arXiv:2304.10124v1 [cs.AI] 20 Apr 2023

通过多智能体非对称进化强化学习掌握非对称多人游戏

孙成露，张一驰，张宇，卢子凌，刘景彬，徐思佳和张卫东

网易人工智能实验室{sunchenglu, zhangyichi1, zhangyu15, lzln6779, liujingbin01, xusijia, zhangweidong02}@corp.netease.com

摘要

非对称多人游戏（AMP）是一种流行的游戏类型，其中多种智能体在游戏中相互竞争或协作。由于非对称环境具有不平衡的特性，采用典型的自我对弈训练方法，很难训练出能够击败顶尖人类玩家的强大智能体。我们提出了非对称进化训练（AET），这是一种新颖的多智能体强化学习框架，可在 AMP 游戏中同时训练多种智能体。我们设计了自适应数据调整（ADA）和环境随机化（ER），以优化 AET 流程。我们在一款名为猫和老鼠的复杂 AMP 游戏中测试了该方法。未使用任何人类数据训练的人工智能，在 65 场比赛中对阵顶尖人类玩家时，胜率达到 98.5%。消融实验表明，所提出的模块有助于提升该框架的效果。

1 引言

作为对现实的虚拟模拟，游戏是许多人工智能研究的良好测试平台。近年来，强化学习（RL）在游戏人工智能领域取得了快速发展。通过强化学习方法训练的许多人工智能，在不同类型的游戏中已达到甚至超越了顶尖人类玩家的水平，例如围棋 [白银等人研究的棋盘游戏， 2018]，麻将 [Li 等人研究的纸牌游戏， 2020]，《Quake》 *3* [Jaderberg 等人研究的第一人称射击（FPS）游戏， 2019]，《StarCraft》 *2* [Vinyals 等人研究的即时战略（RTS）游戏，以及《*Dota*》 *2* [Berner 等人研究的多人在线战术竞技（MOBA）游戏，以及《王者荣耀》 [Ye 等人（ 2020a ）]。这些游戏具有对称竞技场的特征，游戏双方通常拥有相同的初始资源、受控规则、基本能力、最终目标等。这些特征使得自我对弈方法在强化学习的训练过程中发挥着至关重要的作用。训练过程中获得的大量历史自生成模型，凭借足够的多样性和适当的能力，成为了优秀的对手。通过反复与历史模型对战，可以持续且快速地提升其能力 [Berner 等人（2019）； Vinyals等人， 2019], 直至获得强大的人工智能。

非对称多人游戏（AMP 游戏） [Bycer，2019；Harry， 2021] 是一种广受欢迎的游戏类型，也是游戏领域的重要组成部分。AMP 游戏中的双方通常由具有不同模式的智能体组成，例如踢击与防守[Bansal等人， 2018], 躲藏与寻找 [Baker 等人， 2019], 猫和老鼠、第五人格、黎明杀机。AMP 游戏中双方模式的不同意味着，双方在游戏规则、目标、玩家数量等方面都完全不同。玩其中一方的经验无法直接帮助玩家玩另一方[1]。与上文提到的对称游戏类似，大多数 AMP 游戏都是对抗性的零和环境 [Balduzzi 等人， 2019]。在这种情况下，如果采用自我对弈结构训练 AMP 游戏，一方的胜率通常会高于另一方，导致状态分布失衡。也就是说，一方的状态倾向于优势状态，而另一方则会遇到更多劣势状态，因此状态分布不够全面。此外，随着失衡加剧，较弱的智能体很难获得正奖励，从而产生奖励稀疏问题 [Trott 等人， 2019]。因此，AMP 游戏无法采用经典的自我对弈训练结构来提升人工智能性能。此前，AMP 游戏领域还没有强大的人工智能达到专业人类玩家的水平。猫和老鼠 是一款典型的 1 对 4 AMP 游戏[2]，，玩家可以扮演猫或老鼠。扮演猫时，玩家需要独自对抗四只老鼠；扮演老鼠时，玩家则需要与另外三名鼠玩家协作对抗猫。这款游戏具有策略多样、信息不完备、多智能体协作、状态空间高维、动作操作复杂等特点。因此，猫和老鼠 是 AMP 游戏领域合适的测试平台，附录 A 详细介绍了其玩法。

本文提出了一种采用非对称进化训练（AET）方法的多智能体强化学习框架。利用该框架，无需使用任何人类数据，即可同时为游戏双方训练两种模式的人工智能。作为多智能体模型的老鼠人工智能展现出了卓越的水平。

[1]虽然《*Dota 2* 》或《*StartCraft 2* 》中的玩家可以在不同阵营选择不同的英雄或种族，但这些游戏仍属于对称博弈，因为双方拥有相同的规则、目标等。从另一个角度看，对称游戏中的不同英雄或种族可以用不同的独热编码器或嵌入来表示。[2] https://tomandjerry.fandom.com/wiki/Tom_and_Jerry_Chase

通过击败顶尖人类玩家，取得了 98.5% 的胜率。在 AET 过程中，我们发现，AMP 游戏的特性可能导致双方智能体的胜率严重失衡。这一问题会造成老鼠智能体与猫智能体等级之间的严重失衡，进而导致人工智能性能停滞。一些研究人员 [Bansal等，2018年；Baker 等， 2019] 也发现了在 AMP 游戏中训练人工智能时存在的失衡问题，但他们关注的是人工智能的演化和自动课程，而非解决失衡现象。为了解决这一问题，我们采用了自适应数据调整（ADA）和环境随机化（ER）来优化 AET 过程。本研究的主要贡献如下：

- 本文提出了一种融合 AET 的新型多智能体深度强化学习框架，用于训练能够在复杂 AMP 游戏中击败顶尖玩家的人工智能。
- 我们设计了两个模块来解决 AET 中的不平衡问题，实验结果表明，这些模块不可或缺。

2 相关工作

本研究旨在训练能够在大规模 AMP 游戏中达到顶尖人类水平的人工智能。该游戏具有复杂的策略，是一个尚未探索的研究领域。此前，已有许多可借鉴的研究致力于在其他不同类型的游戏中训练顶尖人工智能。

