arXiv:2304.13004v1 [cs.AI] 25 Apr 2023

复杂即时战略游戏中的多智能体强化学习集中式控制

Roger Creus Castanyer Mila
Québec Université de Montréal
roger.creus-castanyer@mila.quebec

摘要

多智能体强化学习（MARL）研究共存于共享环境中的多个学习智能体的行为。与单智能体强化学习相比，MARL 更具挑战性，因为它涉及更复杂的学习动态：每个智能体的观测和奖励都取决于其他所有智能体。在多智能体强化学习的背景下，即时战略游戏代表了极具挑战性的环境，在这类环境中，多个玩家同时交互，并同时控制许多不同性质的单位。事实上，即时战略游戏对当前的强化学习方法而言极具挑战性，仅仅能够使用强化学习处理这类游戏本身就很有意义。本项目提供了在 *Lux AIv2*Kaggle 竞赛中应用强化学习的端到端实践经验。在该竞赛中，参赛者需要设计智能体来控制可变规模舰队，并在一对一场景中与其他参赛者对抗，同时处理多变量优化、资源采集和分配问题。我们采用集中式方法训练强化学习智能体，并报告了整个过程中的多项设计决策。我们提供了项目的源代码：
https://github.com/ roger-creus/centralized-control-lux

1 引言

强化学习（RL）已在许多复杂领域取得巨大成功，例如电子游戏和机器人学 [Mnih等人， 2013]。然而，强化学习领域的大多数研究都假设采用单智能体设置，即智能体与静态或预定环境进行交互。近年来，人们对多智能体强化学习（MARL）的兴趣日益浓厚。在多智能体强化学习中，多个智能体与共享环境进行交互，并学习进行协调或竞争。近期许多最具影响力的强化学习成果都采用了多智能体强化学习。例如，OpenAI Five [Berner等人， 2019], 由五个人工智能智能体组成的团队，在复杂的电子游戏 Dota 2 中击败了一支职业人类玩家队伍。同样，AlphaStar [Vinyals等人， 2019], 这一由 DeepMind 开发的系统，在星际争霸 II 游戏中击败了职业人类玩家。AlphaGo [Silver等人， 2016], DeepMind 开发的另一系统，在围棋这一棋盘游戏中击败了人类世界冠军。最近，Meta AI 发布了 Cicero [FAIR], , 这是一个在外交游戏中表现优于人类的人工智能智能体。这些成功展示了多智能体强化学习的潜力，也凸显了在复杂环境中制定有效策略、实现智能体之间协调与合作的重要性。

由于训练此类复杂系统需要较高的计算需求，大多数研究机构无法为多智能体强化学习提供大量实证证据。然而，近年来在设计经济实惠但复杂的环境方面取得了许多进展，这些环境对 StarCraft 和 Dota II 提出了类似的挑战 [Huang et al., 2021, Chen et al., 2023, Suarez et al., 2019]。具体而言，这些挑战体现在单位之间的协调、联合动作空间的规模、非平稳动态以及非马尔可夫奖励方面，从而使多智能体强化学习研究群体能够通过实证研究新的技术和算法。近期已经举办了多项竞争活动

预印本。审稿中。

在此类经济实惠的环境中开展研究，是因为由于多智能体强化学习和即时战略游戏的隐含复杂性，使用强化学习解决这些问题仍然具有挑战性。在本文中，我们参加了 *Lux AI v-2*Kaggle 竞赛[1]，并提供了从定义环境观测和动作空间，到通过自博弈训练强化学习智能体的端到端经验。*Lux* 环境如图 1 所示。附录 A 和环境规范[2] 提供了有关 Lux 的更多详细信息。

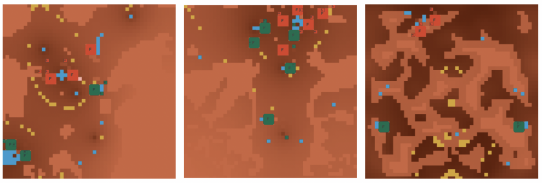

图1：以下是 *Lux AI v2*环境中的三张不同的48×48地图。 玩家1和玩家2在伪随机生成的地图上争夺资源。黄色单元格表示矿石，蓝色单元格表示冰，棕色单元格表示卢布，带有字母F的3×3方格表示工厂，带有字母L和H的小方格分别表示轻型和重型机器人。

