arXiv:2301.11857v2 [cs.AI] 6 Feb 2023

# 策略-价值对 齐与基于搜索的多智能体学习中的鲁棒性

**Niko A. Grupen**[1] , **Michael Hanlon**[1] , **Alexis Hao**[1] , **Daniel D. Lee**[1] , **Bart Selman**[1]康奈尔大学

**摘要**

结合搜索与学习的大规模人工智能系统在博弈中已达到超人类水平的表现，但也被证明会以令人意外的方式失败。这类模型的脆弱性限制了其在现实世界部署中的有效性和可信赖性。在这项工作中，我们系统地研究了其中一种算法——AlphaZero，并识别出两个与探索本质相关的现象。首先，我们发现了策略-价值失配的证据——对于许多状态，AlphaZero 的策略与价值预测彼此矛盾，这揭示了 AlphaZero 的目标中准确落子选择与价值估计之间的张力。此外，我们发现 AlphaZero 的价值函数内部存在不一致性，尽管其策略采取最优策略，这仍导致其泛化能力较差。基于这些洞见，我们提出了 VISA-VIS：一种新颖方法，可提升 AlphaZero 中的策略-价值对齐和价值鲁棒性。实验表明，我们的方法可将策略-价值失配最多降低 76%，将价值泛化误差最多降低 50%，并将平均价值误差最多降低 55%。

## 1 引言

近年来，结合自我博弈强化学习与搜索的学习算法越来越受欢迎，因为它们能够在组合复杂环境中达到超越人类的性能水平 [Silver 等, 2017b]。最近，对 AlphaZero[ Silver 等, 2017a] 的研究发现，它遵循与对弈能力和网络容量相关的神经缩放定律 [Neumann and Gros，2022]——这进一步表明，与其他领域（例如大语言模型 [Kaplan 等，2020]）一样，混合算法仍有很大的扩展空间。随着这些算法不断扩展，以应对现实世界的复杂性，人们不禁要问，最先进的混合系统是否已准备好在现实世界中部署。

迄今为止，研究人员和从业者对支撑这些混合算法的神经网络为何会做出这样的决策，仍知之甚少——例如 AlphaGo 对阵李世石 [Metz，2016]中的著名“第37手”——以及他们能从中了解到关于环境及其中其他智能体的哪些信息。尽管基于概念的可解释性领域已取得显著进展，事后分析表明，许多人类可理解的概念都能从 AlphaZero 的表征中准确回归 [McGrath 等人，2022]，但自我博弈学习仍留下了不透明的产物。例如，利用对抗性博弈的研究表明，AlphaZero 对对手行为的细微扰动出奇地敏感 [Lan 等人，2022]，而且可能被利用，输掉业余人类玩家轻易就能赢下的棋局 [Wang 等人，2022]。

在这项工作中，我们发现了两种与 AlphaZero 探索本质相关、可能是其已知失效模式根源的现象。首先，我们发现了一种新颖的涌现现象——**策略-价值失配**，即 AlphaZero 的策略与价值预测相互矛盾。策略-价值失配是 AlphaZero 损失函数的直接结果：该函数分别包含策略误差项和价值误差项，却没有对二者的一致性作出约束。我们提出了策略-价值失配的信息论定义，并展示了如何对任意 AlphaZero 网络进行测量。接下来，我们发现了证据，表明**AlphaZero 的价值函数与其训练环境中的对称性之间存在不一致**。因此，AlphaZero 的价值预测对访问频率较低的状态和未见状态的泛化能力出奇地差。

关键的是，我们表明这些**现象会被搜索掩盖**：即使 AlphaZero 在其搜索过程中采取最优策略，这两种现象也可能同时存在。因此，这项工作揭示了一种有趣的张力：搜索使 AlphaZero 得以达到前所未有的规模和性能，但也可能掩盖其学习组件（即策略和价值网络）的缺陷。

为解决这些问题，我们提出对 AlphaZero 进行两项改进，直接针对策略-价值对齐和价值函数一致性。首先，我们引入价值引导选择（VIS）：一种改进的动作选择规则，使用 AlphaZero 的价值网络进行单步前瞻来计算动作概率。随后，VIS 会以随机方式交替采用这一价值动作选择规则和 AlphaZero 的标准策略（该策略由搜索概率得出）。在训练过程中，这会迫使 AlphaZero 的策略与价值预测保持一致。接下来，我们

