大规模智能体协作与竞争中涌现的集体智能

Hanmo Chen[1], [*], [‡]、**Stone Tao**[2], [*]、**Jiaxin Chen**[3], [†]、**Weihan Shen**[3],
Xihui Li[1], [‡]、**Chenghui Yu**[1], [‡]、**Sikai Cheng**[4], [‡]、**Xiaolong Zhu**[3],**Xiu Li**[1]
[1]清华大学、深圳国际研究生院， [2]加利福尼亚大学圣迭戈分校[3]Parametrix.ai， [4]佐治亚理工学院
{chm20,xh-li21,ych20}@mails.tsinghua.edu.cn, stao@ucsd.edu{
jiaxinchen,weihanshen,xiaolongzhu}@chaocanshu.ai scheng326@gatech.edu.cn,
li.xiu@sz.tsinghua.edu.cn

arXiv:2301.01609v2 [cs.AI] 5 Jan 2023

摘要

受到地球上不同种群通过协作与竞争不断演化的生物体启发，我们通过大规模智能体强化学习研究人工集体智能的涌现。为此，我们提出了一种新的大规模智能体强化学习环境 Lux，其中两个队伍中的大量动态智能体争夺有限资源并抵御黑暗。在 Lux 中，我们在课程学习阶段采用标准强化学习算法训练智能体，并借助像素到像素策略网络实现集中式控制。随着智能体通过自博弈共同进化，我们观察到智能经历了多个阶段，从习得原子技能到发展出群体策略。由于这些习得的群体策略源自个体决策，而非显式的协调机制，我们认为人工集体智能是由大规模智能体协作与竞争涌现而来的。我们还通过指标和消融研究进一步分析各种习得策略的涌现，旨在为大规模智能体环境中的强化学习实现提供见解。

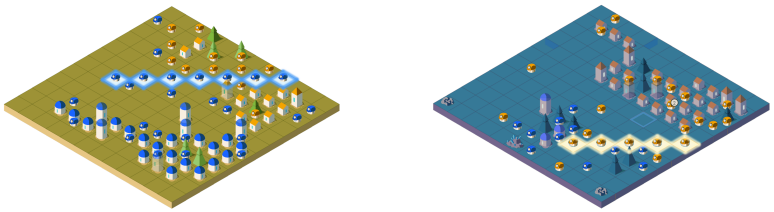
1 引言

复杂的群体行为和社会行为广泛存在于地球上的人类和动物之中。在浩瀚的生态系统中，种群之间的协作与竞争，以及不断变化的环境，共同构成生物体大规模协同进化的自然驱动力（Wolpert & Tumer, 1999年；Dawkins & Krebs, 1979）。种群之间的大规模协同进化催生了能够完成个体无法胜任任务的群体策略（Ha & Tang, 2022）。受到自然界这一自组织机制的启发，即集体智能源于大规模智能体之间的协作与竞争，我们提出通过在大规模智能体环境中训练强化学习智能体，来模拟集体智能的涌现。我们希望这能成为大规模智能体强化学习研究的垫脚石，并为复杂的大规模智能体问题提供一种启发性方法。

多智能体强化学习（MARL）的最新进展展现了其通过多智能体协作完成复杂任务的潜力，例如进行星际争霸 II（Vinyals 等人，2019）和 DOTA2（Berner 等人，2019）。然而，在这些场景中，智能体数量仍仅有几十个，与自然界的种群规模相去甚远。为了支持大规模的多智能体协作与竞争，我们将大规模智能体场景重新引入多智能体强化学习。为此，我们提出了 Lux 这一协作与竞争环境，其中两个种群的数百个智能体争夺有限资源，并抵御黑暗。我们认为，Lux 是一个适合开展集体智能实验的测试平台，因为它提供了一个开放环境，让数百个智能体能够协作、竞争并不断进化。

[*]共同贡献。
[†]通讯作者。
[‡]在 Parametrix.ai 担任研究实习生期间完成的工作。

从算法角度来看，大规模智能体场景给强化学习算法带来了巨大挑战，因为信用分配问题变得愈发棘手。一些研究（Lowe 等人，2017年）着眼于多智能体之间的信用分配问题，但缺乏扩展至大规模智能体场景的能力。为解决这一问题，我们提出了一种用于 Lux 的集中式控制方案，将像素到像素建模架构（Han 等人，2019年）与近端策略优化（PPO）（Schulman 等人，2017年）算法相结合。借助该方案，我们避免了信用分配问题；对阵 Toad Brigade 队伍（TB）提出的最先进策略（Isaiah 等人，2021）的胜率最高可达 90%。该队伍曾在 Kaggle 上的 Lux AI 竞赛中获得第一名[1]。
通过自博弈和课程学习阶段，我们观察到大规模智能体协同进化经历了多个阶段，从移动和建造等原子技能，到高效领土占领和长期资源管理等群体策略。值得注意的是，群体策略源于个体决策，无需任何显式的协调机制或层级结构，这展示了集体智能如何随着协同进化而涌现。通过定量分析，我们进一步发现，集体智能能够从大规模智能体的协作与竞争中涌现，并产生超出我们预期的行为。例如，如图1所示，智能体学会排成对角线队列并整体移动，从而将地图的一部分隔离开来。无需任何先验知识，这一高效策略便通过自发探索涌现出来。此外，我们还进行了详细的消融研究，以展示一些可能有助于大规模智能体强化学习的实现技术。



（a）蓝色是我们的策略，黄色是 TB。　　（b）黄色是我们的策略，蓝色是 TB。

图1：我们的策略与 TB 之间的两个回合，其中我们的工人排成对角线队列。我们的智能体发现，这是一种有效扩张领土并限制敌人移动的方式。

我们的主要贡献如下：1）我们重新引入大规模智能体强化学习，将其作为研究集体智能的场景，并提出一个名为 Lux 的新环境作为起点。2）我们提供证据表明，在 Lux 中，集体智能通过大规模智能体的协同进化、协作与竞争而涌现。3）我们讨论了所提解决方案的实现细节，这些细节或可为大规模智能体强化学习提供有价值的见解。

2 相关工作

多智能体环境。许多环境（例如多智能体粒子环境（MPE）（Lowe 等人，2017年）和 Google Research Football（Kurach 等人，2020年））被提出用于研究多智能体协作与竞争。对于多智能体协作，星际争霸多智能体挑战（SMAC）（Samvelyan 等人，2019年）提供了一个通用测试平台。然而，SMAC 专注于分散式微操场景，其中参与游戏的智能体仅约 30 个。在大规模智能体环境中，Neural MMO（Suarez 等人，2021）是一种开放式大型多人在线（MMO）游戏环境，最多支持 1024 个智能体。MAgent（Zheng 等人，2018年）是一种网格世界环境，最多支持一百万个智能体。我们提出了 Lux，这是一种大规模智能体强化学习环境，可支持数千个智能体同时在一个步骤中采取行动。与以往的大规模智能体环境不同，Lux 融入了即时战略（RTS）游戏动态机制，类似于 Battlecode（2022）和 MiniRTS（Tian 等人，2017年）。此外，Lux 通过频繁生成与死亡来扩大智能体数量，从而为这种大规模、高度动态的场景中的复杂策略提供了可能性。

[1]https://www.kaggle.com/c/lux-ai-2021/

多智能体强化学习中的信用分配。智能体之间的信用分配（Chang 等人，2003年）是多智能体协作面临的一项关键挑战。若干基于价值的多智能体算法（Sunehag 等人，2017年；Rashid 等人，2018年；Iqbal 和 Sha，2018年）使用线性模型或神经网络将全局价值分解为个体价值，这可视为一种隐式信用分配方式。另一种方法是计算智能体特定优势函数。例如，Foerster 等人（2017年）使用反事实遗憾来衡量各智能体对队伍的贡献。在复杂游戏中，Berner 等人（2019年）和 Ye 等人（2020年）为每个智能体使用手工设计的团队奖励，作为一种显式信用分配方法。与隐式价值分解方法相比，这种显式奖励塑形方法需要先验领域知识，且缺乏泛化能力。然而，这两种方法都仅适用于小规模群体场景，难以扩展到大规模、动态变化的智能体群体。Han 等人（2019年）采用网格级集中式策略，而非分散式策略来解决这一问题。它使用卷积神经网络，将像素级观测映射为每个像素上的动作，从而避免信用分配问题，同时实现高效的多智能体协作。在此基础上，我们将这种像素到像素架构与近端策略优化算法（Schulman 等人，2017年）及课程学习阶段相结合，应用于 Lux 环境。

集体智能与涌现行为。 集体智能，包括自组织和涌现行为（Wolpert & Tumer，1999年；Woolley 等人，2010年），与生物学和经济学研究有着悠久的渊源。关于涌现行为的研究通常强调，群体策略是在设计环境中通过多智能体协同进化而涌现的，而非由人工设计的协作机制产生。Baker 等人（2019年）在捉迷藏环境中使用强化学习智能体和自动课程（Leibo 等人，2019年），从而促成了工具使用行为的涌现。Yang 等人（2018年）提出使用百万智能体强化学习，研究智能体的群体行为如何随环境资源而变化。Zheng 等人（2021）使用双层深度强化学习框架，在经济环境中训练智能体和社会规划者，其中最优税收政策是协同适应的结果。Johanson 等人（2022年）研究了在由生产者和消费者构成的基于微观经济学的环境中，以物易物行为的涌现。这些环境中的动态通常会使智能体的行为处于人类可以理解的范围内，从而限制了可能涌现的策略。由于即时战略游戏为集体智能提供了理想的培养皿，我们将即时战略游戏动态融入环境，使简单规则能够诱发复杂的群体策略。

3 LUX

如同地球，适合集体智能演化的环境必须支持大规模智能体之间的竞争与协作。为此，我们提出了一个名为 Lux 的开源环境。在该环境中，两个队伍中的数百个智能体争夺资源并建造城市，如图 2 所示。