Lux环境提出了几个关键挑战：（*i*）Lux提供了实体观测，在将其输入强化学习智能体之前需要进行预处理（例如，参见此处的观测示例）；（*ii*）在每个回合中，每位玩家都需要为可变规模舰队中的单位发出原始动作；（*iii*）这些单位具有不同的动作空间和规格；（*iv*）资源稀缺且对生存至关重要；（*v*）地图和地形存在高度差异，每场游戏中的工厂数量也各不相同；最后，（*vi*）由于涉及另一个智能体，这是一个非平稳环境。可以说，这些工程、算法和理论方面的挑战共同构成了一个极其复杂的强化学习环境。在本文中，我们逐一应对这些挑战，并报告我们的实证经验。

2 背景

在多智能体强化学习的背景下，Lux环境可以建模为一个多智能体马尔可夫决策过程（MMDP）。MMDP [Boutilier, 1996, van der Pol等人，2021] 是一个元组（$\mathcal{N}$，$\mathcal{S}$，$\mathcal{A}$，$\mathcal{T}$，$\mathcal{R}$），其中 $\mathcal{N}$是由 m 个智能体组成的集合，$\mathcal{S}$是状态空间，$\mathcal{A} = \mathcal{A}_1 \times \ldots \times \mathcal{A}_m$是MMDP的联合动作空间，$\mathcal{T}$: $\mathcal{S} \times \mathcal{A}_1 \times \ldots \times \mathcal{A}_m \times \mathcal{S} \to [0, 1]$是转移函数，$\mathcal{R}$：$\mathcal{S} \times \mathcal{A}_1 \times \ldots \times \mathcal{A}_m \to \mathcal{R}$是即时奖励函数。这种分散式设置对同一队伍中$m$个智能体之间的协调进行建模。$\mathcal{S}$是全局状态空间，转移函数则根据m维联合动作分布，对未来全局状态的概率进行建模。同样，该框架将全局奖励分布定义为单个全局状态和m 个联合动作的函数，这意味着m 个智能体协作以最大化单一奖励函数。此外，在Lux这类1对1环境中，对手单位的动作属于状态空间 $\mathcal{S}$；若对手是学习型智能体或会随时间演化，则转移分布 $\mathcal{T}$是非平稳的。

分散式多智能体强化学习中的信用分配颇具挑战性 [Chang et al., 2003]。因此，一些研究探索了价值分解方法 [Sunehag et al., 2017]，以学习更简单的智能体特定价值函数或评论家 [Iqbal and Sha, 2019]。另一些研究则利用手工设计的智能体特定奖励分布，独立训练每个智能体 [Ye et al., 2020]。然而，这些研究假设智能体数量固定，因此无法适用于 Lux 这类单位不断被摧毁和建造的大规模环境。最后，一些研究定义了集中式设置，并采用像素到像素的架构，在网格状环境中利用多智能体控制 [Huang et al., 2021, Chen et al., 2023, Han et al., 2019]。在此背景下，多智能体马尔可夫决策过程转变为标准的单智能体问题。然而，动作空间的规模、单位间协作以及转移函数的非平稳性仍然构成挑战。在本工作中，我们同样采用集中式方法训练强化学习智能体，并使用近端策略优化（PPO）对其进行优化 [Schulman et al., 2017]。PPO 是一种基于演员-评论家策略梯度的算法。与传统的 REINFORCE 或 A2C 相比 [Grondman et al., 2012],，它定义了一个替代目标，使收集到的同策略数据能够用于多次经过修正的训练更新。在第3节中，我们将详细介绍模型架构和 PPO训练过程。

[1] https://github.com/Lux-AI-Challenge/Lux-Design-S2[2]
https://www.lux-ai.org/specs-s2