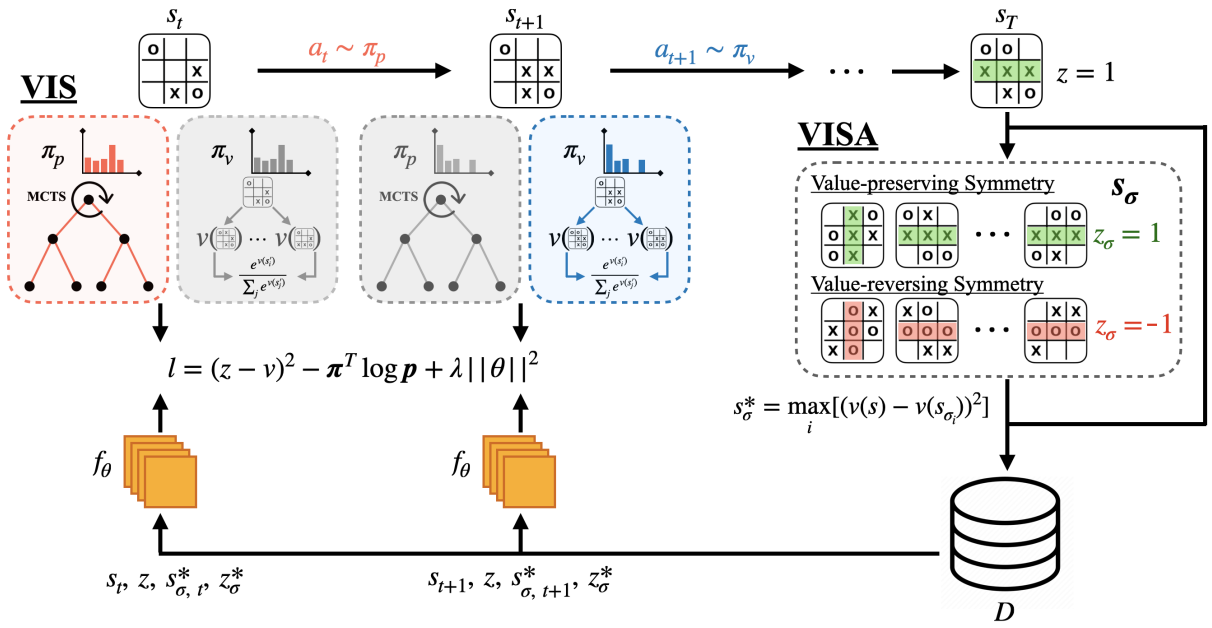


图1：我们提出的方法概览。VISA-VIS结合了对AlphaZero算法的两项改进：(i) **价值引导选择（VIS）**，动作选择以随机交替的方式进行：从蒙特卡洛树搜索的搜索概率（$a_t \sim \pi_p$）中采样动作，以及从由AlphaZero的价值函数（$a_t \sim \pi_v$）推导出的一组动作概率中采样；(ii) **价值引导对称增强（VISA）**，一种针对AlphaZero价值函数不确定性的数据增强形式。具体来说，在将一个状态$s_t$添加到经验回放缓冲区 $D$之前，VISA会生成一组对称状态变换 $\boldsymbol{s_\sigma}$，用价值函数评估每个变换后的状态 $s_{\sigma_i}$，并返回与原始状态 $s_t$的AlphaZero价值估计差异最大的状态 $s^*_\sigma$。 $s_t$ 和 $s^*_\sigma$ 都会被添加到 $D$。

引入价值引导对称增强（VISA）：一种考虑 AlphaZero 价值网络不确定性的数据增强技术。与那些不加甄别地向网络的经验回放缓冲区添加状态的数据增强技术不同，VISA 会考察给定状态的对称变换，并且只存储变换后的状态中其价值与网络对原始未变换状态的价值估计差异最大的那个。