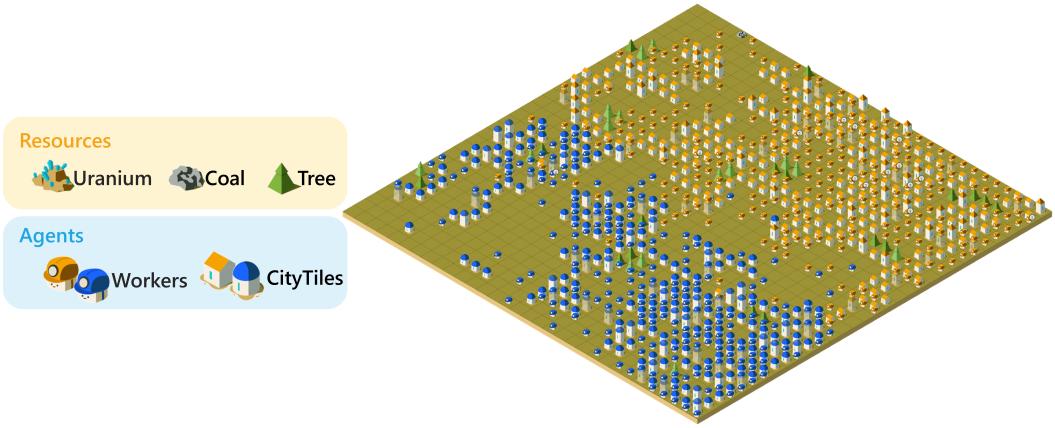


图2：**数百个智能体分属两个队伍的 Lux 游戏快照**。工人可以收集资源并建造城市地块。夜间，城市地块和工人需要燃料才能存活；燃料耗尽后，它们会被黑暗吞噬。最终，拥有更多城市的队伍获胜。

设置。 地图是一个大小为 12 至 32 的二维方格网格，散布着各种资源。一场回合包含 360 个回合，分为 9 个昼夜循环，每个循环由 30 天和 10 个夜晚组成。有两种基本单位，分别名为 工人 和城市地块。每个队伍开始时拥有一个工人 和一个城市地块。工人 可以收集资源并建造城市地块，而城市地块 也可以建造新的工人。工人会自动收集相邻的资源，并在友方城市地块上时将资源转化为燃料。夜晚时，城市地块和工人 会消耗燃料以存活；燃料耗尽后，它们将被黑暗吞噬。游戏结束时，拥有更多城市地块 的队伍获胜。有关该环境的更多详情，请参见附录 A.1。

观测与动作空间。 在观测方面，每支队伍都能完整获知游戏状态，包括全局地图、自身信息以及对手信息。在动作方面，每支队伍都需要为每个工人 和城市地块做出决策。工人 可以沿 4 个基本方向移动，并在资源充足时建造城市地块 。然而，工人 不能移动到有敌方城市地块 或工人的地块上。城市地块 可以建造工人，也可以进行研究以增加队伍的研究点数。研究点数达到一定数量后，队伍的工人 便能开采煤炭和铀等高级资源，将其转化为更多燃料。

Lux 中的多智能体强化学习。 对于多智能体强化学习研究而言，Lux 为多智能体建模和信用分配问题带来了颇具挑战性的情境。与其他环境不同，Lux 中的智能体数量庞大且动态变化。在 32×32 地图上，一支队伍的智能体数量可达到 1000。此外，工人 和城市地块 不断建造和损失，这使动态智能体的多智能体建模更加困难。精心设计的信用分配方案在小规模问题中或许有用；然而，面对数量庞大且动态变化的智能体，由于组合复杂性，这种方案并不切实际。此外，胜负稀疏奖励也给艰难的探索带来了另一项挑战。

Lux 中的即时战略游戏。乍看之下，Lux 似乎是一款袖珍版的即时战略游戏，类似于《星际争霸 II》。Lux 中的智能体需要在经济决策和个体控制之间取得平衡，这要求数百个智能体进行高层协作。然而，Lux 与即时战略游戏的主要区别在于单位的控制方式。在即时战略游戏中，低层级的单位行动由固定规则执行，因此人类玩家或 AI 可以专注于宏观策略和经济决策。在 Lux 中，移动和建造等原子动作都由学习得到的策略控制，因此其动作空间约比《星际争霸 II》大 10^{180}, 个数量级（Vinyals 等人，2019）。因此，成功的策略需要同时学会原子技能和群体策略，这对集体智能的涌现意义重大。

4 方法论

总体而言，我们使用标准算法近端策略优化（Schulman 等人，2017）和广义优势估计（Schulman 等人，2016）训练策略。为实现大规模智能体协同，我们采用像素到像素架构作为集中式策略（Han 等，2019；Isaiah 等人，2021），将观测和动作都表示为图像，并使用残差网络（He 等人，2016）作为骨干网络。为解决稀疏奖励问题，我们设计了三个阶段，每个阶段采用不同的奖励，构成渐进式课程。为便于说明，我们将地图上的每个单位称为“智能体”，并将控制所有智能体的集中式策略网络称为“策略”。

4.1 像素到像素架构

我们使用具有像素到像素架构的集中式策略来建模大规模智能体控制问题（Han 等，2019）。策略网络将图像作为输入观测，并以动作图的形式输出每个像素上的动作。更多实现细节见附录 A.2。

策略网络架构。 我们的策略网络架构如图 3 所示。输入图像（$C \times H \times W$）包含 C 个通道，其中包含关于自身、对手和全局状态的信息。我们采用 ResNet 风格卷积网络作为骨干网络。对于动作，我们使用一个卷积核大小为 1、输出通道数等于动作维度的卷积层。此外，我们使用一个展平层和一个全连接层，由评论家进行价值估计。我们使用有效动作掩码来消除不必要的探索。

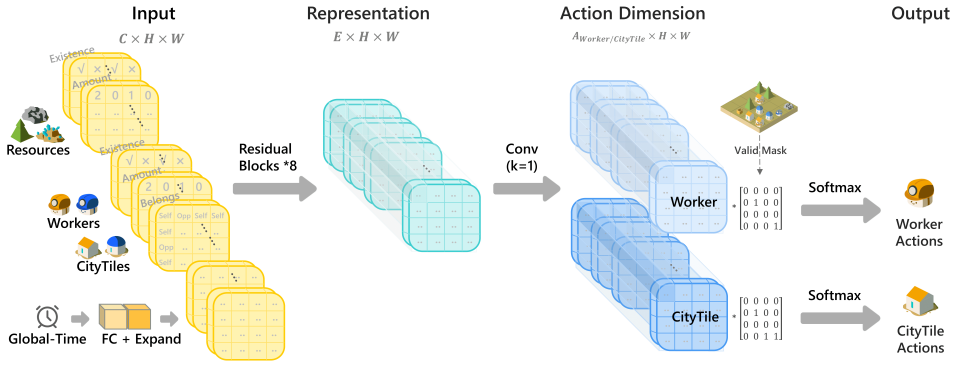


图 3：**策略网络架构**。 C 表示输入通道， H， W 表示地图的高度和宽度。 E 表示骨干网络中的特征图通道。输出通道 $A_{\text{Worker/CityTile}}$ 对应相应的动作维度。

为何采用集中式策略。 在 Lux 中，我们的策略需要在每个时间步控制数百个智能体。虽然多智能体强化学习中的分散式策略计算效率高且易于扩展，但它需要精心设计的信用分配机制。然而，在 Lux 中，由于智能体数量庞大且不断变化，信用分配问题变得愈发棘手。为避免这一问题，我们采用集中式策略来控制地图上的所有智能体。这种基于卷积网络的像素到像素架构兼具集中式方法和分散式方法的优势。卷积层在智能体之间充当参数共享机制，这与分散式方法中的共享策略网络类似。这种参数共享机制通过数据重用提高学习效率。此外，深层堆叠结构为全局信息提取和多智能体通信提供了较大的感受野，从而自然地避免了信用分配难题（Han 等，2019）。

4.2 课程训练阶段

在 Lux 中，智能体的目标是拥有比对手更多的城市地块，但最终结果只提供稀疏奖励（获胜得 1 分，-1 失败得 0 分），因此形成了困难探索问题（Badia 等，2020）。奖励塑形是强化学习中处理这一问题的常用方法。然而，人工设计的奖励很容易使智能体陷入特定的行为模式，策略也因此受限。因此，我们设计了三个奖励各不相同的阶段，构成渐进式课程。首先，我们使用密集奖励引导策略掌握基本技能。随后，我们逐步减少学习信号，并利用稀疏奖励鼓励策略探索更多样化的策略。

阶段 1：基本技能的密集奖励。首先，我们使用手工设计的密集奖励来鼓励基本技能。具体而言，对四种行为给予奖励，即增加工人和城市地块、研究点数以及燃料。更多详情见附录 A.2。

阶段 2：带缩放信号的稀疏奖励。在阶段 2 中，只有在一个回合结束时才会给予奖励。然而，我们的策略仍需要通过长期推理获得引导。我们通过加入关于获胜条件的微弱信号来调整奖励，即 $\pm\sqrt{|N_{self} - N_{op}|}$，其中 N_{self} 和 N_{op} 分别表示我方和敌方的城市地块数量，从而鼓励占有更多城市地块以赢得胜利。

阶段 3：胜负稀疏奖励。最后一个阶段采用胜负稀疏奖励（获胜时为 1，失败时为 -1）。在前两个阶段接受人工设计的引导后，胜负稀疏奖励鼓励我们的策略探索更多高级策略。

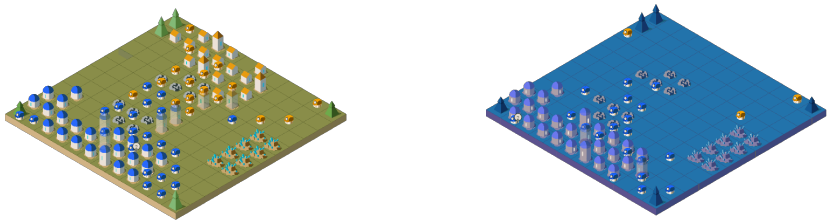
5 涌现式集体智能 NCE

通过大规模智能体协作与竞争，我们观察到了智能体演化的三个阶段。从头开始训练时，智能体会迅速掌握收集资源、建造城市等原子技能。经过约500万回合训练后，数十个智能体在区域尺度上开始出现初步协作。随着训练继续，协作范围从区

域尺度扩展至全局，涵盖长期经济决策和对数百个智能体的精确控制。这些全局策略源于大规模智能体交互与协同进化所促成的个体决策，无需任何显式协调机制，标志着集体智能的涌现。

5.1 原子技能

智能体的第一步是掌握原子技能。在密集奖励的引导下，工人学会朝资源移动以收集燃料，并建造城市地块、为其补充燃料，如图 4a 所示。不过，在这一阶段，智能体更倾向于各自行动，无法进行群体决策。例如，工人 往往会建造超出自身供养能力的城市地块，导致它们耗尽燃料，像图 4b 所示那样使大型城市突然崩溃。