3 方法论

本节定义了能够在 Lux环境中训练强化学习智能体的组件。主要要求包括定义状态空间和动作空间、稠密奖励分布、功能完备的模型架构以及训练流程。尽管这些要求存在于所有强化学习应用中，但随着环境复杂度的提升，其复杂性也会随之增加，从而使强化学习在即时战略游戏中面临挑战。下面将介绍我们的设计决策，其目标是将 Lux 转变为一个全观测马尔可夫决策过程，从而能够训练单个、也因此是集中式的强化学习智能体。

观测空间。我们通过提取若干包含游戏独立信息的特征图来定义观测。这些特征图堆叠在一起，形成大小为 $(H \times W \times C)$ 的对象，其中 $H = 48, W = 48$ 是固定地图大小，在我们的例子中为 $C = 17$。这些特征图提供了地图上友方和敌方单位、工厂及资源的位置信息。表 1 详细描述了观测空间。

动作空间。我们定义了两种动作空间，一种用于工厂（可以生产机器人并培育地衣），另一种用于机器人（可以移动、挖掘和转移资源）。工厂的动作空间是多离散的，包含 2 个组件：第一个表示工厂在地图上的位置，第二个允许执行 4 种动作。单位的动作空间是多离散的，包含 6 个组件：第一个表示单位在地图上的位置，其余 5 个定义联合动作（例如：转移 - 冰 - 南）。附录 A 详细描述了动作空间。此外，我们还实现了无效动作掩码 [Huang和Ontañón, 2020]，以防止智能体从游戏中不会执行的输出动作中学习——模型会在 48x48 地图的每个单元格输出一个动作，但智能体不应从不会下发给实际单位的动作中学习。动作掩码用于将非法动作的对数几率有效地设为较大的负值，使其对数概率为 0，梯度也为 0。

奖励分布。我们使用奖励塑形来定义一种手工设计的奖励分布，以鼓励玩家产水。虽然玩家通过在 1000 个游戏步骤内尽可能多地培育地衣来获胜，但在 Lux 中，资源需求至关重要，因此不能让强化学习智能体仅凭获胜奖励学会产水。为了产水——这是生存和培育地衣所必需的——单位必须高效导航、挖掘冰、将其转移到工厂并进行精炼。可以说，后者是一系列过于复杂的动作，难以通过稀疏或延迟的奖励来发现。关于奖励分布的更多详细讨论见第 D 节。

自博弈。我们实现了优先级虚拟自博弈（PFSP） [Vinyals等人， 2019] ，为强化学习智能体提供稳定的训练流程。与基于种群的方法 [Jaderberg等人， 2017], 相比，在 PFSP 中，单个学习智能体会与自身过去的检查点池进行对战。该池以固定大小的队列实现，因此每经过 N 次训练更新，就会存入一个新检查点并移除最旧的检查点。训练期间，抽取最新检查点的概率为 50%，否则会随机抽取其他检查点。第 E 节将更详细地讨论自博弈。

架构。本文采用了一种演员-评论家像素到像素架构 [Han et al., 2019]，与 *Gridnet* [C hen et al., 2023], 类似，并使用 PPO 进行训练。模型架构如图 2 所示。在训练过程中，模型通过对所有有效的机器人动作和工厂动作的对数几率求和，计算联合动作概率。后者对应于概率空间中的乘法，并假设工厂动作与机器人动作之间相互独立。评论家用于计算回报和优势，而 PPO 更新则使演员预测的、具有更高估计优势的动作变得更加可能。

4 实验

一般来说，强化学习需要大量数据才能发挥作用，而多智能体强化学习的样本效率通常会低几个数量级 [Garnelo等，2021，Berner等，2019，Vinyals等人， 2019]。此外，Lux环境运行缓慢，在单块 A6000 GPU上运行时，每秒只能达到60个环境步数。我们进行了超参数搜索[3]，并行运行多个任务，每个任务训练智能体1000万环境步数，总计耗费1,296小时的计算时间。之后，我们选取表现最佳的配置，使用8块RTX8000 GPU并行运行124小时，共训练2.2亿环境步数。报告……更多详情请参见F节。

