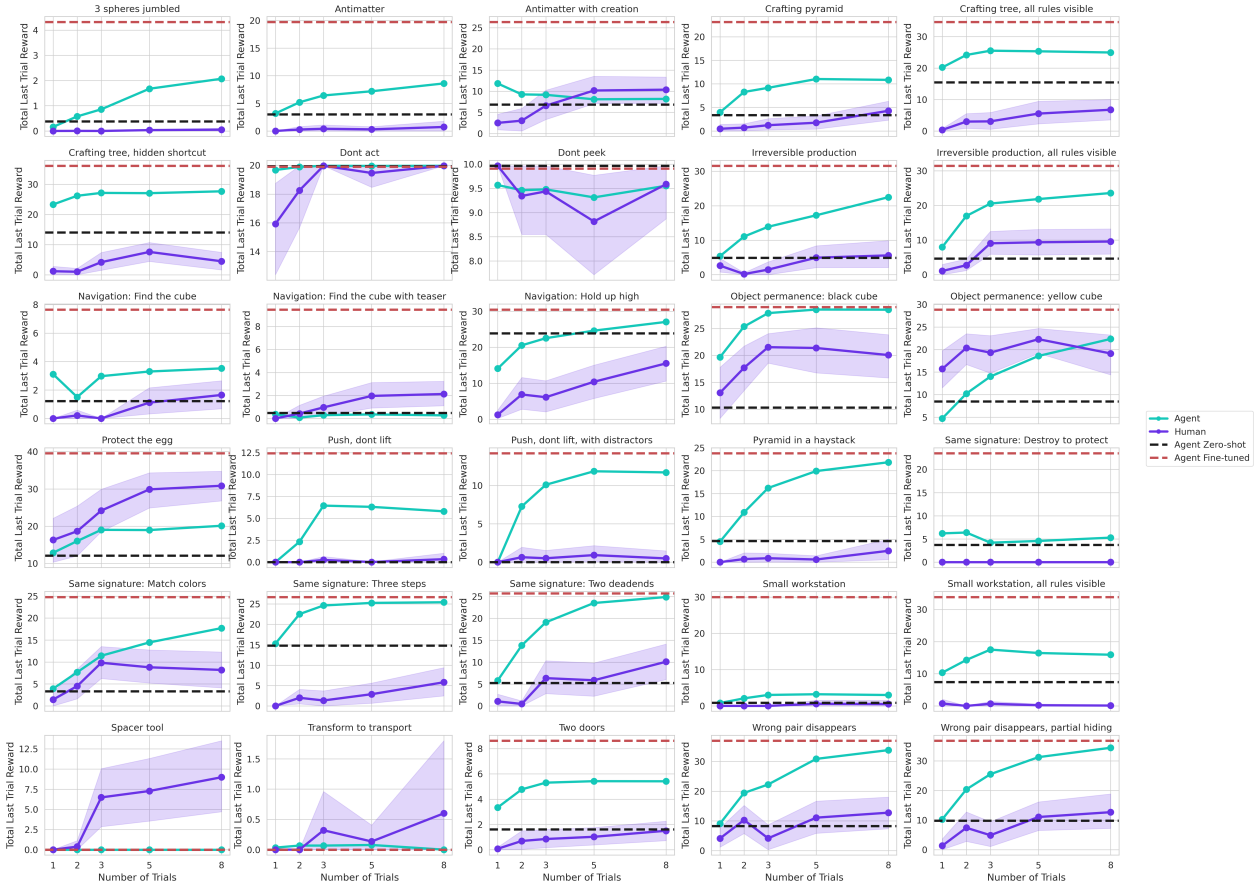
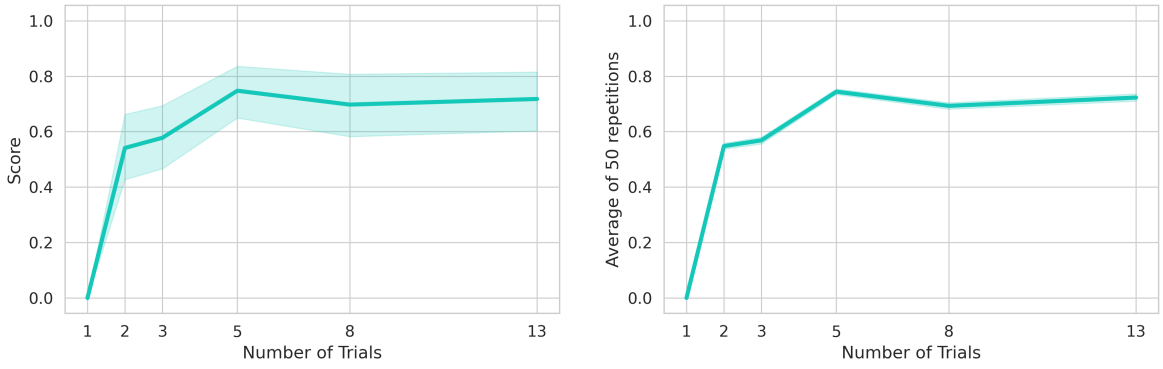