2016年，阿尔法围棋 [白银 等人， 2016] 作为一款围棋人工智能，击败了世界冠军。阿尔法围棋通过强化学习与蒙特卡洛树搜索（MCTS）相结合的方式进行训练 [Coulom， 2006]。它采用自我博弈方法持续提升性能，展现了自我博弈在对称游戏人工智能训练中的强大作用。2018年，一款名为 OpenAI Five 的 *Dota 2*（一种复杂且对抗性强的多人在线战术竞技游戏）人工智能击败了许多顶尖职业电竞选手 [Berner 等人，2019]。OpenAI Five 从零开始通过强化学习进行训练；漫长的自我博弈训练过程和庞大的历史模型池保证了人工智能的多样性和稳定性。其训练过程和奖励重塑为我们的工作提供了指导。2019年，AlphaStar [Vinyals 等人， 2019] 作为一款星际争霸*II*人工智能，达到了人类顶尖水平。它先通过监督学习（SL）获得基础人工智能，随后结合强化学习与联赛训练框架，使人工智能不断进化。用于处理复杂实体的精巧网络架构，以及离策略学习改进方法，为我们的工作提供了参考。2020年，一款名为 Suphx 的日本麻将人工智能在天凤平台上达到十段 [Li 等人， 2020]。它同样结合了监督学习与强化学习，并采用全局奖励预测和预言机引导来提升人工智能性能。2020年，一款名为绝悟的王者荣耀人工智能问世，它利用强化学习和自我博弈训练击败了顶尖电竞选手 [Ye 等人，2020b；Ye 等人，2020a]。通过设计禁选流程，它成功控制了大多数英雄。他们通过在策略梯度部分对优势函数进行双重裁剪，改进了近端策略优化（PPO） [Schulman 等人，2017]；我们也采用双重裁剪来维持训练稳定性。除了上述研究之外，许多优秀的研究也运用强化学习在游戏领域训练多智能体人工智能，例如雷神之锤 *3* [Jaderberg 等人， 2019]以及*Google Research Football* [Kurach等人， 2020]。

上述游戏的特点是，游戏中对立双方的智能体采用相同的模式。这意味着双方在操作模式、限制规则和策略思路上几乎相同。双方人工智能可以采用相同的训练方法，例如数据处理和网络。我们的工作与上述研究的区别在于，我们通过设计的 AET 框架，同时训练了两个模式截然不同的人工智能。此外，我们结合 ADA 和 ER，通过解决不平衡问题来改进框架并提高人工智能的多样性。易于实现的系统设计有利于快速迭代模型。实验结果证明，训练出的人工智能已达到较高水平。

3 方法

3.1 AsymmetricRL 训练框架

所提出的训练框架主要由五个部分组成：鼠标训练器、猫训练器、采样器集群、评估器服务器和日志服务器，如图1所示。在该框架中，设计了一种变体演员-评论家结构来训练智能体。与传统演员-评论家结构不同[Mnih 等人， 2016], 该变体同时训练猫策略 $\pi_\theta^c(a_t^c|s_t^c)$、鼠标策略 $\pi_\mu^m(a_t^m|s_t^m)$、猫价值函数$V_\phi^c(\tilde{s}_t^c)$和鼠标价值函数 $V_\psi^m(\tilde{s}_t^m)$。 s_t^c和 s_t^m分别表示猫智能体和老鼠智能体在时间步 t观测到信息后的状态。令 $\tilde{s}$ 表示包含不可见信息的状态。 a_t^c 和 a_t^m分别表示猫智能体和老鼠智能体选择的动作。 θ 和 μ 分别是猫智能体和老鼠智能体的策略网络参数。同样， ϕ 和 ψ 表示价值网络的参数。

训练器和采样器分别由 GPU 服务器和 CPU 集群构成。基于各自采样器集群构建的老鼠和猫智能体在游戏环境中相互竞争。从老鼠采样器的游戏环境中提取的老鼠轨迹会发送到老鼠回放缓冲区。猫轨迹的传输路径也类似。训练器会定期从回放缓冲区中提取数据进行模型训练并更新网络参数，随后将最新模型发送到模型池。评估器服务器会在 CPU 集群上执行模型评估。猫鼠双方智能体会根据模型评估结果选择合适的模型作为对手。各模块中的各种日志都会作为反馈传输到日志服务器。

3.2 学习系统

算法

智能体使用 PPO [Schulman 等人、 2017], 一种标准的演员-评论家算法 [Konda 和 Tsitsiklis， 1999]进行训练。策略优势通过广义优势估计（GAE） [Schulman 等人、 2015]计算。由于所提出的分布式训练框架中模型的异步更新，采用双重裁剪近端策略优化 [Ye 等人，2020年b]来缓解离策略训练中的不稳定性。The

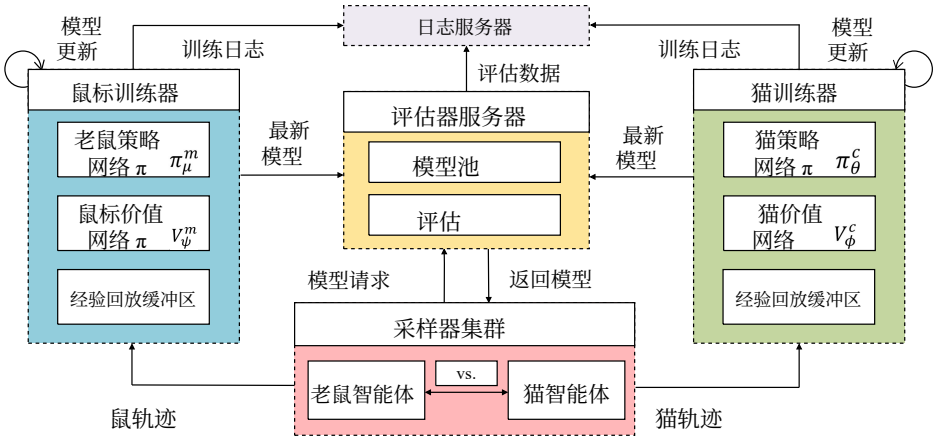

图1：非对称强化学习训练框架概览。

.