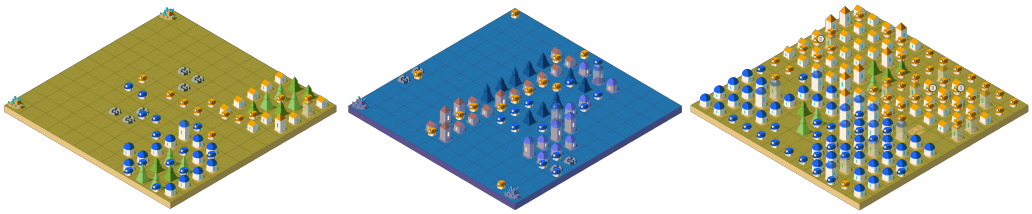
(a) 工人 收集资源并建造城市地块。　　(b) 城市地块耗尽燃料并崩溃。

图 4：**自博弈回合中原子技能的示意图**。 智能体习得原子技能，例如收集资源和建造城市地块。然而，由于缺乏群体协调，城市地块常常耗尽燃料并崩溃。

5.2 区域协调

随着训练的进行，区域协调逐渐出现，涉及局部区域内数十个智能体。例如，智能体学会在建造城市地块 之前谨慎选择位置，并形成自组织模式，以高效占据资源。我们介绍一些区域策略示例：

建造规划。 由于相邻建造的城市地块 可以共享燃料并降低夜间成本，智能体逐渐学会城市地块的位置对于城市生存和节省燃料至关重要。我们发现，智能体会探索出几种建造规划模式，如图 5 所示：1）在资源附近建造城市地块，以便更快获取燃料来源。2）将城市地块 排成一长行，形成如长城般阻挡敌人进攻的城市。3）将城市地块 建成方块，以降低夜间燃料成本。

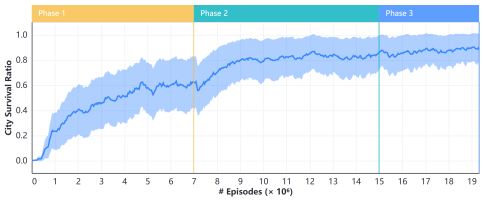


（a）在资源附近建造的城市地块。　（b）排成一行建造的城市地块。　（c）建成方块的城市地块 。

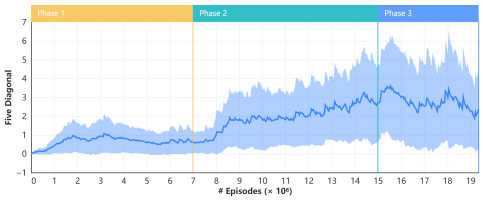
图5：**建造规划中涌现出的三种模式**。a）在资源附近建造，以便快速获取燃料。b）排成一行建造，形成用于防御的长城。c）以区块形式建造，以节省燃料。

我们使用城市生存率（一个回合中最终的城市地块数量除以该回合中城市地块的最大数量）来定量衡量这些建造模式的效果。如图6a所示，区域尺度的建造规划有效地帮助城市地块抵御黑暗。

领土划分。 我们还观察到一种自组织结构：如图1所示，几名工人排成一列对角线。这些工人同时向前和向后移动



(a) **城市存活率**：最终数量除以单个回合中的最大城市地块数量。



(b) **五对角线**：在单个回合中，五个或更多工人排列成对角线队列的次数。

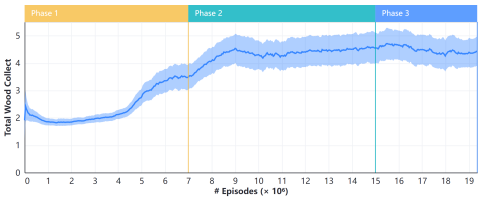
图 6：使用城市存活率和五对角线对区域协调进行定量分析。两项指标均通过自博弈进行评估。a) 随着训练的持续，城市存活率不断提高，表明区域建造规划能够有效帮助城市地块长久生存并发展。b) 在训练过程中，五对角线形状的出现频率不断增加，说明智能体逐渐掌握了这一策略。

，以整体形式保持队形；当其中任何一名死亡时，附近的新工人就会补上。在这种阵型中，他们能够有效守卫并扩张队伍的领土，同时限制敌人的移动。我们测量了一项名为五对角线的统计指标（在一场游戏中，五名或更多工人排成一列对角线的次数），以考察这种策略的使用频率。图6b中的结果表明，从长远来看，我们的智能体使用这一策略的频率通常会随着训练而增加，这说明它是一种习得的策略，而非偶然情况。

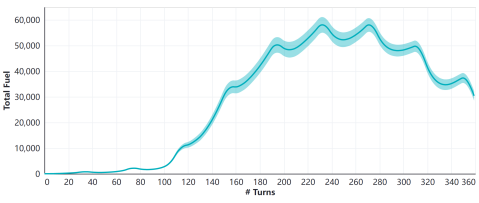
5.3 全局策略

与 SMAC 的微观管理场景（Samvelyan 等人，2019年）类似，多智能体协作中常常会出现数十个智能体之间的区域协调。然而，我们的智能体更进一步，实现了数百个智能体之间规模大得多的协调。下面我们对几种全局策略进行阐释和分析：

可持续发展。 在 Lux 中，城市发展与资源消耗之间的平衡是关键要素。在早期阶段，快速发展的城市常常面临严重的燃料短缺。随着时间推移，我们的策略逐渐学会根据资源分布和对手的行为，以可持续的速度发展城市，并使其与燃料生产相协调。我们观察到的另一种现象是保留树木。由于树木是 Lux 中唯一的可再生资源，森林保护对于保障长期燃料供应至关重要。我们的智能体会刻意避免过度砍伐森林，以保留树木，并在树林附近建造 城市地块 来抵御敌人的进攻。我们还使用另一项指标——木材总采集量（采集到的木材总量除以初始生成的木材总量）——来衡量这种森林保护策略对燃料供应的影响。图7a中的结果表明，这些保护策略显著提高了木材利用效率，使我们的智能体有时能够采集到超过地图上初始木材总量500%的木材。



(a) **木材总采集量**：采集的木材总量除以最初生成的树林数量。



(b) **燃料总量**：单个回合中的燃料储备使用我们最新模型在固定地图上的表现。

图7：**使用木材总采集量和燃料总量对全局策略进行定量分析**。这些指标通过自博弈进行评估。a) 随着我们的策略逐渐增强森林保护意识，木材利用效率也随之提高。b) 在一个回合中，我们的燃料储备持续累积，直到第240回合。此后，它会尝试建造更多城市地块以取得胜利。

全力争胜。 另一个令人惊讶的策略是，当回合即将结束时，我们的策略会迅速砍伐所有受保护的树木，并尝试尽可能多地建造城市地块以赢得胜利

如图 7b 所示。此外，我们观察到，有时城市在回合结束时只剩下极少的燃料，这表明几乎所有资源都已得到充分利用，因为结束时剩下的任何资源都是一种浪费。在结束前高效利用所有剩余资源极具挑战性，因为这需要整体计算所有工人和城市地块的燃料总消耗，并对每个智能体进行精确控制。我们认为，这一策略完美展现了长期经济决策与大规模智能体动员相结合所涌现出的集体智能。

6 实验

在本节中，我们通过消融研究，探讨大规模智能体设置下的策略实现和通用强化学习算法：1）我们通过采用不同的训练流程，研究课程学习阶段是否必不可少。结果表明，我们设计的课程学习能够帮助解决训练早期由稀疏奖励导致的困难探索问题，并促进超越人工设计的复杂策略涌现。2）我们进一步展示了模型在不同地图尺寸下的泛化能力。在尺寸为 32 的地图上评估时，在尺寸为 12 的地图上训练的策略仍保留了一些基本策略。经过仅 1.8 百万回合的微调后，迁移策略在尺寸为 32 的地图上对阵 TB 的胜率达到 90%。结果表明，我们的模型能够通过卷积层学习空间结构，从而为环境学习到可泛化的表示。3）我们将集中式策略与一种经过精心设计、采用团队奖励的标准分散式解决方案进行比较。集中式策略的胜率达到 98%。**实现细节见附录 A.2。分散式策略的实现和实验见附录 A.3。**

6.1 课程学习阶段的设计

我们开展实验，以探究课程学习阶段的必要性。我们采用了五种不同的方案：a）仅采用阶段1；b）采用阶段1和阶段 2，不采用阶段 3；c）采用阶段1和阶段 3，不采用阶段 2；d）采用阶段1、阶段 2和阶段 3（原方案）；e）仅采用阶段 3。

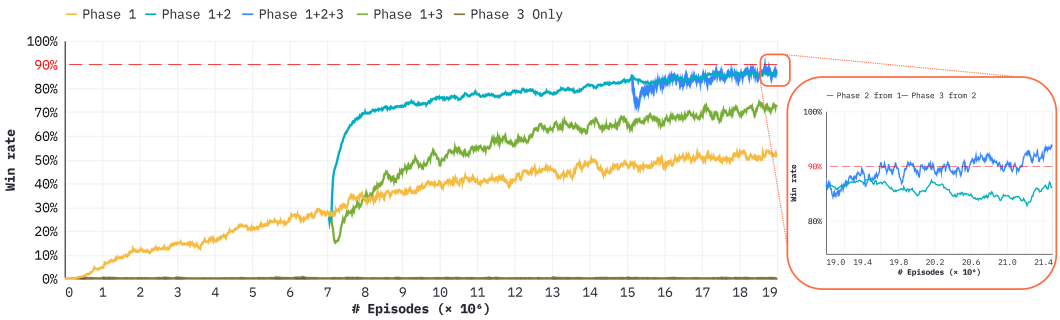


图 8：**不同训练阶段的胜率曲线**。所有胜率均在大小为 12 的地图上与 TB 对战进行评估。采用直接稀疏奖励进行训练时，胜率为 0%；仅采用阶段1的密集奖励进行训练时，胜率收敛在约 50%。与阶段 1+2 相比，阶段 1+3的提升较慢，胜率也较低，约为 70%。阶段 1+2 的胜率达到 85%，阶段 1+2+3 则进一步将其提升至 90% 以上。

阶段1 采用密集奖励，这对于帮助集中式策略为单个智能体习得原子技能至关重要。如图 8 所示，直接使用稀疏奖励进行训练时，我们的策略几乎无法学会任何基本技能；由于探索问题难以解决，胜率约为零。