[3] https://api.wandb.ai/links/rogercreus/49p dlt7a

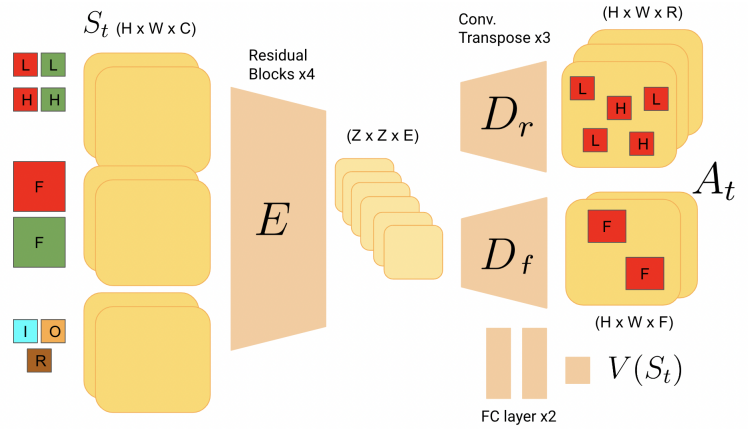

图 2：我们的像素到像素架构。我们向模型输入一组包含单位、工厂和资源信息的 C 个特征图。我们使用残差网络 [He 等人， 2016]，并通过挤压与激励层 [Hu 等人， 2018] 将观测编码为低维表示。我们将这些表示输入三个不同的头部：评论家输出一个表示状态价值的单一标量值；机器人行动者 输出与机器人动作组件维度数量相同的特征图；工厂行动者 输出数量与工厂动作组件相同的动作。

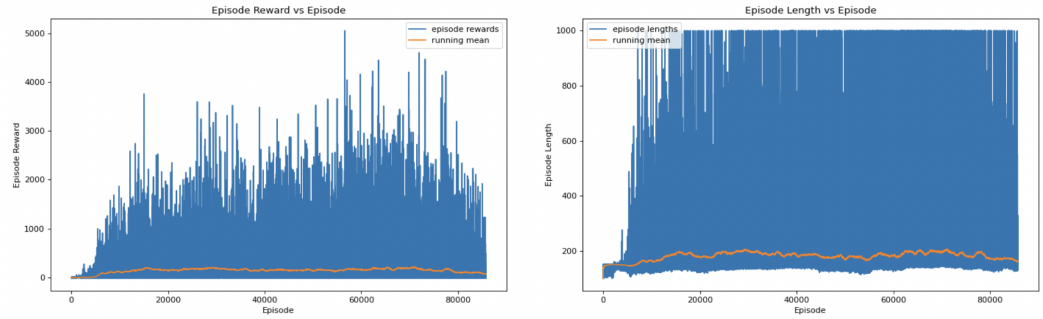

图 3：该图展示了训练期间的回合奖励（左）和回合长度（右）。智能体经过 220M 次环境交互训练。尽管智能体在许多回合中学会了收集资源，并在游戏的 1,000 个步骤中存活下来，但其表现存在较大程度的方差。由于每个回合都有显著不同的地图和数量不固定的工厂，智能体很难实现泛化。

5 结论与未来工作

本研究为多智能体强化学习定义了一个集中式训练框架，并展示了如何在极具挑战性的强化学习环境中，使用PPO训练像素到像素的演员-评论家架构。我们发现了在即时战略游戏中应用强化学习面临的若干挑战，阐明了我们的设计决策，并讨论了可能的改进方向。我们还提供了一个开源代码库，可作为未来类似竞赛的起点 [4]，并分享了所有实验的日志[5]。有多个研究方向可以改进所提出的方法，其中包括：（*i*）更好地定义观测空间和动作空间；（*ii*）使用归纳偏置更适合网格状即时战略游戏的其他架构（例如，避免下采样瓶颈 [Chen 等，2023],U-Net [Ronneberger 等， 2015],LSTM [Hochreiter 和 Schmidhuber， 1997]）；（*iii*）实施基于群体的训练，以塑造有利于学习智能体的学习动态 [Garnelo 等， 2021]；以及（*iv*）扩展架构规模并实现高度分布式训练。