图 F.1 | AdA 在 30 项留出的单智能体手工编写任务中分别与 19 名人类玩家的比较。以黑色虚线表示经过训练、旨在优化零样本性能的基线智能体的性能，这表明 AdA 并未为了实现适应而牺牲零样本泛化。经过微调、以适应手工编写任务的智能体所获奖励也以红色虚线表示，用于粗略展示在每项任务的一次试验中可获得的最高奖励。



(a) (b)

图 F.2 | **(a)** 50 个样本的单次重复得分均值及 95% 置信区间。**(b)** 50 个样本的汇总 50 次重复得分均值及 95% 置信区间。

得分从重复 1 次时最高 0.43 降至重复 50 次时最高 0.06。

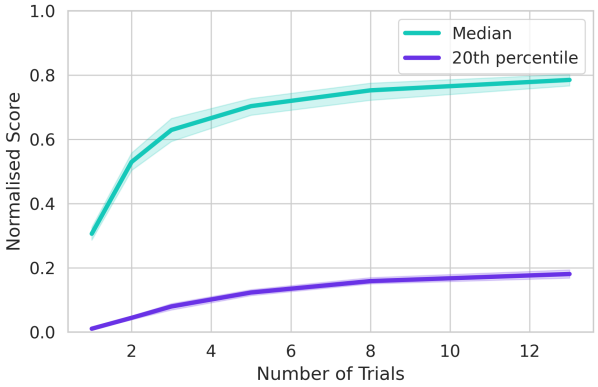


图 F.3 | 95%：在包含 5 个使用不同初始化种子进行训练的智能体的测试集上，中位数和第20百分位数周围的自助法置信区间。观测到的最大标准差，在中位数周围为 0.04，在第20百分位数周围为 0.02。

智能体初始化变异。 在考虑任务变异后，图F.3 展示了训练期间由智能体初始化种子造成的方差。我们绘制了得分随 k 变化的情况，测试集使用的是 AdA 的总计 7600 万参数版本，并采用了 5 个不同的初始化种子。对于中位数，在任意试次数下观测到的最大标准差为 0.04；对于 20th 百分位数，最大标准差为 0.02。由于初始化种子造成的方差较小，我们只用一个初始化种子进行消融实验，以节省计算资源。请注意，我们在消融实验部分展示的结果之间的差异远远超过一个标准差。

F.4. 通过第一人称演示进行提示

图 F.4 展示了在 30 个单智能体手工编写的探测任务中，使用微调智能体进行提示的 AdA 与未提示的基线相比的性能。图中显示了一组 AdA 能够利用提示中的信息，从而获得满分或接近满分的任务。此外，还有一些任务中 AdA 似乎无法做到这一点。除一种情况外，提示都不会损害性能。