策略的重要性比率的形式为：

$$r_t^k = \frac{\pi^k(a_t^k|s_t^k)}{\pi_{old}^k(a_t^k|s_t^k)}, k \in c, m \tag{1}$$

其中， π_{old}^k 是智能体的旧策略， k 表示猫智能体或老鼠智能体。

智能体的策略目标形式如下：

$$L_p^k = \begin{cases} -\hat{\mathbb{E}}_t[max(min(r_t^k \hat{A}_t^k, c_t^k \hat{A}_t^k), \eta^k \hat{A}_t^k], & \hat{A}_t^k < 0 \\ -\hat{\mathbb{E}}_t[min(r_t^k \hat{A}_t^k, c_t^k \hat{A}_t^k)], & \hat{A}_t^k \geq 0 \end{cases} \tag{2}$$

ˆ其中， $\mathbb{E}_t[...]$ 表示有限批次样本的期望， $c_t^k = clip(r_t^k, 1-\varepsilon^k, 1+\varepsilon^k)$表示时间步*t*处 PPO 的截断比率。ˆ A_t^k 是智能体在时间步*t*处的优势函数。 ε^k 和 η^k 分别是智能体的原始截断和双重截断 PPO 超参数，且 $k \in c, m$。

智能体的价值网络目标函数形式如下：

$$L_v^k = \hat{\mathbb{E}}_t[(V^k(\tilde{s}_t^k) - G_t^k)^2], k \in c, m \tag{3}$$

其中， $G_t^k = V_{old}^k(\tilde{s}_t^k) + \hat{A}_t^k$表示智能体的回报，而$V_{old}^k$由旧价值函数计算得出。

特征处理

游戏的原始数据可通过游戏提供的数据接口获取。随后对数据进行处理，生成小图像和向量信息，作为网络输入。小图像的形状为 $72 \times 48 \times 8$，包含 8 个通道，72×48 表示重塑后的尺寸。小图像的通道内容如表 1 所示。交互实体的位置按照各类实体的重要性顺序绘制在2^{th} - 6th 通道上。

向量信息包括控制中的智能体、队友、对手、全局状态、交互实体和记忆信息的内容，如表 2 所示。队友信息仅对老鼠智能体可用，而对手信息是最近一次观察到的对手信息。主要的交互实体包括奶酪、老鼠洞、陷阱、物品等。记忆信息

通道	内容
1th	地形图
2th - 6th	交互实体的位置
7th	猫智能体的位置
8th	鼠标智能体的位置

表1：小图像的通道内容。

记录了许多重要的历史消息，例如各种实体的历史动作和历史位置。

网络架构

该网络主要由三个部分组成：状态编码器、策略网络和价值网络。图2展示了网络架构的整体概览。在状态编码器中，小图像和向量信息分别输入残差网络 [He 等人,2016] 和多层感知机（MLP）进行处理。随后，将获得的隐藏状态进行拼接，以获得编码状态，并将其传递至中间多层感知机（Mid-MLP），从而提取瓶颈信息。最后，将瓶颈信息传递至策略网络和价值网络，这两个网络同样采用多层感知机结构。此外，还引入不可见信息来训练价值网络，以提高价值估计的准确性。我们定义了18种动作和16个动作方向来控制智能体的行为，其中，动作方向用于施放某些技能或使用某些物品。网络的详细信息见附录 B。

我们的网络结构没有使用循环神经网络（RNN）来存储历史信息，因为记忆信息的嵌入可以使模型在不使用 RNN 的情况下记住重要的历史信息。此前的一些探索表明，使用记忆信息代替长短期记忆网络（LSTM）可以加快训练过程，并减少模型参数的数量。

非对称奖励塑形

完成游戏中特定任务的智能体将获得奖励。由于非对称游戏的特点，需要设计两套奖励塑形。此外，

类型	内容
控制智能体 队友智能体 对手智能体	位置、生命值、角色等级、技能等级、技能冷却时间、增益效果或减益效果、持有物品等。
全局信息	游戏阶段、时间、进度等
交互实体	位置、生命值、与玩家的距离、主要状态等
记忆信息	重要历史信息

生命值（HP）：生命值；CD：冷却时间。

表 2：向量信息的内容。

奖励塑形过程应有助于平衡两种不同人工智能之间的胜率，以维持均衡的历史模型池。塑形奖励分为三种形式：普通奖励、引导奖励和团队奖励。在训练过程中，部分引导奖励会逐渐衰减，以防止智能体对某些固定策略过拟合。其他引导奖励仅在每个回合首次达成目标时发放，以避免无用的重复操作。完成某些重要任务后，鼠方智能体将获得团队奖励。所设计奖励的详情见附录 C。

3.3 非对称进化训练（AET）

自我对弈是在训练过程中提升对称竞技场游戏人工智能水平的一种不可或缺的方式。通过与当前自我模型以及训练过程中生成的历史自我模型进行对战，可以快速发现训练模型的弱点。击败具有不同策略的各种模型，能够使人工智能具备更强的泛化能力和更优的性能。

对于 AMP 游戏，双方是完全不同的人工智能，具有不同的规则、目标、玩家数量等。一方人工智能的性能提升取决于双方整体水平和多样性的提升。因为如果一方人工智能的性能停滞不前，就可能出现过拟合现象。因此，AMP 游戏中的两种不同人工智能必须同时进行训练和更新，以确保双方的人工智能都能得到提升。基于上述原因，我们提出了 AET，同时训练猫鼠双方智能体，如图 3 所示。两类智能体以 80% 的概率与最新的对手模型对战，以 20% 的概率与历史模型进行竞争。最新模型持续与历史模型对战，以避免陷入非传递性局面 [Gleave 等人， 2019] 并留下策略漏洞。我们使用 pFSP [Vinyals 等人， 2019] 来计算从模型池中选择历史模型的概率，模型a 与模型b 对战的概率记为

$$P_a^b = \frac{f(d_a^b)}{\sum_{c \subseteq C} f(d_a^c)} \tag{4}$$

其中，d_a^b 表示模型 a 击败模型b的胜率，模型 b 从对手模型池C中选取。 c 表示对手模型池 C中的模型。

$f(x) = (1-x)^p$ 是加权函数，用于控制胜率的重要性；在本研究中， p 设为 1。

当一个新模型被添加到历史模型池（HMP）时，每个模型都会与对手 HMP 中的所有模型进行 n 次对战，作为评估过程，以获得所有模型新的平均胜率。本研究中，HMP 的规模设定为 500，这是根据资源配置确定的。如果智能体的采样器数量约为 12,500，且其中 20% 的采样器加载 HMP 中的模型，那么将 HMP 的规模设定为 500 后，每个模型就可以加载到约 5 个采样器上。因此，在这种配置下，经验回放缓冲区基本上能够包含所有历史模型的数据，从而确保对手多样性。