VISA和VIS能够同时提升AlphaZero的策略-价值对齐和价值鲁棒性。我们将这种联合方法称为VISA-VIS。在多种对称双人零和博弈的实验中，我们表明，与AlphaZero相比，VISA-VIS可将策略-价值失配降低**最高76%**，将价值泛化误差降低**最高50%**，并将平均价值预测误差降低**最高55%**。

**贡献：** 总之，我们提出了五项主要贡献：

•我们系统地研究了 AlphaZero，并发现了两种被 AlphaZero 的搜索所掩盖的现象——策略-价值失配和价值函数不一致性。

• 我们定义了一种信息论度量，用于衡量策略-价值失配，可用于评估任何经过训练的 AlphaZero 网络。

• 我们提出了价值引导选择（VIS）：一种针对 AlphaZero 的改进型动作选择规则，能够强制策略与价值预测保持一致。

•我们提出价值引导对称增强（VISA）：一种利用对称性和价值函数不确定性来学习更加一致的值网络的数据增强技术。

• 我们提出了一种新方法 VISA-VIS，将 VIS 和 VISA 整合为一个扩展 AlphaZero 的单一算法。实验结果表明，与原版 AlphaZero 相比，VISA-VIS 可将策略-价值失配降低最高 76%，将价值泛化误差降低最高 50%，并将平均价值预测误差降低最高 55%。

## 2 相关工作

结合搜索与自博弈强化学习的混合算法持续取得成功，促使研究者关注其鲁棒性与可解释性。

**对抗样本** 在监督学习场景中，已有充分记录表明，合成的 [Goodfellow等人,2014] 以及自然产生的 [Hendrycks 等人，2021] 对抗样本都会使基于神经网络的分类器产生次优行为。近期，类似手段也被证明能有效找出强化学习智能体的弱点。例如，已有研究表明，对智能体观测进行微小扰动会导致奖励大幅下降[Huang 等人, 2017; Kos and Song，2017] 而利用对智能体行为的了解还会加剧其性能退化 [Lin 等人，2017]。

在多智能体场景中，强化学习智能体尤其容易受到环境中其他智能体的对抗行为影响 [Gleave等人， 2019]。尤其值得关注的是 Wang等人 [2022], 的研究，该研究表明，一名职业级

AlphaZeroGo 智能体可能会被简单的对抗性走法诱骗而输掉比赛，而业余人类玩家很容易就能应对这些走法。进一步的研究表明，AlphaZero容易受到其他状态和动作空间扰动的影响[Lan 等人， 2022]。我们认为，这些令人意外的失效模式正是本文所述现象的表现——策略-价值失配和价值函数不一致性。

**可解释性**强化学习可解释性方法分为两类：一类通过（i）引入内在可解释的策略与价值网络架构，提升智能体决策过程的透明度；另一类则为先前训练的网络生成事后解释。除了直接源自监督学习对应方法的传统途径（如显著性图[Greydanus等，2018年；Selvaraju 等， 2017]）之外，奖励无关行为分析 [Omidshafiei 等，2022]以及内在可解释策略学习方案[Grupen 等， 2022] 方面的进展，极大地提升了强化学习智能体决策过程的可解释性（且适用于多智能体场景）。

对于 AlphaZero 等大规模人工智能系统，关于事后、基于概念的可解释性的开创性研究表明，训练过程中，许多人类可理解的概念都编码在 AlphaZero 的策略网络中 [McGrath等，2022]。其他研究则采用模型探测与行为分析，理解 AlphaZero 在 Hex 游戏中所表征的概念 [Forde 等，2022]。这些研究凸显了我们工作中提出的搜索-学习困境——AlphaZero 能够学会复杂的对弈策略和精妙的人类可理解概念，同时其策略与价值预测却表现出不一致性。

## 3 背景

### 3.1 AlphaZero

AlphaZero 是一种结合了自博弈强化学习与蒙特卡洛树搜索（MCTS）的混合算法。下面我们将回顾 AlphaZero 的搜索和学习过程，因为这两者对于理解我们的方法至关重要。