对图 F.4 中任务的分析表明，对于较短的导航任务，提示很有用，例如 `Navigation: hold up high`，在这类任务中，智能体沿着一条简短而简单的路径到达目标物体。然而，对于更长、更复杂的导航任务，例如 `Navigation: find the cube`，提示并不能提升性能，这可能是因为完整的示范太长，无法放入智能体的记忆上下文中。

在涉及生成规则的任务中，我们观察到类似的模式。对于解题路径中最多包含 2 条生成规则的任务，例如 `Same signature: match colors`，我们发现未经提示的智能体会探索不同的物体，以确定应触发哪条规则。在获得示范提示后，它会立即触发正确的规则，并取得满分。一个例外是 `Same signature: destroy to protect`，其中一条生成规则涉及摧毁某个物体，而智能体似乎没有记住示范中的这一点。对于使用 3 条或更多生成规则的任务，例如 `Same signature: three steps`（解题路径中有 3 条生成规则）和 `Small workstation`（有 5 条生成规则），智能体往往只记得需要触发的部分规则，并在示范之后继续进行探索行为

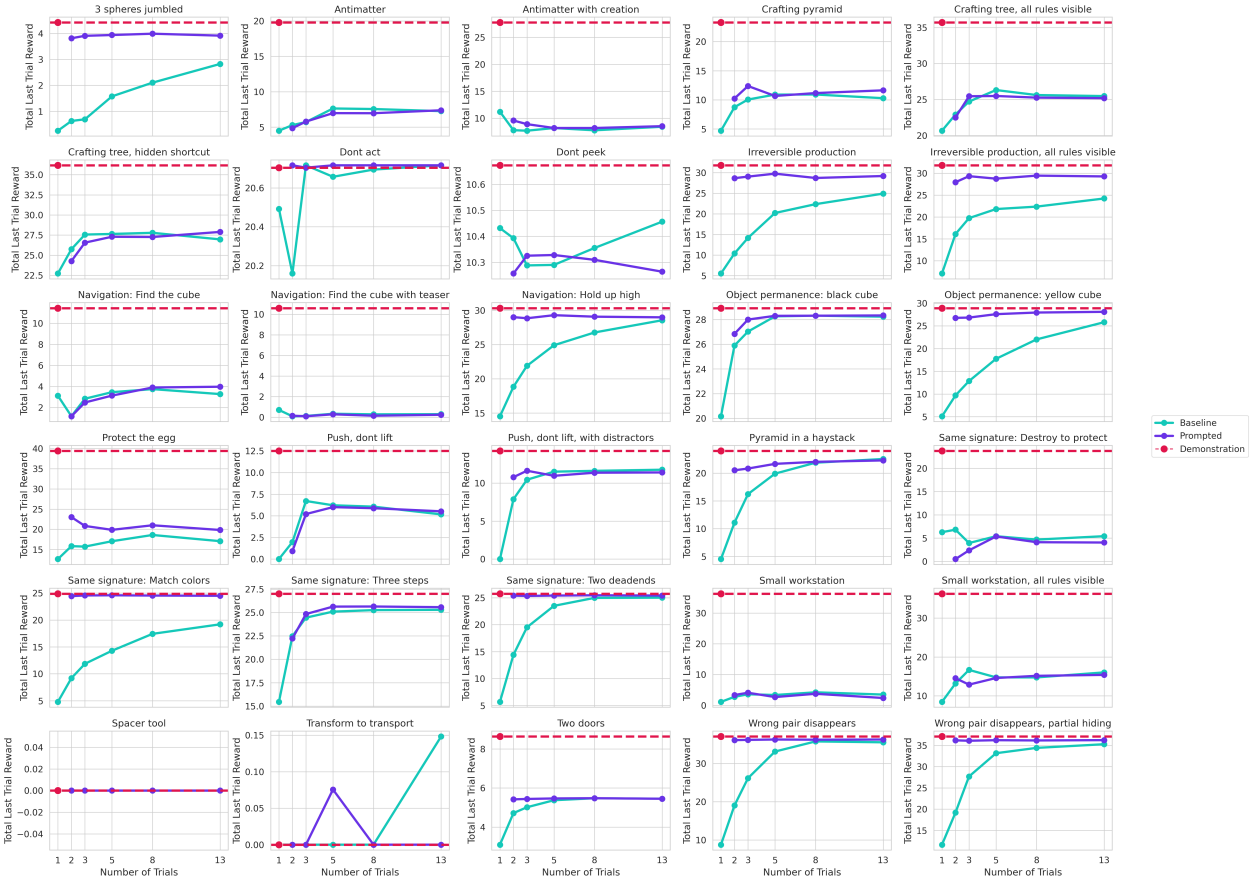


图 F.4 | ：在全部 30 个手工编写的单智能体任务中，对比提示智能体与无提示基线。红色虚线表示教师在第一次试验中向提示智能体提供第一人称演示所获得的得分。