在 AET 期间，AMP 游戏的特性会导致双方智能体的胜率失衡。这种现象会造成两类智能体的策略不稳定和性能失衡，导致双方的人工智能水平都陷入停滞。首先，在资源相同的情况下，由于游戏动态不平衡，胜率较高的一方会使每个智能体获得更多“正向”轨迹，而另一方则会有更多“失败”轨迹，这会促使较弱一方进行更多探索以改变局面。然而，在找到击败当前对手的方法之前，探索通常会带来更多失败。经典的自我博弈方法可以避免这个问题。如果当前模型在训练期间性能下降，对手的最新模型也会随之变弱。接下来，当前模型会持续挑战自我模型和固定的历史模型，以弥补不足。通过进一步训练和持续探索，当前模型最终将击败模型池，并取得更好的性能。但在 AET 中，对手不会停止进步并等待当前模型变强。其次，当一方性能更好时，它在更多情况下遭遇不利局面的可能性较小。较强的一方会逐渐忘记如何应对劣势局面，导致其在不利状态下表现不佳。

为了解决这两个问题，我们采用了自适应数据调整（ADA）和环境随机化（ER）。在我们的环境中，训练老鼠人工智能比训练猫 AI 困难得多，因为老鼠人工智能是一个多智能体模型，需要复杂的协作才能制定出成功的策略。此外，在训练初期，鼠方智能体的胜率几乎为零，因为鼠方智能体很难通过随机探索在规定时间内完成任务。因此，老鼠人工智能在训练初期需要更多的采样器资源

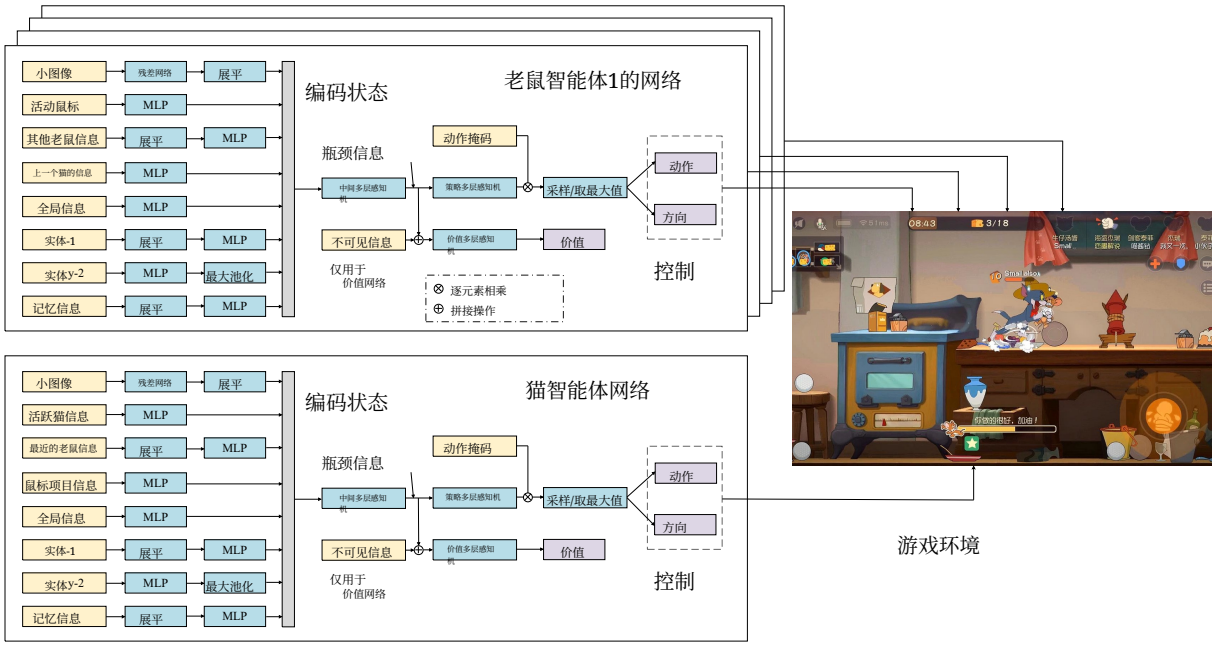

图2：网络架构概览。老鼠智能体和猫智能体的输入各不相同。主动信息概括了控制智能体的状态。其他老鼠信息表示其他老鼠能够观测到的大部分信息。上次状态信息记录了对手的上一状态。实体-1表示部分实体的信息，实体-2则包含数量可变的实体信息。实体-2通过多层感知机和最大池化层进行处理。记忆信息将通过一个多层感知机进行嵌入。策略多层感知机的输出将与动作掩码相乘，再通过Softmax层获得动作概率。动作掩码根据当前状态提取。最后，通过概率分布采样或argmax函数选择动作及动作方向。在残差网络和多层感知机中，均使用ReLU作为激活函数。

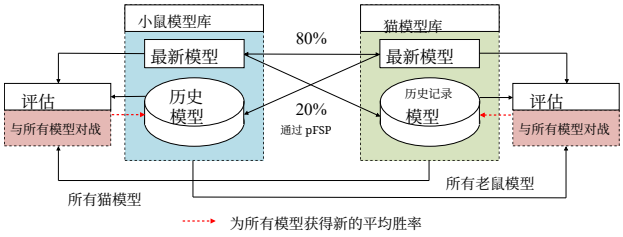

图 3：AET 概述

。但是，随着训练进度的推进，鼠方智能体经历的回合数多于猫智能体，这使老鼠获得了更多探索机会。在这种情况下，老鼠将逐渐摆脱劣势，而猫的胜率会迅速下降。因此，采用固定比例的采样器资源很难维持相对平衡的胜率。为此，自适应数据调整（ADA）旨在根据当前模型的评估结果动态调整采样器资源。例如，当猫的胜率高于老鼠时，采样器资源将向老鼠倾斜，直到老鼠人工智能通过更多探索获得相对平衡的胜率。鼠方智能体的采样器资源自适应比例可表示为：

$$r^m = min(\alpha, max(\beta, r^m_{old} + dif)) \tag{5}$$

$$dif = \begin{cases} 0.25(w^c - w^m), & \text{if } w^c \geq w^m \\ w^c - w^m, & \text{otherwise} \end{cases} \tag{6}$$

其中，α 和 β 分别是该比率的上限和下限。r^m_{old} 表示老鼠智能体采样器资源的旧比率。w^c 和 w^m 分别是猫智能体和老鼠智能体的胜率。α 和 β 的取值取决于游戏双方的不平衡特性；在本研究中，它们分别设为 0.8 和 0.5。由于每场游戏中老鼠智能体的数量是猫智能体的四倍，将 α 设为 0.8 可以确保老鼠智能体获得的资源不超过猫智能体资源的四倍。将 β 设为 0.5，可确保较难训练的一方（老鼠智能体）获得比易训练的一方（猫智能体）更多的资源。采样器资源的新比率 r^m 将按设定的时间间隔更新。实际应用中，采样器由采用 ADA 的训练框架控制，并选择性地训练猫模型或老鼠模型。该框架会持续重新分配资源，直至达到目标比率。