阶段 2 采用缩放稀疏奖励，在整个学习过程中发挥双重作用：加速学习并提升性能。首先，使用密集奖励进行持续学习时，胜率收敛于 50%；而采用阶段 2 后，胜率迅速升至 70%。另一方面，如果不采用阶段 2，从阶段1切换到阶段 3会更具挑战性，即使训练时间更长，性能仍然较低。这表明，缩放稀疏奖励可以作为一种恰当的

过渡方式，在密集奖励与胜负稀疏奖励（应用于阶段 3）之间发挥作用，因为它明确告诉策略，拥有更多城市是获胜的关键。

阶段 3 采用稀疏的胜负奖励，进一步将最终表现提升至 90% 以上。阶段 2 训练的胜率收敛于 85%，若不进行阶段 3 训练，胜率便止步于此；胜负稀疏奖励则推动我们的策略继续探索，从而使总胜率达到 90%。

6.2 强化学习表示的泛化

我们提供了明确证据，表明从卷积神经网络和强化学习算法中学到的表示可以泛化到不同尺寸的地图。首先，我们将经过 12 尺寸地图训练的策略网络直接转移到 32 尺寸地图上。如图9所示，在更大的地图上仍能保留基本技能，例如工人 采集资源并为城市提供燃料，甚至还能展现出一些结构化的城市建造规划，以包围并保护木材资源。其次，经过约 100 万回合的额外微调后，该策略能迅速适应更大的地图，并在对战 TB 时达到 90% 的总胜率；相比之下，如图10所示，从头开始训练需要 160 万回合才能达到 20% 的胜率。

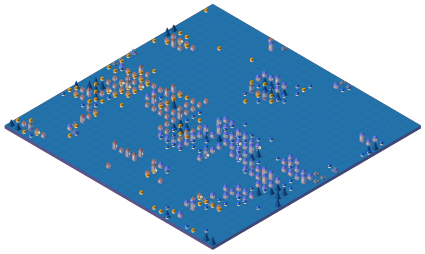


图9：**自博弈回合中泛化能力的示意图。**将针对 12×12 地图训练的策略直接迁移到 32×32 地图时，仍能保留一些策略，例如工人按照计划建造城市并为城市提供燃料。

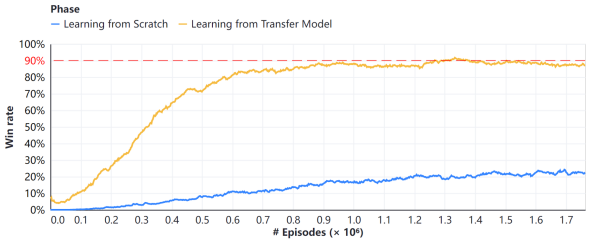


图10：**从头开始训练和迁移在 32×32 地图上的胜率曲线。**两者均以 TB 为对手进行评估。经过 100 万步的微调阶段后，迁移策略的胜率达到 90%；而从头开始训练需要 160 万。回合才能达到 20% 的胜率。

结果证明了我们模型的泛化能力，并为如何加快大型地图上的策略训练提供了思路：我们可以先在小型地图上预训练策略，再通过微调阶段将其迁移到大型地图上。这一过程显著缩短了训练时间，因为较小的地图能够加快环境模拟和网络更新。例如，在 CPU 上，Lux 环境在尺寸为 32 的地图上的模拟速度比尺寸为 12 的地图慢 $2.5\times$；在 GPU 上，尺寸为 32 的地图上的策略网络更新速度慢 $5\times$。由于在尺寸为 12 的地图上训练既节省时间又能提高计算效率，我们的“预训练与微调”方案能够以更少的训练小时数实现更高的胜率。

7 讨论与未来工作

我们证明了大规模智能体的协作与竞争能够催生集体智能。作为概念验证，我们提出了 Lux，这是一个容纳数百个智能体并融入即时战略游戏动态的环境。通过标准强化学习算法和像素到像素的集中式建模，我们观察到智能体策略演化的多个阶段。我们的智能体展现出雄心勃勃的群体策略，这些策略基于对大规模智能体的精准个体控制，且无需显式协调机制，表明集体智能已经涌现。

我们希望基于 Lux 开展的工作能够成为迈向人工集体智能的一块基石。在 Lux 中，我们观察到单个时间步中的智能体数量最多可达 2000 个，但与自然界中数百万乃至数十亿个相互合作与竞争的生物体相比，这仍相形见绌。正如附录 A.4 中的实验所示，Lux 环境可以轻松扩展以容纳更多智能体，但当智能体数量达到百万级时，模拟和推理会变得极其缓慢。展望未来，环境设计与工程以及训练算法都需要进行大量

修改，以适应如此大的规模。我们也承认，Lux 中的即时战略游戏动态可能与现实世界问题并不完全一致。不过，以 Lux 为蓝图，可以纳入类似 Zheng 等人（2021）的经济规则和动态，从而为现实世界中的经济决策和政策提供一些参考。

参考文献

Adri`a Puigdom`enech Badia、Pablo Sprechmann、Alex Vitvitskyi、Zhaohan Daniel Guo、Bilal Piot、Steven Kapturowski、Olivier Tieleman、Mart´ın Arjovsky、Alexander Pritzel、Andrew Bolt 和 Charles Blundell。永不放弃：学习定向探索策略。载于 第八届国际学习表征会议，*ICLR 2020*，亚的斯亚贝巴，埃塞俄比亚，*2020*年*4*月*26*日至*30*日。OpenReview.net，2020。网址：https://openreview.net/forum?id=Sye57xStvB。

Bowen Baker、Ingmar Kanitscheider、Todor M. Markov、Yi Wu、Glenn Powell、Bob McGrew 和 Igor Mordatch。多智能体自课程学习中涌现的工具使用。*CoRR*，abs/1909.07528，2019年。网址：http://arxiv.org/abs/1909.07528。

Battlecode。Battlecode，2022。网址：https://battlecode.org/。

Christopher Berner、Greg Brockman、Brooke Chan、Vicki Cheung、Przemyslaw Debiak、Christy Dennison、David Farhi、Quirin Fischer、Shariq Hashme、Christopher Hesse、Rafal J´ozefowicz、Scott Gray、Catherine Olsson、Jakub Pachocki、Michael Petrov、Henrique Pond´e de Oliveira Pinto、Jonathan Raiman、Tim Salimans、Jeremy Schlatter、Jonas Schneider、Szymon Sidor、Ilya Sutskever、Jie Tang、Filip Wolski 和 Susan Zhang。使用大规模深度强化学习玩 Dota 2。*CoRR*，abs/1912.06680，2019年。网址：http://arxiv.org/abs/1912.06680。

Yu-Han Chang、Tracey Ho 和 Leslie Pack 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，2003年。

Richard Dawkins 和 John Richard Krebs。物种之间及物种内部的军备竞赛。伦敦皇家学会会刊：*B*辑，生物科学，205(1161):489–511，1979年。

Jakob Foerster、Gregory Farquhar、Triantafyllos Afouras、Nantas Nardelli 和 Shimon Whiteson。反事实多智能体策略梯度。全国人工智能大会，2017年。

David Ha 和 Yujin Tang。深度学习的集体智能：近期发展的综述nts。集体智能，1(1):26339137221114874，2022。

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 Kamalika Chaudhuri 和 Ruslan Salakhutdinov（编），第*36*届国际机器学习大会论文集，第97卷，机器学习研究论文集，第2576–2585页。PMLR，2019年6月9日至15日。网址：https://proceedings.mlr.press/v97/han19a.html。

Kaiming He、张祥雨、任少卿和孙剑。用于图像识别的深度残差学习。载于电气电子工程师学会计算机视觉与模式识别会议论文集，第 770–778 页，2016年。

ShaIqbal 和 Fei Sha。用于多智能体强化学习的演员-注意力-评论家。国际机器学习会议，2018年。

Pressman Isaiah、Kirwin Liam 和 Sturrock Robert。Kaggle Lux AI 2021，2021年12月。网址：https: //github.com/IsaiahPressman/Kaggle_Lux_AI_2021。

Michael Bradley Johanson、Edward Hughes、Finbarr Timbers 和 Joel Z. Leibo。多智能体强化学习中的涌现式易货行为。*CoRR*，abs/2205.06760，2022。doi：10.48550/ arXiv.2205.06760。网址：https://doi.org/10.48550/arXiv.2205.06760。

Diederik P Kingma 和 Jimmy Ba。Adam：一种随机优化方法。*arXiv* 预印本 *arXiv:1412.6980*，2014年。

Karol Kurach、Anton Raichuk、Piotr Stanczyk、Michal Zajac、Olivier Bachem、Lasse Espeholt、Car- los Riquelme、Damien Vincent、Marcin Michalski、Olivier Bousquet 和 Sylvain Gelly。Google Research Football：一种新颖的强化学习环境。载于 第三十四届 *AAAI* 人工智能大会，*AAAI 2020*，第三十二届创新人工智能应用大会，*IAAI 2020*，第十届 *AAAI* 人工智能教育进展研讨会，*EAAI 2020*，美国纽约，*2020*年*2*月*7*日至*12*日，第 4501–4510 页。AAAI 出版社，2020年。

Joel Z Leibo、Edward Hughes、Marc Lanctot 和 Thore Graepel。自动课程与社会互动中创新的涌现：多智能体智能研究宣言。*arXiv* 预印本 *arXiv:1903.00742*，2019年。

Ryan Lowe、Yi I Wu、Aviv Tamar、Jean Harb、OpenAI 的 Pieter Abbeel 和 Igor Mordatch。面向混合合作竞争环境的多智能体演员-评论家。神经信息处理系统进展，第30卷，2017年。

Tabish Rashid、Mikayel Samvelyan、Christian Schroeder de Witt、Gregory Farquhar、Jakob Foer- ster 和 Shimon Whiteson。QMIX：用于深度多智能体强化学习的单调价值函数分解。*arXiv*：学习，2018年。

Mikayel Samvelyan、Tabish Rashid、Christian Schroeder de Witt、Gregory Farquhar、Nantas Nardelli、Tim G. J. Rudner、Chia-Man Hung、Philiph H. S. Torr、Jakob Foerster 和 Shimon Whiteson。星际争霸多智能体挑战。*CoRR*，abs/1902.04043，2019年。

John Schulman、Philipp Moritz、Sergey Levine、Michael I. Jordan 和 Pieter Abbeel。使用广义优势估计进行高维连续控制。载于 Yoshua Bengio 和 Yann LeCun（编），第四届国际学习表征会议，*ICLR 2016*，圣胡安，波多黎各，*2016*年*5*月*2*日至*4*日，会议论文集，2016年。