**设置** AlphaZero 维护一棵搜索树，其中每个节点都是一个状态 $s$，每条边 $(s, a)$ 表示从 $s$采取动作 $a$。每条边由一组统计数据 $\{N(s, a), W(s, a), Q(s, a), P(s, a)\}$描述，分别记录该边的访问次数、动作价值总和、平均动作价值和选择概率。AlphaZero 使用一个具有参数 $\theta$ 的双头神经网络 $f_\theta$，按如下方式计算状态 $s$的策略概率 $\boldsymbol{p}$ 和价值估计 $v$：

$$(\boldsymbol{p}, v) = f_\theta(s) \tag{1}$$

蒙特卡洛树搜索使用 $f_\theta$ 在搜索过程中评估状态。

**搜索** 在每个状态 $s$下，AlphaZero 都会执行 $n$ 次蒙特卡洛树搜索模拟。在每次模拟中，蒙特卡洛树搜索都会遍历博弈树，选择使上置信界 $Q(s, a) + U(s, a)$ 最大的边，其中：

$$U(s,a) = cP(s,a)\frac{\sqrt{\sum_b N(s,b)}}{1+N(s,a)} \tag{2}$$

这会鼓励探索（搜索树中）访问次数较少的状态；而 $c$ 是用于确定探索程度的系数。当遇到叶节点 $s_L$ 时，会对其进行扩展，并由 AlphaZero 的网络进行评估，以生成 $(\boldsymbol{p}_L, v_L) \sim f_\theta(s_L)$，然后按如下方式初始化：

$$N(s,a) = 0, W(s,a) = 0, Q(s,a) = 0, P(s,a) = p_{L,a} \tag{3}$$

随后，在沿搜索树向上的反向传递过程中，按如下方式更新每条经过的边的统计数据：$N(s, a) = N(s, a) + 1$，$W(s, a) = W(s, a) + v_L$，$Q(s, a) = W(s, a)/N(s, a)$。

**动作选择**蒙特卡洛树搜索后，AlphaZero 构建一个策略 $\boldsymbol{\pi}$，其中，从状态 $s$选择动作 $a$ 的概率 $\pi(a|s)$ 直接根据访问次数计算：

$$\pi(a|s) = \frac{N(s,a)^{1/\tau}}{\sum_b N(s,b)^{1/\tau}} \tag{4}$$

而 $\tau$ 是控制探索的温度参数。在训练和测试时的推演过程中，AlphaZero 都会根据该策略采样动作 $a \sim \boldsymbol{\pi}$。

**学习**从初始状态 $s_0$（例如空棋盘）开始，AlphaZero 通过自我博弈进行训练，使用上述搜索过程选择动作，直到达到终止状态 $s_T$。根据游戏规则，为 $s_T$ 生成得分 $z$，并将轨迹中每个时间步 $t \in [0, T]$ 对应的元组 $(s_t, a_t, z)$ 存储到经验回放缓冲区 $D$ 中。AlphaZero 的神经网络经过更新，以最小化损失函数：

$$l = (z - v)^2 - \boldsymbol{\pi}^T \log \boldsymbol{p} + \lambda ||\theta||^2 \tag{5}$$

其中，$(\boldsymbol{p}, v) \sim f_\theta(s)$ 如公式 (1) 所示，$\boldsymbol{\pi}$ 由公式 (4) 得出，而 $\lambda$ 是对 $L2$ 正则化进行加权的系数。直观而言，公式 (5) 鼓励网络 (i) 最小化预测值 $v$与游戏结果 $z$之间的误差；以及 (ii) 最大化预测策略概率 $\boldsymbol{p}$ 与搜索概率 $\boldsymbol{\pi}$之间的相似度。

## 4 方法

本节介绍我们提出的 VISA-VIS 方法，该方法结合了对 AlphaZero 算法的两项改进：价值引导选择（VIS）和价值引导对称增强（VISA）。我们将在下面的小节中分别详细介绍这两项改进，并在图1中概述我们提出的方法。

### 4.1 价值引导选择

回顾公式 (5) 中 AlphaZero 的损失函数。尽管该目标鼓励分别准确预测策略和值，却没有对二者的一致性施加任何约束。因此，AlphaZero 的网络可能会输出与其价值预测相矛盾的策略概率 $\boldsymbol{p}$ ——例如，在推理过程中，$\boldsymbol{p}$ 为某个获胜动作分配了很大的概率，而 $v$ 却预测当前状态是必败状态。我们发现，这种情况在实践中经常发生。为了确保策略-价值一致性，我们提出了价值引导选择（VIS）。VIS 是一种新颖的动作选择规则，其中