ER 用于扩展探索情形、稳定训练过程，并通过让强势方遭遇更多不利局面来平衡双方的胜率。当双方胜率持续一段时间失衡时，每个游戏回合都有 45% 的概率随机提升弱势方的角色等级，并有 35% 的概率让强势方遭遇一些随机的困难情形，例如一名队友被淘汰。此外，每个回合还有 15% 的概率

让弱势方在游戏开始时暂时得到强化。ER 的这些情形是通过分析游戏数据、观察失败案例并咨询职业玩家设计的。这种方法会持续发挥作用，直到双方胜率趋于平衡。它可以确保强势方接受全面训练并应对各种情况。借助 ADA 和 ER，双方能够动态维持相对平衡的局面，从而实现持续进化的目标。

4 实验

整个训练过程耗费了138块 Nvidia 2080Ti GPU 和 55,000个 CPU。猫模型和老鼠模型的参数量分别为437万和489万。正如上一节所述，老鼠智能体与猫智能体之间的资源分配会根据实际情况进行实时调整。

4.1 与人类玩家对局的结果

人工智能模型部署在游戏服务器上，每个游戏客户端都会向服务器发送动作请求。客户端与服务器之间的通信是异步的，即如果当前帧未收到服务器发来的新动作，客户端就会执行上一个动作。游戏帧率（FPS）为 15，时间延迟设为 3 帧。客户端每隔 2 帧请求一次动作，中间帧会继续执行上一个动作。因此，平均响应时间为 $(3+4)/2/15 = 0.233$s。由于网络通信波动，通过部署测试得出的实际平均响应时间为 0.252s。

游戏中共有六个段位：青铜、白银、黄金、铂金、钻石和大师，分别代表排名前 100%、70%、45%、20%、5% 和 1% 的玩家。我们邀请了 20 名玩家挑战我们的人工智能，其中包括 5 名大师段位玩家、8 名钻石段位玩家和 7 名铂金段位玩家。这 5 名大师段位玩家均为游戏评测员，其中一人曾以排名第一的成绩赢得冠军赛。为了避免人类玩家临时组队所产生的影响，我们让这些人类玩家控制猫，与我们的老鼠人工智能对战。人类玩家可以从 汤姆、闪电、保镖汤姆、塔拉 和 剑客汤姆 中选择一个猫角色。总共进行了 65 场游戏，我们的老鼠人工智能赢得了其中 64 场，胜率达到 98.5%。我们的人工智能与 20 名职业玩家进行了 20 场游戏，并赢得了其中 19 场。这些职业玩家平均每回合获得 1619.2 分，抓获 4.05 只老鼠，并淘汰 0.65 只老鼠。人工智能在与另外 15 名玩家进行的剩余 45 场游戏中获胜，这些玩家平均每回合获得 856.6 分，抓获 1.62 只老鼠，并淘汰 0.11 只老鼠。通过测试，我们的人工智能在不同的不利情况下均展现出了鲁棒性能，尤其是在与这些职业玩家对战时。职业玩家认为，该人工智能在大师段位中处于顶尖水平，并拥有多样化的策略和出色的类人行为。我们的智能体与顶尖人类玩家进行的大部分游戏回合均可通过以下链接观看：https://sourl.cn/fjhGMG，其中采用了人类玩家的观察视角。

4.2 训练过程

模型首先经过训练以击败最高难度机器人，从而获得基础模型。最高难度机器人是游戏中内置的最高级别人工智能，由游戏工作室精心设计了行为树 [Iovino 等人，2022] 。据游戏工作室报告，最高难度机器人的段位约为初级黄金段位。从零开始训练得到基础模型需要 7 天，其对最高难度机器人的胜率可超过 95%。通过对基础模型进行微调，开展了多个分支版本的训练。经探索发现，引入 AET、ADA、ER，以及结合大规模训练进行角色扩展，是重要的改进节点。结合大规模训练进行角色扩展是最后阶段，该阶段增加了更多对手角色和训练资源。所提出的人工智能的训练过程如图 4 所示。如上所述，最终版本的老鼠人工智能对战人类玩家的胜率可达 98.5%，而最终版本的猫 AI 对战最终版本的老鼠人工智能的胜率可超过 40%。我们使用 TrueSkill 分数 [Herbrich 等人，2006] 作为评估模型性能的指标。假设猫和老鼠的最高难度机器人具有相同的 TrueSkill 分数。老鼠人工智能使用固定阵容进行训练，该阵容包括侦探杰瑞、罗宾汉杰瑞、国王杰瑞和泰菲。训练超参数见附录 B，这些参数是在项目早期通过一系列搜索测试选定的。由于超参数组合数量庞大，我们无法保证这组超参数是最优选择，但可以保证在类似应用中取得良好效果。训练过程使用 Python 3.6 和 PyTorch 1.6 实现。

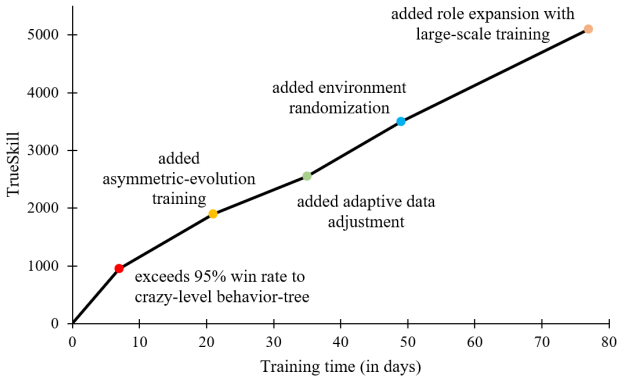

图 4：训练过程。

4.3 消融实验

进行了两组消融实验，以确定所提出方法中关键组件的作用。如第 3.3 节所述，ADA 和 ER 是 AET 中的两个关键组件。此外，训练过程中历史模型的规模也十分重要。因此，第一