该　　。这些任务上的性能通常与未经提示的基线相当　　e.

另一个似乎会影响提示效果的因素，是世界中物体的拓扑结构和配置，这一点可从 `Small workstation` 任务中看出。虽然教师为这些任务提供的示范呈现了一条清晰的轨迹，但智能体随后会撞上并移开干扰物体，导致出现演示中未曾观察到的环境状态（因此也无法从记忆中回忆出来）。另一方面，良好的物体配置似乎能让智能体更容易从示范中学习，这一点可从 `Irreversible production` 中看出。在这里，触发最后一条生成规则所需的物体彼此靠得很近。尽管该任务的解题路径需要 3 条生成规则，智能体仍表现出最优行为。对于一些更微妙的要求，例如物体的相对位置或工具使用，智能体似乎也无法通过提示推断出来。在 `Antimatter` 等任务中可以观察到这一点：提示后的 AdA 能够触发销毁规则，但我们没有观察到它立即将物体彼此隐藏起来。

提示也可能有助于消除智能体在训练期间形成的偏差。未提示智能体在小型 k 中的得分低于 `Object permanence: black cube`，这一点反映了这一情况。这些任务完全相同，唯一的区别是目标分别为前往右侧的黄色立方体或左侧的黑色立方体。这表明，智能体在训练期间可能形成了偏好黑色立方体或朝左侧物体移动的偏差。提示智能体在 `Object`上的得分显著更高，