John Schulman、Filip Wolski、Prafulla Dhariwal、Alec Radford 和 Oleg Klimov。近端策略优化算法。*CoRR*，abs/1707.06347，2017年。网址：http://arxiv.org/abs/ 1707.06347。

Joseph Suarez、Yilun Du、Clare Zhu、Igor Mordatch 和 Phillip Isola。用于大规模多智能体研究的神经 MMO 平台。载于 J. Vanschoren 和 S. Yeung（编），神经信息处理系统数据集与基准赛道论文集，第 1 卷，2021。网址：https://datasets-benchmarks-proceedings.neurips.cc/paper/2021/file/44f683a84163b3523afe57c2e008bc8c-Paper-round1.pdf。

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价值分解网络。*arXiv* 预印本 *arXiv:1706.05296*，2017年。

Yuandong Tian、Qucheng Gong、Wenling Shang、Yuxin Wu 和 C. Lawrence Zitnick。Elf：一个面向即时战略游戏的全面、轻量且灵活的研究平台。神经信息处理系统进展（*NIPS*），2017年。

Oriol Vinyals、Igor Babuschkin、Wojciech M Czarnecki、Micha¨el Mathieu、Andrew Dudzik、Juny-oung Chung、David H Choi、Richard Powell、Timo Ewalds、Petko Georgiev 等。利用多智能体强化学习在星际争霸 II 中达到宗师级水平。《自然》，575(7782):350–354，2019年。

DavidH Wolpert和Kagan Tumer。《集体智能导论》。*arXiv*预印本 *cs/9908014*，1999年。

Anita Williams Woolley、Christopher F Chabris、Alex Pentland、Nada Hashmi和Thomas W Malone。人类群体表现中存在集体智能因子的证据。《科学》，330(6004):686–688，2010年。

Yaodong Yang、Lantao Yu、Yiwei Bai、Ying Wen、Weinan Zhang 和 Jun Wang。利用百万智能体强化学习研究 AI 群体动态。在第 *17* 届国际自主智能体与多智能体系统会议论文集，AAMAS ’18，第 2133–2135 页，南卡罗来纳州里奇兰，2018年。国际自主智能体与多智能体系统基金会。

叶德恒、陈贵斌、张文、陈胜、袁博、刘博、陈佳、刘钊、邱福豪、于洪生等。利用深度强化学习玩完整的 MOBA 游戏。神经信息处理系统进展，33:621–632，2020。

郑连敏、杨嘉诚、蔡涵、周明、Weinan Zhang、Jun Wang 和俞扬。《Magent：一个用于人工集体智能的多智能体强化学习平台》。载于第三十二届*AAAI*人工智能大会，2018年。

郑思蒂芬、亚历山大·特罗特、苏尼尔·斯里尼瓦萨、戴维·C·帕克斯和理查德·索彻。《人工智能经济学家：通过双层深度强化学习实现最优经济政策设计》。*CoRR*，abs/2108.02755，2021。网址：https://arxiv.org/abs/2108.02755。

A 附录

A.1 Lux 详细规则

为便于理解，正文中对环境规则（包括单位类型和动作空间）进行了简化。本部分将详细介绍环境设计和规则。我们使用的 Lux 版本与 Kaggle Lux AI S1竞赛[2]中的版本兼容。规则也可参见 https://www.lux-ai.org/specs-2021，以下内容是对原始规则重新排版并稍作修改后的版本。

背景。 夜色深沉，恐怖无处不在。两支队伍必须抵御黑暗、收集资源，并不断发展。白天，大家争分夺秒地收集资源并建造设施，为即将来临的夜晚做好准备。请谨慎规划并扩张——任何无法产生足够光亮的城市都会被黑暗吞噬。

环境。 在 Lux AI挑战赛第一赛季中，两支竞争队伍分别控制一支由单位和城市地块 组成的队伍，收集资源以供给各自的城市，主要目标是在回合制游戏结束时拥有尽可能多的城市地块。双方都能完整了解整个游戏状态，并利用这些信息优化资源收集、与对手争夺稀缺资源、建造城市以获得分数。每位参赛者都必须使用自己选择的编程语言编写策略。每个回合，你的智能体有 3 秒时间提交行动，超出的时间不会累积到后续回合。每场游戏都会分配 60 秒的时间额度，每当你超过单回合 3 秒的限制时，便会消耗这部分时间。用完全部 60 秒后，若再次超过 3 秒限制，你的智能体将被冻结，无法再提交行动。

地图。 Lux 的世界以二维网格表示。坐标向东（向右）和向南（向下）递增。地图始终为正方形，边长可以是 12、16、24 或 32 个地块。（0,0）坐标位于左上角。

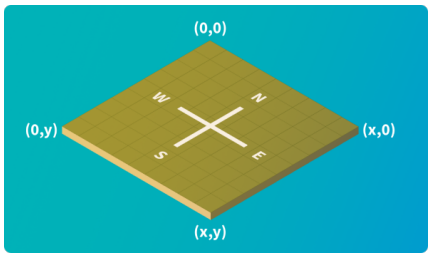


图11：Lux 中地图的规格。

地图包含多种特征，包括资源（木材、煤炭、铀）、单位（工人、货车）、城市地块和道路。为避免地图偏向某一位玩家，地图保证始终关于垂直轴或水平轴对称。每位玩家开始时都会拥有一块城市地块 和一个位于该工人 上的城市地块。

资源。 资源共有 3 种：木材、煤炭和铀（燃料效率依次递增）。这些资源由工人采集，然后工人 移动到

[2]https://www.kaggle.com/c/lux-ai-2021/

表 1：资源收集与转换的规格。

资源类型	研究点数前置要求	单位燃料价值	每回合收集单位数
Wood	0	1	20
Coal	50	10	5
铀	200	40	2

城市地块上，资源便会被卸下，随后转化为城市的燃料。某些资源需要研究点数达到要求后才能采集。木材尤其特别，可以再生。每回合，每个木材地块的木材数量都会增加其当前木材数量的 2.5%（向上取整）。已经耗尽的木材地块不会再生。只有木材数量少于 500 的木材地块才能再生。

收集机制。 每回合结束时，工人 会自动从所有相邻（北、东、南、西或中心）的资源地块中获取资源，前提是他们能够从这些地块收集资源。收集量按照当前的对称公式计算：

按铀、煤炭、木材资源的顺序依次处理：

- 每个单位都会向相邻的、属于当前正在处理的资源类型的地块提出收集请求，要求从每个地块收集相同数量的资源，使收集到的资源量足以让该单位的货物超过容量。例如，工人 携带着 60 单位木材，且相邻有 3 个木材地块时，会向每个地块请求 14 单位木材，最终获得 40 单位木材，并浪费 2 单位。

- 然后，当前正在处理的资源类型的所有地块都会尝试满足请求。如果无法满足，就会确保所有未满足的请求获得相同数量的资源，剩余资源则会被浪费。例如，如果 4名工人 正在开采一块有 25 单位木材的地块，其中一人只请求 5 单位，而其他人各请求 20 单位木材，那么首先所有工人各获得 5 单位木材，剩下 5 单位木材，可供另外 3名工人 继续收集。这些木材可以平均分配给最后 3名工人，每人分得 1 单位，剩下的 2 单位则会被浪费。

工人无法在城市地块上采矿。相反，如果城市地块上至少有一个工人 ，该城市地块 每回合就会自动以与工人 相同的速度收集相邻资源，并将所有资源直接转化为燃料。城市地块 的收集机制与工人 相同，你可以将城市地块视为一个独立的工人 来收集资源。

行动。 单位和城市地块 在满足特定条件时可在每回合执行动作。通常，所有动作会同时生效，并根据回合开始时的游戏状态进行验证。接下来的几节将详细介绍单位和城市地块 。

城市地块。 城市地块 是一种占据一个地块空间的建筑。相邻的城市地块 共同组成一座城市。每个城市地块在拥有冷却时间 < 1时，可以执行一个动作。

行动：

- 建造工人 - 在此城市地块 上建造工人 单位（如果当前拥有的工人 数量加上 + 货车数量等于拥有的城市地块数量，则无法建造工人 ）。
- 建造运输车 - 在此城市地块上建造货车单位（如果当前拥有的工人 + 货车数量等于拥有的城市地块数量，则无法建造货车）。
- 研究 - 将队伍研究点数增加 1。

单位。单位有两种类型：工人和货车。每个单位在冷却时间结束后 < 1都可以执行一次动作。所有单位都可以选择移动行动，并向 5 个方向中的任意一个移动：北、东、南、西或中心。此外，所有单位都可以携带通过自动采集或资源转移获得的原始资源。工人最多可携带 100 单位的资源，而货车最多可携带 2000 单位的资源。当单位移动到友方城市地块上时，构成该城市的城市地块会将其携带的所有资源转换为燃料。

没有城市地块的地块上最多只能有一个单位。此外，单位不能移动到敌方队伍的城市地块上。不过，单位可以在友方城市地块上彼此叠放。如果两个单位试图移动到同一个非城市地块地块上，则视为发生碰撞，移动行动将被取消。

表2：冷却时间规格。

单位类型	基础冷却时间
城市地块	10
工人	2
Cart	3

工人。行动：

• 移动 - 将单位朝 5 个方向之一移动：北、东、南、西或中心。
• 掠夺 - 将单位所在地块的道路等级降低。5。
• 转移 - 将单位货物中的任意数量的一种资源类型转移给另一相邻单位（回合开始时相邻），但不得超过后者的货物容量。多余部分将退还给原单位。

• 建造城市地块- 在此工人正下方建造一个城市地块，前提是工人的货物中装有任意类型、总量为100的资源（货物已满），且该地块为空。如果建造成功，所有携带的资源都会被消耗，并建成一个初始资源为0的城市地块。

货车。行动：

• 移动 - 将单位朝5个方向之一移动：北、东、南、西或中心。
• 转移 - 将单位货物中的任意数量的一种资源类型转给另一个相邻单位（回合开始时相邻），但不得超过后者的货物容量。超出的部分会退回原单位。

冷却时间。 城市地块、工人和货车在每次行动后都会进入冷却状态。单位和城市地块 只有在冷却时间 < 1 时才能行动。每回合结束时，在修建和掠夺道路之后，每个单位的冷却时间会减少 1，并额外减少其回合结束时所在道路的等级。道路等级不会影响城市地块 ，其冷却时间始终减少 1。冷却时间最低为 0。执行行动后，单位的冷却时间会增加基础冷却时间，具体数值见表2。