[3]这里的“大规模”是指使用更多图形处理器和中央处理器资源，以及更大的相关参数，例如批量大小。

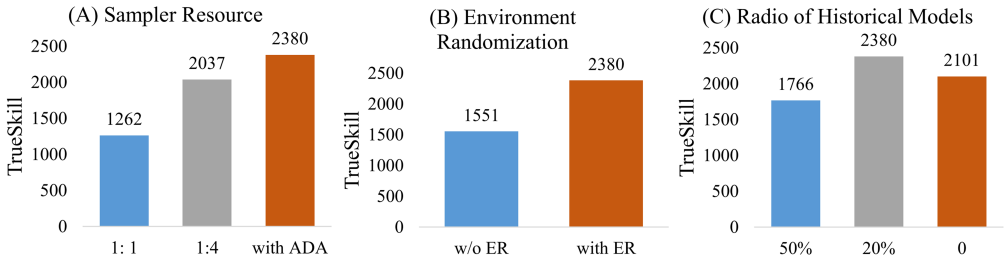

图 5：AET 中关键组件消融测试的 TrueSkill 结果。（A）比较猫与老鼠智能体之间不同的采样器资源分配方案（1:1、1:4、使用 ADA）；（B）比较使用与不使用 ER 的效果；（C）比较历史模型的不同占比（50%、20%、0）。

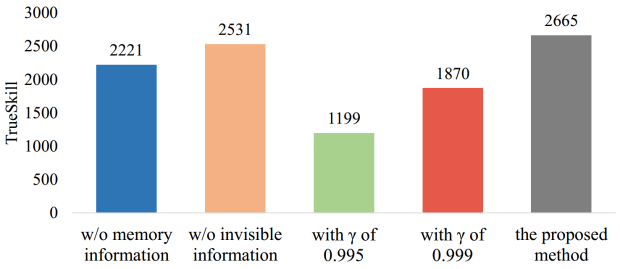

图 6：学习系统中采用的不同因素的 TrueSkill 结果。

一组实验进行了三项消融测试，以验证这些组件的有效性。测试以老鼠和猫智能体的基础模型作为起始模型，这些模型是通过对抗最高难度机器人训练得到的收敛模型。训练 7 天后老鼠人工智能的 TrueSkill 值被用作评估指标。图 5（A）表明，使用 ADA 进行训练有助于提升人工智能性能，而保持资源平衡是最不适合我们这种非对称环境的选择。换句话说，在对称环境中训练过程的成功经验无法直接用于 AMP 环境。如图 5（B）所示，ER 能显著提高学习效果。这表明，增加样本多样性，尤其是困难场景的样本，有利于提升模型的性能和泛化能力。图 5（C）表明，使用 20% 的历史模型进行训练，效果优于使用 50% 的历史模型或完全不使用历史模型。原因可能是，与固定权重的历史模型进行大量对战，会使训练模型过拟合于某些固定策略，从而轻松获得正奖励，同时也会减缓模型的改进。适当增加与历史模型对战的比例，则可通过提升样本多样性来改善模型性能。从第一组实验结果可以看出，传统的自我对弈设置（不使用 ADA 和 ER）无法很好地适用于非对称环境。

在第二组实验中，我们比较了学习系统中采用的三个因素，包括记忆信息、价值网络的不可见信息，以及折扣因子 γ的选择。我们使用不同设置训练了五个老鼠人工智能，使其与猫模型对战 7 天；猫模型则采用本文提出的方法进行训练。图 6 展示了不同版本老鼠人工智能之间的 TrueSkill 分数。结果表明，使用记忆信息能够提升训练效果。不可见信息通过改进价值估计，有助于训练。参数 γ 也是另一个重要因素：使用 0.995 和 0.999 时的 TrueSkill 分数远低于本文提出的方法，其 γ 设为 0.9995。这表明，在这款游戏中，动作可能会对策略产生长期影响。每项消融实验的训练过程都使用了 4000 个采样器。

5 结论、局限性与未来工作

本文提出了一种采用非对称进化训练方法的多智能体深度强化学习框架。我们设计了自适应数据调整和环境随机化，以解决训练过程中的不平衡问题。我们首次证明，基于强化学习的智能体无需使用任何人类数据，便能在复杂的非对称多人游戏中达到顶尖的人类水平表现。消融实验表明，所设计的模块有助于训练过程。然而，我们的工作仍存在一些局限。与其他已达到顶尖人类玩家水平的人工智能类似，训练过程中也融入了与环境相关的人类专业知识，例如奖励塑形。未来，我们计划构建内在奖励 [Zheng 等人，2020] 以取代复杂的奖励塑形过程。此外，猫和老鼠 有许多角色，但由于计算成本高昂，我们只训练了其中部分角色。因此，未来我们将让训练出的人工智能支持游戏中的全部角色，并举办一场让猫 AI 挑战顶尖人类玩家的比赛。此外，训练过程中使用的超参数可能并非适用于其他环境的最佳选择，因此开展新的研究时仍需进行调整。目前，还没有具备复杂多样策略的开源非对称多人游戏环境，因此我们的研究是在专有游戏环境中进行测试的。接下来，我们将尝试开发一个开源的非对称多人游戏测试平台，并将我们的非对称训练框架开源，以训练更多人工智能。

References

[Baker 等人， 2019] Bowen Baker、Ingmar Kanitscheider、Todor Markov、Yi Wu、Glenn Powell、Bob McGrew 和 Igor Mordatch。多智能体自动课程中涌现的工具使用。载于《国际学习表征会议》，2019年。[Balduzzi 等人， 2019] David Balduzzi、Marta Garnelo、Yoram Bachrach、Wojciech Czarnecki、Julien Perolat、Max Jaderberg 和 Thore Graepel。对称零和博弈中的开放式学习。载于《国际机器学习会议》，第434–443页。PMLR，2019年。[Bansal 等人， 2018] Trapit Bansal、Jakub Pachocki、Szymon Sidor、Ilya Sutskever 和 Igor Mordatch。通过多智能体竞争涌现复杂性。载于《国际学习表征会议》，2018年。[Berner 等人， 2019] Christopher Berner、Greg Brockman、Brooke Chan、Vicki Cheung、Przemysław Dkebiak、Christy Dennison、David Farhi、Quirin Fischer、Shariq Hashme、Chris Hesse 等人。基于大规模深度强化学习的 Dota 2。arXiv 预印本 arXiv:1912.06680，2019年。[Bycer， 2019] Josh Bycer。非对称游戏设计。
h
t
t
p
s://
m
edium.com/super-jump/asymmetrical-game-design-2d3ccbc2b4ab，[在线], 2019。[Coulom， 2006] R´emi Coulom。蒙特卡洛树搜索中的高效选择与回传算子。载于《国际计算机与游戏会议》，第72–83页。施普林格，2006年。[Gleave 等人， 2019] Adam Gleave、Michael Dennis、Cody Wild、Neel Kant、Sergey Levine 和 Stuart Russell。对抗策略：攻击深度强化学习。载于《国际学习表征会议》，2019年。[Harry， 2021] Alston Harry。十款最佳非对称多人游戏排名。
https://www.thega
mer.com/best-asymmetrical-multiplayer-games/，[在线], 2021。[He 等人， 2016] 何恺明、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。[Herbrich 等人， 2006] Ralf Herbrich、Tom Minka 和 Thore Graepel。TrueSkillTM：一种贝叶斯技能评级系统。《神经信息处理系统进展》，第19卷，2006年。[Iovino 等人， 2022] Matteo Iovino、Edvards Scukins、¨Jonathan Styrud、Petter Ogren 和 Christian Smith。机器人学与人工智能中的行为树综述。《机器人与自主系统》，154：104096，2022年。[Jaderberg 等人， 2019] Max Jaderberg、Wojciech M Czarnecki、Iain Dunning、Luke Marris、Guy Lever、Antonio Garcia Castaneda、Charles Beattie、Neil C Rabinowitz、Ari S Morcos、Avraham Ruderman 等人。基于种群强化学习的3D多人游戏人类水平表现。《科学》，364(6443)：859–865，2019年。