[4]https://github.com/roger-creus/centralized-control-lux[5]
https://api.wandb.ai/links/rogercreus/49pdlt7a

参考文献

Christopher Berner、Greg Brockman、Brooke Chan、Vicki Cheung、Przemysław D. ebiak、Christy Dennison、David Farhi、Quirin Fischer、Shariq Hashme、Chris Hesse 等。通过大规模深度强化学习实现 Dota 2。*arXiv* 预印本 *arXiv:1912.06680*，2019 年。Craig Boutilier。在多智能体决策过程中进行规划、学习与协调。载于*TARK*，第 96 卷，第 195–210 页。Citeseer，1996 年。Yu-Han Chang、Tracey Ho 和 Leslie Kaelbling。所有学习都是局部的：全局奖励博弈中的多智能体学习。神经信息处理系统进展，第 16 卷，2003 年。Hanmo Chen、Stone Tao、Jiaxin Chen、Weihan Shen、Xihui Li、Sikai Cheng、Xiaolong Zhu 和 Xiu Li。大规模智能体合作与竞争中涌现的集体智能。*arXiv* 预印本 *arXiv:2301.01609*，2023 年。Meta 基础人工智能研究外交团队（FAIR）†、Anton Bakhtin、Noam Brown、Emily Dinan、Gabriele Farina、Colin Flaherty、Daniel Fried、Andre w Goff、Jonathan Gray、Hengyuan Hu 等。通过结合语言模型与战略推理，在外交游戏中实现人类水平的表现。《科学》，378(6624)：1067–1074，2022 年。

Marta Garnelo、Wojciech Marian Czarnecki、Siqi Liu、Dhruva Tirumala、Junhyuk Oh、Gauthier Gidel、Hado van Hasselt 和 David Balduzzi。选择你的战斗：将交互图作为群体层面目标，以实现战略多样性。*arXiv* 预印本 *arXiv:2110.04041*，2021 年。

Thore Graepel、Tom Minka 和 R TrueSkill Herbrich。贝叶斯技能评级系统。神经信息处理系统进展，19:569–576，2007 年。

Ivo Grondman、Lucian Busoniu、Gabriel AD Lopes 和 Robert Babuska。演员-评论家强化学习综述：标准策略梯度与自然策略梯度。*IEEE*《系统、人类与控制论汇刊》*C* 部分（应用与评论），42(6):1291–1307，2012 年。

Lei Han、Peng Sun、Yali Du、Jiechao Xiong、Qing Wang、Xinghai Sun、Han Liu 和 Tong Zhang。电子游戏人工智能中多智能体强化学习的网格级控制。载于国际机器学习会议，第 2576–2585 页。PMLR，2019 年。

Kaiming He、Xiangyu Zhang、Shaoqing Ren 和 Jian Sun。用于图像识别的深度残差学习。载于*IEEE*计算机视觉与模式识别会议论文集，第 770–778 页，2016 年。

Sepp Hochreiter 和 Jürgen Schmidhuber。长短期记忆。神经计算，9(8)：1735–1780，1997。Jie Hu、Li Shen 和 Gang Sun。挤压与激励网络。载于 *IEEE*计算机视觉与模式识别会议论文集，第 7132–7141 页，2018 年。Shengyi Huang 和 Santiago Ontañón。深入研究策略梯度算法中的无效动作掩码。*arXiv* 预印本 *arXiv:2006.14171*，2020 年。Shengyi Huang、Santiago Ontañón、Chris Bamford 和 Lukasz Grela。Gym-μrts：面向使用深度强化学习开展经济实惠的完整即时战略游戏研究。载于 *2021 IEEE*游戏会议（*CoG*），第 1–8 页。IEEE，2021 年。Shariq Iqbal 和 Fei Sha。用于多智能体强化学习的演员-注意力-评论家。载于 机器学习国际会议，第 2961–2970 页。PMLR，2019 年。Max Jaderberg、Valentin Dalibard、Simon Osindero、Wojciech M Czarnecki、J eff Donahue、Ali Razavi、Oriol Vinyals、Tim Green、Iain Dunning、Karen Simonyan 等。神经网络的基于种群的训练。*arXiv* 预印本 *arXiv:1711.09846*，2017 年。Volodymyr Mnih、Koray Kavukcuoglu、David Silver、Alex Graves、Ioannis Antonoglou、Daan Wierstra 和 Martin Riedmiller。使用深度强化学习玩 Atari 游戏。*arXiv* 预印本 *arXiv:1312.5602*，2013 年。