道路。 货车在地图上行驶时，会逐渐铺设道路，使所有单位都能移动得更快。每回合结束时，货车所在地块的道路等级会提升 0.75。道路等级越高，单位移动和执行行动的速度就越快。所有地块的道路等级初始为 0，最高为 6。此外，城市地块 的道路等级会自动达到上限 6。工人还可以通过掠夺行动破坏道路，每次使道路等级降低 0.5。如果一座城市被黑暗吞噬，该城市城市地块 上所有地块的道路等级都会归零。

昼夜循环。 昼夜循环由一个 40 回合的周期组成，前 30 回合为白天回合，最后 10 回合为夜间回合。一场对局共有 360 回合，构成 9 个周期。夜间，单位和城市需要产生光照才能存活。每个夜间回合，每个单位和城市地块 都会消耗一定量的燃料，消耗速率见表 3。特别是，单位会使用携带的资源产生光照，而城市地块 会使用燃料产生光照。工人和货车只有在不处于城市地块上时才需要消耗资源。在城市之外，工人 和货车必须消耗完整单位的资源来满足夜间需求。例如，如果工人 携带了 1 份木材和 5 份铀，就会先消耗一整份木材来获得 1 单位燃料，然后再消耗一整份铀来满足剩余 3 单位燃料的需求，浪费 37 单位燃料。单位总是会优先消耗效率最低的资源。

最后，在夜间，单位会额外获得 $2\times$基础冷却时间。如果任何单位在夜间耗尽燃料，就会从游戏中移除，永远消失在黑夜之中。然而，如果城市耗尽燃料，整座城市及其拥有的所有城市地块 都会陷入黑暗，并从游戏中移除。

游戏结算顺序。 为避免对每个回合如何结算的细节产生混淆，我们在此列出游戏结算顺序以及动作的应用方式。游戏中所有动作都会先根据当前游戏状态进行验证，以确定其是否有效。随后，动作与游戏事件将按以下顺序结算，并在每个步骤内同时处理：

表 3：文件燃料消耗的规格，n_{adj} 表示相邻友方城市地块的数量。

单位类型	城市内燃料消耗	城市外燃料消耗
城市地块	$23 - 5 \times n_{adj}$	N/A
工人	0	10
Cart	0	4

1. 城市地块行动，并增加冷却时间。2. 单位行动，并增加冷却时间。3. 建造道路。4. 收集资源。5. 在城市地块上掉落资源。6. 如果处于夜间，则让单位消耗资源，并让城市地块 消耗燃料。7. 让未耗尽的木材地块重新长出木材。8. 处理/计算每个单位和城市地块的冷却时间。

验证标准唯一的例外是，单位可以在格子之间顺畅移动，也就是说，如果两个单位相邻，它们可以在一个回合内交换位置。除此之外，不允许执行诸如一个单位建造城市地块，然后另一个单位移动到新建城市地块上的动作，因为当前状态中尚不存在这座新建的城市，而且单位不能移动到城市地块以外的其他单位所在位置。

获胜条件。 经过 360 个回合后，拥有最多城市地块 的队伍获胜。如果平局，则棋盘上拥有单位最多的队伍获胜。如果仍然平局，则游戏判定为平局。如果某个队伍不再拥有任何单位或城市地块，游戏也可能提前结束。此时另一支队伍获胜。

A.2 补充实现细节

Det本节展示了有关我们策略实现的详细信息，包括特征 e
engi工程、网络设计和强化学习算法实现。

PPO 实现。采用标准 PPO 损失（Schulman 等人，2017年）作为策略损失，以优化策略网络并估计优势；我们使用广义优势估计（Schulman 等人，2016年），轨迹长度设为 32。对于阶段1中使用密集奖励的训练，我们将 GAE 参数设为$\lambda = 0.95$，折扣因子设为 $\gamma = 0.99$。对于阶段 2 和阶段 3 中使用稀疏奖励的训练，我们将$\lambda = 1$，$\gamma = 1$。我们采用 PPO2 和裁剪操作，并将裁剪比率设为 $\epsilon = 0.1$。使用均方损失优化评论家的价值头，并将价值损失的权重设为 0.5。我们还计算熵损失，以系数 0.1 鼓励探索。我们使用 Adam（Kingma 和 Ba，2014年）作为优化器，学习率为 $1e-4$。计算资源方面，我们使用一块 NVIDIA-V100 GPU 和 600 个 CPU 核心。

输入。 输入到我们策略网络中的特征由两部分组成：1）包含时间步和燃料总量等全局信息的向量。全局信息向量中的特征及其对应的数据规范列于表4。2）包含地图信息以及我方和敌方工人 与城市地块 位置的图像输入列于表5。

密集奖励设计。 表6列出了详细的密集奖励设计。针对城市地块 和工人的特定行为，设置了四种奖励。总奖励是四个子奖励之和。

数据预处理。整个数据预处理流程如图12所示。全局信息向量分为两部分：维度为51的独热特征和维度为18的其他特征。独热特征输入输出维度为9的线性层进行嵌入。为了统一输入的图像形状，长度为9的嵌入向量和长度为18的其他全局信息向量分别扩展为各自通道中的图像尺寸，其中每个通道内的所有像素值都相同。随后，这些扩展后的图像特征（$9 \times H \times W$和 $18 \times H \times W$、 H和 W分别表示地图高度和宽度）与原始图像

表 4：输入特征第 1 部分：全局信息向量。

特征描述	Type	范围	归一化系数
当前周期	独热编码	[0,8]	N/A
当前周期中的当前回合	独热编码	[0,39]	N/A
是否处于夜间	独热编码	[0,1]	N/A
自有城市地块数量	Int	[0,1024]	100
敌方城市地块数量	Int	[0,1024]	100
己方单位数量	Int	[0,1024]	100
敌方单位数量	Int	[0,1024]	100
己方研究点数	Int	[0,200]	200
敌方研究点数	Int	[0,200]	200
己方燃料总量	Int	$[0,10^6]$	2300
敌方燃料总量	Int	$[0,10^6]$	2300
己方每个城市地块的平均燃料	浮点数	$[0,10^4]$	230
敌方每个城市地块的平均燃料	浮点数	$[0,10^4]$	230
己方燃料总消耗	Int	$[0,10^5]$	230
敌方燃料总消耗	Int	$[0,10^5]$	230
己方平均燃料成本/个城市地块	浮点数	[0,23]	23
敌方平均燃料成本/个城市地块	浮点数	[0,23]	23
如果己方队伍能采集煤炭	Bool	[0,1]	N/A
敌方队伍能否采集煤炭	Bool	[0,1]	N/A
己方队伍能否采集铀	Bool	[0,1]	N/A
敌方队伍能否采集铀	Bool	[0,1]	N/A

特征（$37 \times H \times W$）分别输入使用卷积核大小为1的卷积神经网络。然后，将三部分输入图像沿通道维度拼接在一起。再经过另一层卷积核大小为1、输入通道数为64、输出通道数为128的卷积层后，作为残差网络骨干网络输入的张量形状为 $128 \times H \times W$。

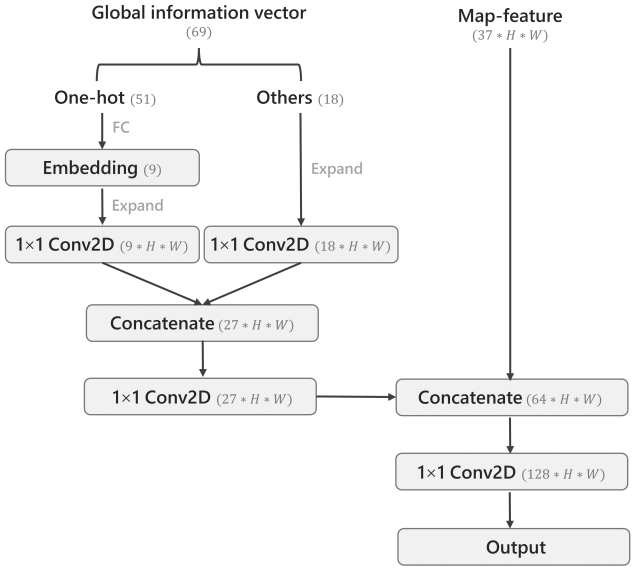


图12：**数据预处理流程**。全局信息分为两部分：独热编码信息和其他信息。独热向量首先通过全连接层进行嵌入，然后扩展至地图大小。经过 1×1 二维卷积后，独热编码信息与其他特征在通道维度上进行拼接，再通过另一层 1×1 二维卷积。全局特征和地图特征通过另一层 1×1 二维卷积进行拼接，该层的输入通道为64，输出通道为128。

表 5：输入特征第 2 部分：图像特征。

特征描述	Type	归一化系数
如果没有 工人	Bool	N/A
如果是自己的 工人	Bool	N/A
如果有敌方工人	Bool	N/A
如果没有矿车	Bool	N/A
如果有己方矿车	Bool	N/A
如果有敌方矿车	Bool	N/A
如果没有城市地块	Bool	N/A
如果有己方城市地块	Bool	N/A
如果敌方城市地块	Bool	N/A
道路等级	浮点数	6
工人冷却时间	浮点数	10
如果工人 可以行动	Bool	N/A
货车冷却时间	浮点数	10
货车是否可以行动	Bool	N/A
城市地块冷却时间	浮点数	10
如果城市地块可以行动	Bool	N/A
是否为资源	Bool	N/A
木材数量	Int	100
木材是否可以再生	Bool	N/A
煤炭数量	Int	100
铀数量	Int	100
工人携带木材数量	Int	100
工人携带煤炭数量	Int	100
工人铀携带量	Int	100
如果工人 达到携带上限	Bool	N/A
推车木材携带量	Int	100
推车煤炭携带量	Int	100
推车铀携带量	Int	100
城市地块燃料消耗	Int	100
城市地块平均燃料	浮点数	230
如果城市地块能撑过今晚	Bool	N/A
撑过今晚所需的燃料城市地块	Int	230
如果工人位于城市地块	Bool	N/A
如果货车位于城市地块	Bool	N/A
X 相对中心距离	浮点数	N/A
Y 相对中心距离	浮点数	N/A