[Konda和Tsitsiklis， 1999] Vijay Konda和John Tsitsiklis。演员-评论家算法。《神经信息处理系统进展》，12，1999年。[Kurach等， 2020] Karol Kurach、Anton Raichuk、Piotr Stanczyk、Michał Zajkac、Olivier Bachem、Lasse Espeholt、Carlos Riquelme、Damien Vincent、Marcin Michalski、Olivier Bousquet等。Google Research Football：一种新型强化学习环境。载于《AAAI人工智能会议论文集》，第34卷，第4501–4510页，2020年。[Li等， 2020] Junjie Li、Sotetsu Koyamada、Qiwei Ye、Guoqing Liu、Chao Wang、Ruihan Yang、Li Zhao、Tao Qin、Tie-Yan Liu和Hsiao-Wuen Hon。Suphx：使用深度强化学习掌握麻将。arXiv预印本arXiv:2003.13590，2020年。[Mnih等， 2016] Volodymyr Mnih、Adria Puigdomenech Badia、Mehdi Mirza、Alex Graves、Timothy Lillicrap、Tim Harley、David Silver和Koray Kavukcuoglu。用于深度强化学习的异步方法。载于《国际机器学习会议》，第1928–1937页。PMLR，2016年。[Schulman等， 2015] John Schulman、Philipp Moritz、Sergey Levine、Michael Jordan和Pieter Abbeel。使用广义优势估计进行高维连续控制。arXiv预印本 arXiv:1506.02438，2015年。[Schulman等， 2017] 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。[Silver等， 2016] David Silver、Aja Huang、Chris J Maddison、Arthur Guez、Laurent Sifre、George Van Den Driessche、Julian Schrittwieser、Ioannis Antonoglou、Veda Panneershelvam、Marc Lanctot等。使用深度神经网络和树搜索掌握围棋。《自然》，529(7587):484–489，2016年。[Silver等， 2018] David Silver、Thomas Hubert、Julian Schrittwieser、Ioannis Antonoglou、Matthew Lai、Arthur Guez、Marc Lanctot、Laurent Sifre、Dharshan Kumaran、Thore Graepel等。通过自我对弈掌握国际象棋、将棋和围棋的通用强化学习算法。《科学》，362(6419):1140–1144，2018年。[Trott等， 2019] Alexander Trott、Stephan Zheng、Caiming Xiong和Richard Socher。保持距离：使用自平衡塑形奖励解决稀疏奖励任务。《神经信息处理系统进展》，32，2019年。[Vinyals等， 2019] Oriol Vinyals、Igor Babuschkin、Wojciech M Czarnecki、Micha¨el 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》宗师级水平。《自然》，575(7782):350–354，2019年。[Ye等，2020a] Deheng Ye、Guibin Chen、Wen Zhang、Sheng Chen、Bo Yuan、Bo Liu、Jia Chen、Zhao Liu、Fuhao Qiu、Hongsheng Yu等。迈向完整的...

使用深度强化学习的多人在线战术竞技游戏。神经信息处理系统进展, 33:621–632, 2020年。

[Ye 等，2020b] Deheng Ye、Zhao Liu、Mingfei Sun、Bei Shi、Peilin Zhao、Hao Wu、Hongsheng Yu、Shaojie Yang、Xipeng Wu、Qingwei Guo 等。通过深度强化学习掌握多人在线战术竞技游戏中的复杂控制。在*AAAI*人工智能会议论文集，第34卷，第6672–6679页，2020年。

[Zheng 等， 2020] Zeyu Zheng、Junhyuk Oh、Matteo Hessel、Zhongwen Xu、Manuel Kroiss、Hado Van Hasselt、David Silver 和 Satinder Singh。学习到的内在奖励能捕捉到什么？载于国际机器学习会议，第11436–11446页。PMLR，2020年。

游戏介绍：猫和老鼠

猫和老鼠 是一款典型的1对4非对称多人在线游戏。游戏双方拥有不同的资源、规则、基础能力和最终目标。玩家将在由杂物间、餐厅、客厅、卧室、院子、厨房、阁楼及其他房间组成的场景中进行游戏。四名鼠玩家和一名猫玩家将在场景中展开一系列复杂互动，以完成各自的任务。游戏开始时会生成游戏场景，各房间的位置随机分布。场景中会随机生成五个老鼠洞和五块奶酪。老鼠队需要合作，将五块奶酪推入场景中的老鼠洞，然后通过墙壁裂缝逃脱。每个老鼠洞只能塞入一块奶酪。所有奶酪都被推入老鼠洞后，场景中的某处会随机生成一条作为逃生通道的墙壁裂缝。两只或更多老鼠成功逃脱，即老鼠获胜。游戏过程中，老鼠可能被猫抓住并绑到火箭上。如果在一定时间内没有队友前来营救，被绑在火箭上的老鼠就会被淘汰。猫独自作战，阻止老鼠取得奶酪并逃脱。若猫成功淘汰三只老鼠，或阻止老鼠在规定时间内推入五块奶酪，或阻止三只老鼠逃脱，即可获胜。