Olaf Ronneberger、Philipp Fischer 和 Thomas Brox。U-Net：用于生物医学图像分割的卷积网络。载于 医学图像计算与计算机辅助干预–*MICCAI 2015*：第 *18* 届国际会议，慕尼黑，德国，*2015*年*10*月*5*日至*9*日，论文集，第三部分 *18*，第 234–241 页。Springer，2015 年。

John Schulman、Filip Wolski、Prafulla Dhariwal、Alec Radford 和 Oleg Klimov。近端策略优化算法。*arXiv* 预印本 *arXiv:1707.06347*，2017 年。

David Silver、Aja Huang、Chris J Maddison、Arthur Guez、Laurent Sifre、George Van Den Driessche、Julian Schrittwieser、Ioannis Antonoglou、Veda Panneershelvam、Marc Lanctot 等。借助深度神经网络和树搜索掌握围棋游戏。《自然》，529(7587)：484–489，2016 年。

Joseph Suarez、Yilun Du、Phillip Isola 和 Igor Mordatch。Neural MMO：用于训练和评估智能体的大规模多智能体游戏环境。*arXiv* 预印本 *arXiv:1903.00784*，2019 年。

Peter Sunehag、Guy Lever、Audrunas Gruslys、Wojciech Marian Czarnecki、Vinicius Zambaldi、Max Jaderberg、Marc Lanctot、Nicolas Sonnerat、Joel Z Leibo、Karl Tuyls 等。用于协作式多智能体学习的值分解网络。*arXiv* 预印本 *arXiv:1706.05296*，2017 年。

Elise van der Pol、Herke van Hoof、Frans A Oliehoek 和 Max Welling。多智能体 MDP 同态网络。*arXiv* 预印本 *arXiv:2110.04495*，2021 年。

Oriol Vinyals、Igor Babuschkin、Wojciech M Czarnecki、Michaël Mathieu、Andrew Dudzik、Junyoung Chung、David H Choi、Richard Powell、Timo Ewalds、Petko Georgiev 等。使用多智能体强化学习达到星际争霸 II 特级大师水平。《自然》，575(7782):350–354，2019 年。

Deheng Ye、Guibin Chen、Wen Zhang、Sheng Chen、Bo Yuan、Bo Liu、Jia Chen、Zhao Liu、Fuhao Qiu、Hongsheng Yu 等。利用深度强化学习实现完整 MOBA 游戏。神经信息处理系统进展，33:621–632，2020 年。

附录A TheLux-v2 竞争

在 Lux 中，人工智能智能体和脚本机器人都在 1对1游戏中争夺资源。游戏分为 3 个阶段：（*i*）起始阶段，双方玩家竞价，决定谁先部署工厂；（*ii*）部署阶段，由竞价获胜的玩家先开始，双方轮流在选定的位置部署工厂；以及（*iii*）正常阶段，双方玩家向所有工厂和机器人下达动作，以收集资源并培育地衣。在这项工作中，我们通过启发式方法将双方玩家在竞价阶段 的出价设为 0，并采用一种部署阶段 策略，根据冰、矿石、碎石和敌方工厂的位置来部署工厂。这样，在我们的环境封装器中，每次调用*env.reset()* 都会预先模拟竞价和部署阶段，并返回正常阶段的初始状态。