表6：阶段1密集奖励的设计细节。

单位	行为	权重
城市地块	研究点数增长	0.01
	建造的单位	0.5
工人	燃料增加	0.0001
	城市地块 已建成	1

残差网络骨干网络。残差网络骨干网络由 8 个残差块组成。每个残差块包含两个卷积层和一个压缩与激励（SE）层。残差块的详细结构如图13所示。

输出。在残差网络骨干网络之后，我们使用多个头来输出动作与价值。残差网络骨干网络学习到的表示首先经过一个谱归一化层。对于动作头，我们为工人、货车和城市地块分别使用三个独立的头。每个头都是一个卷积层，卷积核大小为 1，输出通道数与相应的动作

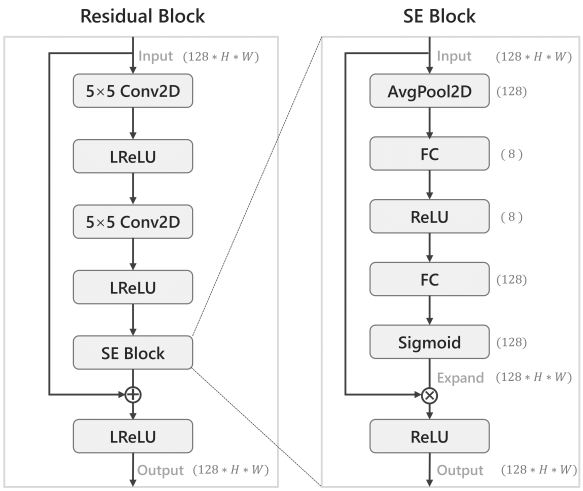


图13：**残差块设计与压缩与激励层**。 每个残差块由两个卷积层（卷积核大小为 $= 5$、填充为 $= 2$、步幅为 $= 1$）和作为激活函数的 LeakyReLU 组成。压缩与激励（SE）层由一个二维平均池化层和两个全连接层组成。

维度相同（工人为 19，货车为 17，城市地块为 4）。对于评论家的头，我们使用平均池化，将大小为 $128 \times H \times W$ 的表示转换为长度为 128 的向量。然后使用一个全连接层，得到评论家估计的单一价值。

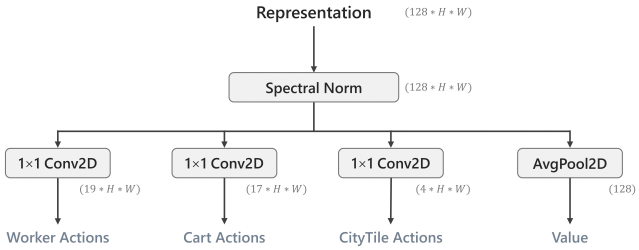


图14：**输出动作与价值**。 首先，经过一个谱归一化层后，接入三个动作头和一个价值头。 1×1 二维卷积用于输出工人、货车和城市地块的动作。二维平均池化和一个全连接层用于价值估计。

有效动作掩码。 有效动作掩码是强化学习中的一种常用技术，可消除不必要的探索并加快学习过程。我们根据以下规则计算有效动作掩码：

- 工人：当冷却时间为 > 1 时，所有动作均无效。当冷却时间为 < 1 时，移动到敌方城市地块 或有其他工人 位于其上的地块均无效；移动到友方城市地块 始终有效；只有当资源达到 100 时，建造城市地块 才有效。什么也不做始终有效。
- 城市地块：当冷却时间为 > 1 时，所有行动均无效。当冷却时间为 < 1 时，在工人数量少于工人 城市地块数量的情况下，建造工人是有效的；当队伍研究点数 < 200时，研究是有效的。始终可以选择不采取行动。

A.3 分散式策略实现与消融研究

本节将详细介绍我们的分散式策略实现，包括输入特征、网络设计和基于规则的智能体。

表 7：分散式策略输入：全局信息。

特征描述	Type	范围
观测到的智能体数量	Int	320
全局城市地块数量	独热编码	[0,320]
全局单位数量	独热编码	[0,320]
当前周期	独热编码	[0,9]
当前周期中的当前回合	独热编码	[0,39]
是否处于夜间	Bool	N/A
己方研究点数	Int	[0,200]
己方队伍能否采集煤炭	Bool	N/A
己方队伍能否采集铀	Bool	N/A

表8：分散式策略输入：自身信息。

特征描述	Type	范围
位置 X	Int	[0,31]
位置 Y	Int	[0,31]
位置 X	独热编码	[0,31]
位置 Y	独热编码	[0,31]
Type	Bool	N/A
是否在城市	Bool	N/A
存活	Bool	[N/A
冷却时间	独热编码	[0,9]
是否处于夜间	Bool	N/A
木材携带量	Int	[0,100]
煤炭携带量	Int	[0,100]
铀携带量	Int	[0,100]
木材携带量	独热编码	[0,100]
煤炭携带量	独热编码	[0,100]
铀携带量	独热编码	[0,100]
Fuel	Int	[0,4000]
Fuel	独热编码	[0,4000]

输入。我们的分散式策略的输入可分为四部分：全局信息（见表7）、自身信息（见表8）、其他智能体（队友和敌人）信息（见表9）以及地图信息（见表10）。

奖励塑形。 每个智能体会获得三种奖励：1）自身奖励，用于鼓励某些行为，例如生存、建造城市、收集资源和为城市提供燃料。2）城市地块奖励。尽管城市地块 奖励是基于规则的，但它用于引导工人 的行为，以支持城市地块。3）团队奖励。它由最终胜利奖励和队伍的平均奖励组成，用于鼓励智能体之间的协作。

网络设计。 输入分为七个部分，即全局特征、自身特征、友方工人特征、友方城市地块 特征、敌人工人 特征、敌人城市地块 特征和图像特征。对于前六种向量特征，我们使用六个不同的双层全连接网络进行特征提取。对于涉及多个单位的特征，我们沿单位维度执行最大池化。对于图像特征，我们使用三个卷积层，并将学习到的表示展平。随后，将这些表示拼接起来，并通过两个全连接层来生成动作和价值。

基于规则的智能体。 为简化起见，我们仅将工人 用作强化学习智能体，并将城市地块 用作基于规则的智能体。城市地块 的决策规则简单直观：1）如果工人 可以建造，就建造一名工人 。2）如果无法建造工人 ，且队伍研究点数达到 < 200，则进行研究。3）否则什么也不做。

表 9：分散式策略输入：其他智能体的信息。

其他智能体	特征描述	Type	范围
自身工人 × 160	是否为友方	Bool	N/A
	位置 X	Int	[0,31]
	位置 Y	Int	[0,31]
	距离	Int	[0,62]
	是否在城市	Bool	N/A
	冷却时间	独热编码	[0,3]
	木材携带量	Int	[0,100]
	煤炭携带量	Int	[0,100]
	铀携带量	Int	[0,100]
自有城市地块 × 160	是否为友方	Bool	N/A
	位置 X	Int	[0,31]
	位置 Y	Int	[0,31]
	距离	Int	[0,62]
	冷却时间	独热编码	[0,9]
	每个城市地块的平均燃料	浮点数	[0,2300]
	每晚燃料消耗	Int	[0,23]
	今晚是否能够存活	Bool	N/A
	今晚生存所需燃料	Int	[0,230]
敌人工人 × 160	是否为友方	Bool	N/A
	位置 X	Int	[0,31]
	位置 Y	Int	[0,31]
	距离	Int	[0,62]
	是否在城市	Bool	N/A
	冷却时间	独热编码	[0,3]
	木材携带量	Int	[0,100]
	煤炭携带量	Int	[0,100]
	铀携带量	Int	[0,100]
敌人城市地块 × 160	是否为友方	Bool	N/A
	位置 X	Int	[0,31]
	位置 Y	Int	[0,31]
	距离	Int	[0,62]
	冷却时间	独热编码	[0,9]
	每个城市地块的平均燃料	浮点数	[0,2300]
	每晚燃料消耗	Int	[0,23]
	今晚是否能够存活	Bool	N/A
	今晚生存所需燃料	Int	[0,230]

与分散式控制相比。为了说明我们的逐像素集中式控制方案，我们将其与分散式控制方案进行了对比实验。与分散式控制方案相比，集中式控制方案在100次运行中的胜率达到98%。如图16所示，分散式策略能够掌握收集资源和建造城市地块等基本技能。在基于团队的奖励的激励下，分散式智能体甚至掌握了初步的区域合作能力。然而，由于这种合作是由预先设计的奖励诱导产生的，因此只能应用于特定场景，无法扩展到其他复杂情形。例如，在一张有树林的局部地图中，分散式智能体起初占据优势；但由于其合作缺乏适应性，我们的智能体逐步在它们周围建造城市，并限制其发展，从而取得优势。结果到了第120回合，集中式策略已控制了地图上的所有资源。此外，集中式策略的演化还涌现出更多群体策略，例如采取积极策略，派遣一些工人占据关键资源并保护其不受对手侵扰。

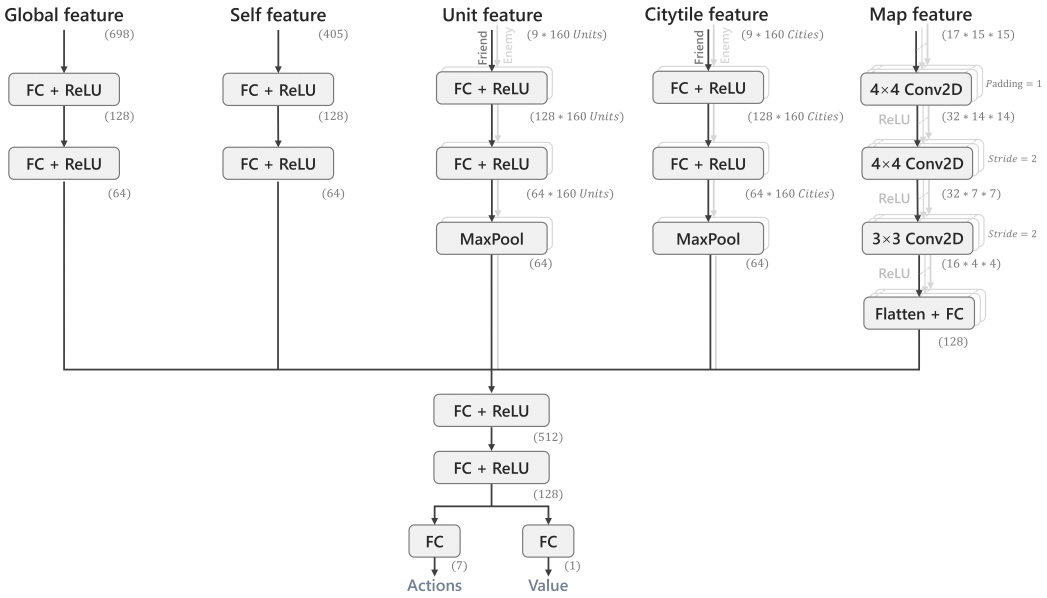


图15：**去中心化策略网络架构**。 对于全局特征和自身特征，我们使用两个全连接层进行提取。对于单位特征和城市地块特征，我们使用两个线性层，然后沿单位维度应用最大池化。地图特征则使用三个卷积层，以及一个展平层和线性层进行处理。随后将输出连接起来，并使用两个全连接层生成输出动作与价值。