老鼠智能体或猫智能体拥有一个主动技能、一个被动技能以及一到两个武器技能。这些技能具有不同的功能，有助于实现最终目标。游戏中有大量不同类型的物品可供获取和使用。此外，非玩家角色（NPC）也会影响游戏进度，例如，场景中的狗会无差别地攻击附近的玩家。由于场景中的几乎所有重要元素（奶酪、老鼠洞、物品等）都是随机生成的，因此游戏设有一个较短的准备阶段。在准备阶段，每位老鼠玩家都可以控制一辆侦察车来观察场景，并根据场景布局制定团队策略，而猫则可以寻找并摧毁侦察车，以提升角色等级。准备阶段结束后，游戏正式开始。这些特性极大地提高了游戏的复杂性、可玩性和趣味性。在图 7 中，展示了汤姆与杰瑞在此作为示例。在该环境中，动作空间为 ~10e9712（144 个离散化动作，每局帧数为4,500）。状态空间为 ~10 e1769（3,456 个离散化位-置，5 个智能体， 100+每个智能体的状态）。猫和老鼠 以 15 FPS运行超过 5 分钟，因此回合长度超过 4,500 步，并且智能体平均每 150 步获得一次奖励。

B 网络架构和训练超参数详情

表 3 列出了所设计网络架构的详细信息。策略多层感知机中的 (18, 16) 表示它有两个策略头，分别对应动作和动作方向。表 4 列出了训练超参数的详细信息。样本重用率是一个浮点值，因为训练框架中存在通信波动，这可能导致训练过程中数据传输时间发生变化。

网络结构	Size
残差网络	3 个残差块，每个块的输出通道数为 32 , 64, 128
状态编码器中的多层感知机	256 × 128
中间多层感知机	1280 × 1024
策略多层感知机	1024 × 512 × 256 ×(18, 16)
价值多层感知机	1024 × 512 × 256 × 1

表 3：网络架构的详细信息。

超参数	价值
帧跳跃	2
批量大小	每个图形处理器 7000
轨迹长度	128
样本重用	1.0 ∼ 1.2
PPO裁剪	0.2
PPO双重裁剪	3
折扣因子 γ	0.9995
GAE折扣因子 λ	0.95
价值损失权重	0.5
熵系数	0.01
优化器	Adam
学习率	5e-5
Adam β_1, β_2	0.99，0.999

表 4：训练超参数。

C 奖励塑形的详细信息

表 5 和表 6 分别列出了猫智能体和老鼠智能体的奖励。这些奖励旨在鼓励智能体完成重要任务。引导奖励用于防止智能体通过轻松完成任务来“钻奖励的空子”。

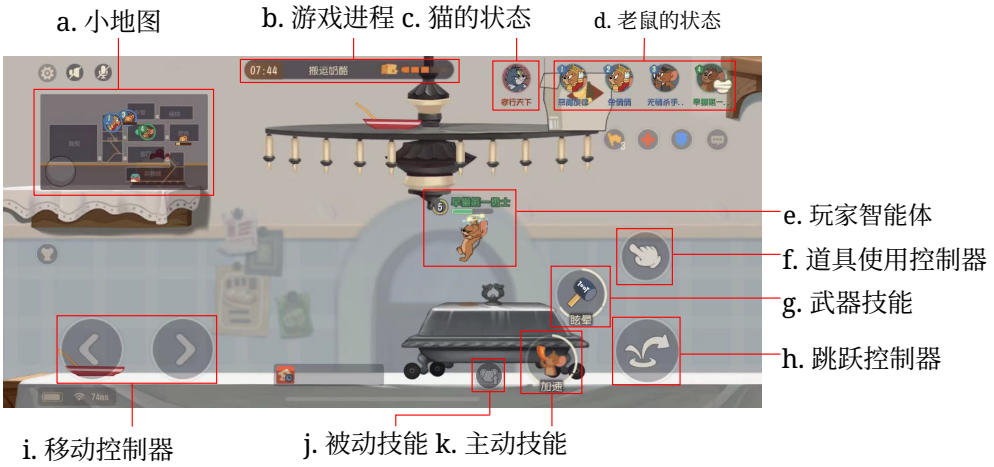

图 7：猫和老鼠的游戏用户界面。(a)-(d) 展示全局游戏信息，(e) 为玩家控制的老鼠智能体，(f)-(k) 为控制按钮。

任务	价值	操作指南
放置捕鼠器	0.025	
抓住老鼠	0.1	
把老鼠绑在火箭上	0.1	
消灭一只老鼠	2	
奶酪被推入老鼠洞	-0.25	
墙壁裂缝的生命值变化	0.01 × CV	
胜利 / 失败	5 / -5	
首次寻找火箭	0.05	√
首次找到黄金洞穴	0.05	√
首次对老鼠造成伤害	0.15	√
摧毁侦察车（准备阶段）	0.5	
失去／夺回蛋糕（准备阶段）	-1 / 1	

HP：生命值；CV：变化值。

表 5：猫智能体的奖励列表。

任务	价值	指导	Team
到最近奶酪的距离	0.001 × 距离 × 衰减系数（从 5 到 1）	√	
到最近被火箭绑住的老鼠的距离	0.01 ×距离 × 衰减系数（从 3 到 1）	√	
抬起奶酪时与最近老鼠洞之间的距离	0.001 × 距离 ×衰减系数（从 5 到 1）	√	
到墙壁裂缝的距离	0.001 × 距离 × 衰减系数（从 5 到 1）	√	
捡起奶酪	0.05 × 系数（从 1 到 0）	√	
生命值变化	0.003 × CV		
被猫抓住了	-0.2		
被火箭淘汰	-2		
从猫或火箭手中救出队友	0.5		
在爆炸物即将爆炸时投掷爆炸物。	0.15		
摧毁关键位置的手套	0.15		
摧毁关键位置的捕鼠夹	0.15		
推奶酪	1 × 进度 / NMPC		
把奶酪推入老鼠洞。	0.5		√
破坏墙壁裂缝	2 × 伤害百分比		
打开墙壁裂缝	1		√
胜利 / 失败	5 / -5		√
侦察车被摧毁（准备阶段）	-0.5		
丢失 / 夺回蛋糕（准备阶段）	1 / -1		

DI：距离；DC：衰减系数；Coef：系数；NMPC：推动奶酪的老鼠数量；DP：伤害百分比。

表 6：老鼠智能体的奖励列表。部分引导奖励将在三天内逐渐衰减；完成相应任务时，所有鼠方智能体都将获得团队奖励。