游戏最多持续 1,000 回合，期间两名玩家同时提交动作。如果一名玩家失去所有工厂，游戏即告结束。否则，地图上地衣较多的玩家获胜。

工厂可以建造轻型或重型机器人（消耗金属和电力），也可以培育地衣（消耗水）。机器人可以向 4 个方向移动（消耗电力），向 4 个方向转移资源（冰、矿石、水、金属和电力），挖掘资源（冰、矿石、碎石），以及从工厂拾取资源或自毁。此外，机器人会按照动作队列行动，这使脚本代理能够在单个步骤中提交完整的动作计划。更新机器人的动作队列会额外消耗电力，从而鼓励对动作序列进行高效规划。在本文中，我们假设使用单个动作，并支付额外的电力成本。

附录 B 观测空间

Lux 最初提供实体观测，这些观测是包含当前游戏状态全部信息的字典（即 JSON 文件）。因此，我们将这些信息解析为可输入深度强化学习模型的格式，并在下文对其进行描述：

表 1：构成输入给智能体的观测的 17 个特征图的描述。负值始终表示与对手相关的属性（例如，unit_ice 中的 -200 表示敌方单位的货物中有 200 单位的冰。每个特征图的大小都与地图相同，为 48x48。

Name	描述	范围	归一化
is_day	当前回合是白天还是夜晚	[0, 1]	-
碎石	每个单元格中的碎石数量	[0, 100]	100
ore	每个单元格中是否有矿石	[0, 1]	-
ice	每个单元格中是否有冰	[0, 1]	-
地衣	每个单元格中的地衣数量	[0, 100]	100
是资源_	每个单元格中是否有矿石或冰	[0, 1]	-
轻型_单位	每个单元格中是否有轻型机器人	[-1, 0, 1]	-
重型_单位	每个单元格中是否有重型机器人	[-1, 0, 1]	-
单位_冰_	机器人货舱中的冰量	[-3000, 3000]	3,000
单位矿石_	机器人货舱中的矿石数量	[-3000, 3000]	3,000
单位_电力	机器人电池中的电力数量	[-1000, 1000]	1,000
单位_在_工厂上	每个机器人是否位于工厂上方	[-1, 0, 1]	-
工厂	每个格子上是否有工厂	[-1, 0, 1]	-
工厂_冰	每个工厂中的冰量	$[-\infty, \infty]$	150
工厂_矿石	每个工厂中的矿石数量	$[-\infty, \infty]$	150
工厂_水	每个工厂中的水量	$[-\infty, \infty]$	150
工厂_金属	每个工厂中的金属数量	$[-\infty, \infty]$	150

附录 C 动作空间

我们将机器人和工厂的联合动作空间定义为一个字典（*gym.spaces.Dict()*），其中工厂的动作_空间*["factories"]* 和机器人动作_空间*["robots"]* 都是多离散对象。多离散空间是简单离散空间 [6] 的多维扩展。在多离散动作空间中，每个组件（即维度）彼此独立，并且可以具有不同的大小。两个多离散动作空间的详细描述见表 2 和表 3。

表 2：机器人动作空间的描述

组件	描述	范围
源单位	执行动作的选定单位的位置	$[0, h \times w - 1]$
动作类型	无操作、移动、转移、拾取、挖掘、自毁	[0, 5]
移动参数	上、右、下、左	[0, 3]
转移参数	中心、上、右、下、左	[0, 4]
转移数量	25%、50%、75%、95%	[0, 3]
转移资源	冰、矿石、电力	[0, 2]

表 2 展示了机器人的动作空间。通过这种方式，可以从每个组件中采样一个值来获得一个动作。不过请注意，第一个组件，即动作类型，决定了将使用哪些动作参数。例如，动作移动 - 右 - 左 - *25%* - 矿石 将执行移动 - 右 ，而动作转移 - 左 - 上 - *50%* - 冰将执行转移 - 上 - *50%* - 冰。拾取、挖掘 和自毁 这些动作不需要其他参数，因为机器人只能拾取固定数量的电力；挖掘仅表示机器人会挖掘其正在挖掘的格子中的任何资源，而自毁的含义不言自明。