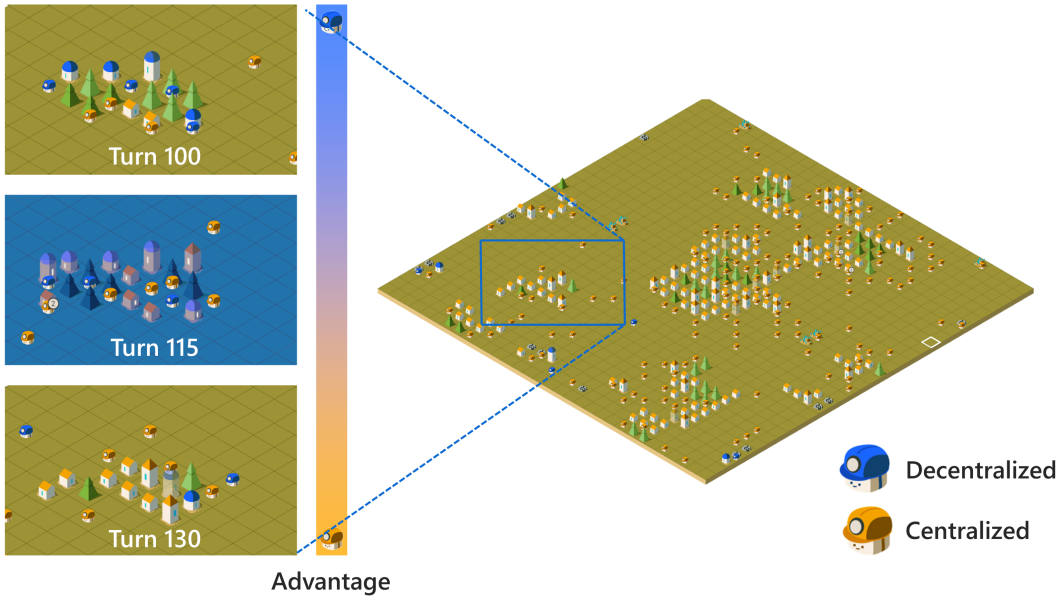


图16：**分散式策略与集中式策略之间的一个回合**。 黄色代表分散式策略，蓝色代表集中式策略。在一场局部战斗中，蓝色起初占据优势，但凭借更好的协作，黄色仅用30回合就扭转了局势。

表 10：分散式策略输入：地图信息。

地图信息	特征描述	Type	范围
资源地图	此处是否有木材	Bool	N/A
	木材储量	Int	[0,1000]
	此处是否有煤炭	Bool	N/A
	煤炭储量	Int	[0,1000]
	这里是否有铀	Bool	N/A
	铀储量	Int	[0,1000]
工人地图	是否为友方工人	Bool	N/A
	是否为敌人工人	Bool	N/A
	工人 冷却时间	Int	[0,3]
	工人 木材携带量	Int	[0,100]
	工人 煤炭携带量	Int	[0,100]
	工人 铀携带量	Int	[0,100]
城市地图	是否为友方城市地块	Bool	N/A
	是否为敌人城市地块	Bool	N/A
	城市地块 冷却时间	Int	[0,9]
	每城市地块平均燃料	浮点数	[0,2300]
	每晚燃料消耗	Int	[0,23]
	今晚是否能够存活	Bool	N/A
	今晚生存所需燃料	Int	[0,230]
	道路等级	Bool	N/A

表 11：去中心化策略奖励设计。

奖励类型	特征描述	权重
工人奖励	工人 死亡惩罚	-1
	工人 存活一个夜晚回合	0.05
	工人存活十个夜间回合	0.5
	工人建造一个城市地块	1每个城市地块
	建造城市节省的燃料	$0.05 \times$（23–燃料成本）
	工人燃料增加	每单位燃料奖励 0.005
	工人燃料捐赠	每单位燃料奖励 0.01
城市瓦片奖励	城市地块死亡惩罚	-1
	城市地块存活一个夜间回合	0.05
	城市地块熬过十个夜间回合	0.5
	城市地块研究点增加	每个点增加*0.02*
	研究点达到 50	1
	研究点达到 200	4
	城市地块建造一名工人	1
团队奖励	最终胜利奖励	100
	团队平均奖励	0.1×友方奖励的平均值

A.4 更多泛化研究

在第 6.2 节中，我们通过将地图尺寸为 12 时训练的策略转移到尺寸为 32 的地图上，展示了模型的泛化能力。我们进一步开展了更多研究，以深入考察所提出模型在更大地图上的泛化能力。结果表明，即使转移到更大的地图上，我们的模型仍然展现出令人惊讶的大规模智能体协同能力。

我们使用在 32×32 地图上训练的模型作为基础模型，并在不同地图尺寸上进行评估，不进行微调。在尺寸为48和64的地图上，我们的策略展现出了卓越的大规模智能体协同能力，如图17所示。

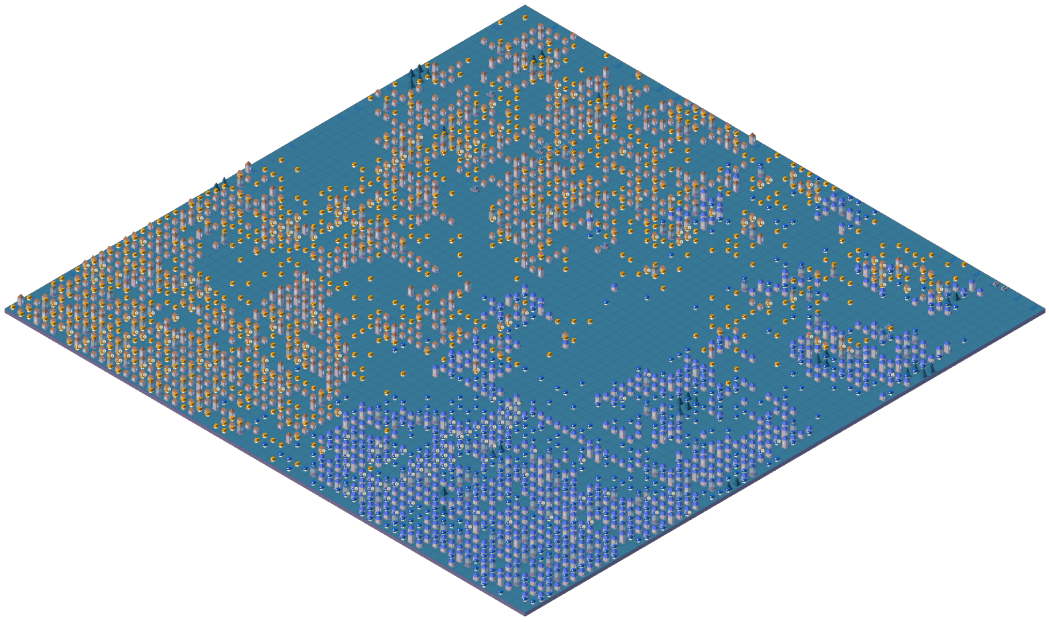


图17：**在大小为64的地图上进行策略迁移**。橙队有1069个城市地块和1059个单位，蓝队有779个城市地块和768个单位。在更大的地图上，我们的策略展现了协调数千个智能体的泛化能力。

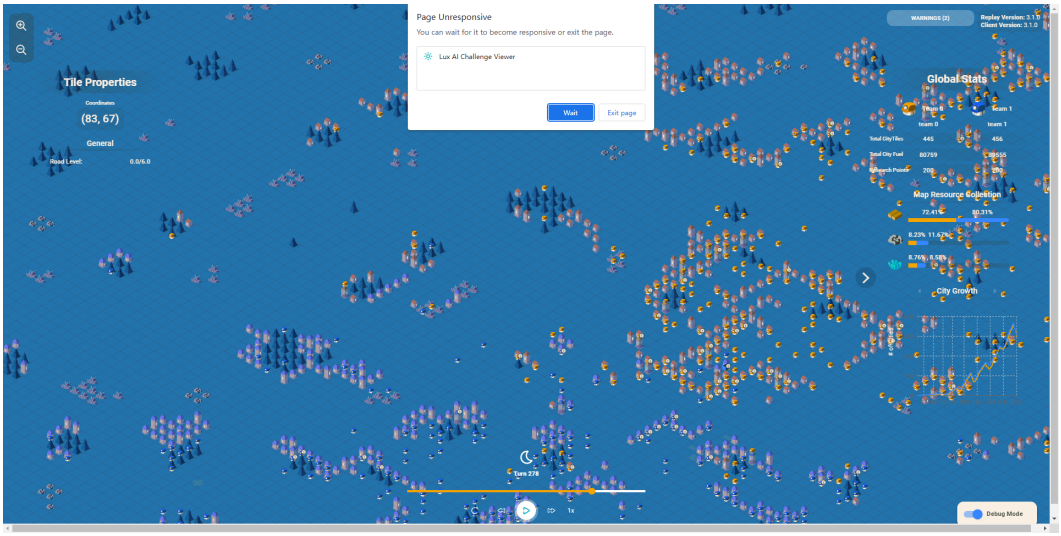


图18：**在大小为128的地图上进行策略迁移**。大地图和冷却机制限制了建造大型城市并铺满地图的能力。不过，我们的策略仍展现出在 32×32 地图上习得的技能和策略。这种大规模设置最终会导致网页查看器无响应。

此外，我们大胆尝试了 128×128 地图。然而，由于冷却机制，智能体很难在 360 回合内穿越地图，因此无法像在 32×32 地图中那样建造遍布地图的大型城市。此外，在更大的地图中，环境模拟速度会慢得多，每个回合约需 30 分钟。这表明，尽管 Lux 能够扩展至数百万智能体规模，但游戏核心和动态机制仍需大量修改，才能适应更大规模。尽管如此，我们发现，如图18所示，智能体仍能展现出它们在 32×32 地图中习得的技能和策略，这证明了我们的方法在百万智能体规模下的潜力。