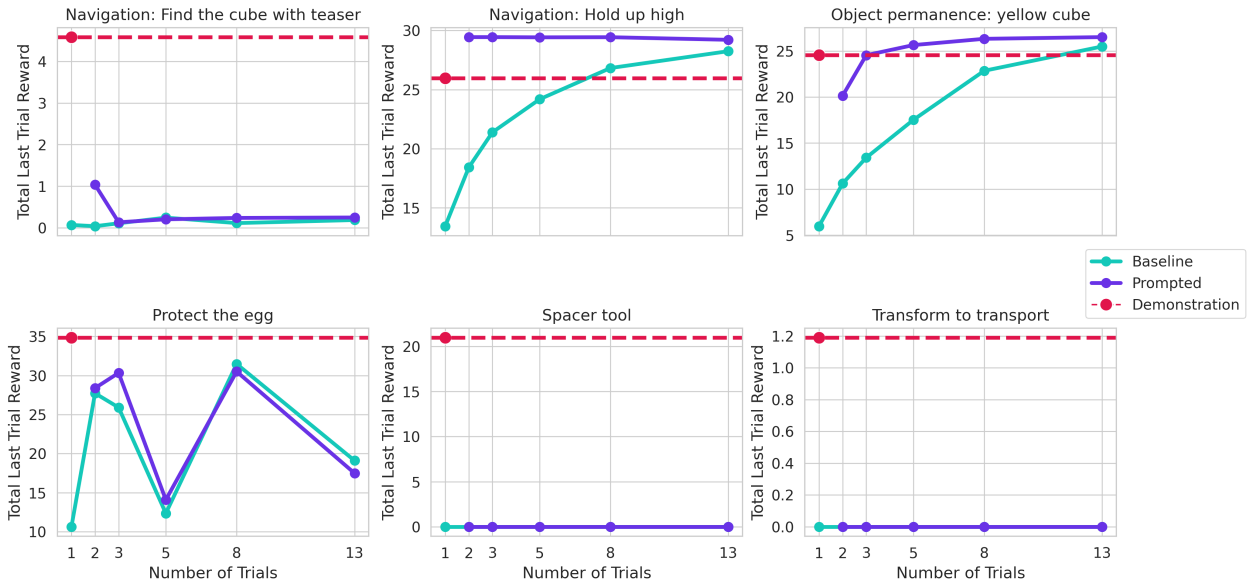


图 F.5 | 通过专家第一人称人类示范提示时，AdA 在 6 个手工编写任务上的性能，与无提示基线相比。

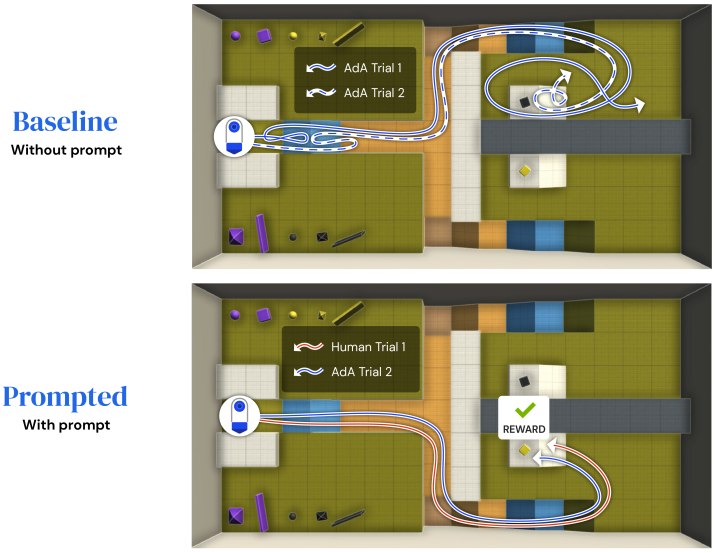


图 F.6 | 俯视图展示了 AdA 在任务 `Object permanence: yellow cube` 中有无人类专家提示时的行为。单独运行时，AdA 似乎倾向于前往黑色立方体，但在此任务中那里是死胡同。通过人类专家轨迹（或微调后的专家轨迹）进行提示后，AdA 能够克服这种偏向，并在第二次试验中前往黄色立方体。

`permanence: yellow cube` 小型 k 的结果表明，提示可能有助于智能体克服这些偏差。

使用人类示范进行提示。我们在少量由人工编写的 6 项任务中，使用专家级人类示范对 AdA 进行提示，如图F.5所示。这些任务经过挑选，涵盖了 AdA 在使用微调教师提示时表现出色、使用微调教师提示时未能成功，以及连微调后的教师也未能完成的任务。

结果呈现出与使用微调后的教师进行提示时相同的模式。在 Navigation: hold up high 和 Object permanence: yellow cube中，经过提示的 AdA 都取得了接近最优的性能，超过了基线和人类示范。图F.6 详细展示了后者的行为。AdA 仍无法从 Navigation: find the cube with teaser 中的示范中学会完成任务，在 Spacer tool 和 Transform totransport中也一样；我们的微调教师也未能完成这些任务。成功的人类示范并未解锁任何 AdA 此前无法展现的能力，这表明这些任务可能与 AdA 的训练任务分布相差过大。使用离策略人类示范进行提示能够取得部分成功，这一点值得注意，也为未来研究开辟了令人振奋的方向。