[6] https:/ /www.gymlibrary.dev/api/spaces/#multidisc rete

表3：工厂的动作空间描述

组件	描述	范围
源单位	选定单位执行动作的位置	$[0, h \times w - 1]$
动作类型	无操作，建造_轻型地衣，建造_重型地衣，生长_地衣	[0, 3]

附录D 奖励分布

强化学习智能体必须掌握多项技能才有机会在 Lux 中获胜，其中最具挑战性的是高效地产生水：每座工厂初始拥有 150 单位的水和金属，并且每回合消耗 1 单位水。工厂一旦耗尽水就会损失，并且只能通过派遣机器人挖掘冰，再将冰转移到工厂进行精炼来产生水。如果强化学习智能体无法掌握这一连串动作，而这需要大量高效的探索，那么训练期间的游戏将持续 150 回合，此时双方玩家都会失去所有工厂。解决这一挑战一直是我们工作的主要阻碍，因此，我们最初的稠密奖励只奖励挖掘冰的机器人，以及精炼冰的工厂。理想情况下，在强化学习智能体掌握这一策略后，它们还需要学习游戏后期动态，在此过程中必须消耗水来生长地衣，同时避免水耗尽。因此，我们提议在经过大量训练更新后改变奖励分布，改为一种稀疏奖励，用于表示游戏最后一回合的地衣数量。后者可以为强化学习智能体建立课程学习，使其掌握基本技能，并将这些技能转移到游戏后期动态中。

附录E 自博弈

自博弈有助于塑造有利于学习者智能体的学习动态。多智能体强化学习领域中有一个完整的子领域，专门探索训练多样化智能体群体的不同策略；这些策略有助于训练最先进的智能体，使其能够进行复杂的电子游戏。基于种群的训练[Garnelo等人， 2021]旨在定义一种指标，用于评估种群中智能体的真实技能，并制定策略，在每场游戏中为池中的每个智能体抽样对手。我们认为，基于种群的训练是近期多智能体强化学习领域最具影响力的工作中的关键特征 [Vinyals等，2019，Berner等， 2019]。然而，这一方法通常被忽视，有时也显得晦涩难懂，因为目前普遍缺乏用于实现此类技术的框架和资源。

在这项工作中，我们还使用TrueSkill [Graepel等人， 2007] 指标对智能体检查点池进行排名。我们没有将该池用作队列并不断移除最旧的检查点，而是使用这种指标来公平评估那些原本要被移除的旧检查点是否确实比其他检查点的技能更低。不过，由于依赖外部库 [7]，我们无法在所使用的并行作业之间扩展排名函数，因此在参数扫描实验中没有使用这一指标。

附录 F 实验

图 F 展示了最终 PPO 运行的完整日志。图 2 中的架构使用 PPO 进行了训练，运行了 2.2 亿环境步数，使用 8 个 Nvidia A600 GPU 耗时 126 小时。图 3 表明：

- **价值损失在训练过程中增加**。价值损失出现峰值通常是一个好迹象，因为这意味着智能体观察到了一些状态转移，而它起初无法正确估计其价值。后者表明智能体正在探索（可能是）高价值行为。不过，在收敛时，我们预计价值损失会逐渐接近 0。
- **熵在开始时增加，并在训练过程中降低**。这通常是一个好迹象，意味着策略会收敛为随机性更低的行为。
- **策略损失在训练过程中逐渐接近 0**。这通常是一个好迹象，并且与熵的降低相关。这意味着策略在更新过程中变化得不那么突然，因此会收敛为随机性更低的策略。

[7] https://trueskill.org/

• **KL散度接近于 0，但会突然出现尖峰**。这些突然出现的尖峰通常是不理想的，表明策略采取了较大的学习步长，这可能导致遗忘之前学到的行为。理想情况下，我们希望 KL 在训练期间保持相近，并接近于 0。