AlphaZero 交替使用其策略（由公式 (4) 中的搜索概率引导）和价值函数来选择动作。

更形式化地，令 $\boldsymbol{s'}$ 表示从给定状态 $s$ 出发所能到达的后继状态集合，其中每个后继状态 $s'_a \in \boldsymbol{s'}$ 都是从 $s$ 执行某个合法动作 $a$ 后得到的。接着，我们将从 $s$ 进行一步价值前瞻所得到的动作概率定义为 $\pi_v$：

$$\pi_v(a|s) = \frac{e^{v(s'_a)}}{\sum\limits_{s'_b \in \boldsymbol{S'}} e^{v(s'_b)}} \tag{6}$$

其中，$v(s)$ 是 AlphaZero 的价值网络对状态 $s$ 的值预测。将公式（4）中定义的搜索策略重新命名为 $\pi_p$ 后，VIS 随即定义动作选择规则：

$$\pi_{\text{VIS}}(a|s) = \begin{cases} \pi_p(a|s) & \text{if } \eta < \epsilon \\ \pi_v(a|s) & \text{otherwise.} \end{cases} \tag{7}$$

其中，$\eta \sim U$ 是从均匀分布 U(0, 1) 中抽取的，$\epsilon$ 是一个阈值，用于决定选择策略动作还是价值动作的可能性。与公式（5）结合使用时，VIS 会通过随机交换 $\boldsymbol{\pi_p}$ 和 $\boldsymbol{\pi_v}$，迫使 AlphaZero 的策略与价值预测保持一致。直观而言，VIS 会促使 AlphaZero 使其策略与价值函数保持一致，因为对于任何一次动作选择，它都无法预知自己将需要使用哪一个。

### 4.2 价值引导对称增强

穷举搜索组合状态空间并不可行，因此在训练期间，AlphaZero 依靠蒙特卡洛树搜索来缩减其遇到的状态集合。然而，这也使 AlphaZero 的网络有责任泛化到训练期间被蒙特卡洛树搜索跳过的状态。我们发现，AlphaZero 对此类状态的价值预测泛化能力较差。

为了提升 AlphaZero 价值网络的泛化能力，我们提出了价值引导对称增强（VISA）：一种明确针对 AlphaZero 价值函数中不确定性的数据增强方法。对于训练期间遇到的每个状态 $s$，VISA 首先生成一组 $\boldsymbol{s_\sigma} = \{s_{\sigma_1}, ..., s_{\sigma_N}\}$ 个额外状态，其中每个 $s_{\sigma_i} \in \boldsymbol{s_\sigma}$ 都是对 $s$ 应用对称变换 $\sigma_i$(s) 的结果。然后，VISA 找出价值不确定性最大的变换后状态 $s^*_\sigma$，如下所示：

$$s^*_\sigma = \max_i [(v(s) - v(s_{\sigma_i}))^2]$$

并将 $s$ 和 $s^*_\sigma$ 都加入 AlphaZero 的经验回放缓冲区。请注意，只添加 $s^*_\sigma$的 VISA 优于随机数据增强——它针对的是价值网络最不确定的增强方式；也优于穷举数据增强——与将所有 $\boldsymbol{s_\sigma}$ 存储在经验回放缓冲区中相比，它更加节省内存。

在本研究中，我们考察三种主要的对称变换：二面体群中的旋转和反射，以及反演。直观来说，在棋盘游戏场景中（如图1所示），这些变换分别对应于旋转棋盘、镜像翻转棋盘（垂直或水平），或翻转每位玩家的棋子。值得注意的是，棋盘反演还要求在计算公式 (5) 时反演真实游戏树价值 $z$。

### 4.3 衡量策略-价值错位

我们还引入了一种策略-价值失配的信息论度量。给定 AlphaZero 的标准策略 $\pi_p$ 和第4.1节中的价值信息策略 $\pi_v$，可以将策略-价值失配计算为 KL 散度：

$$D_{\text{KL}}(\pi_p || \pi_v) \tag{8}$$

直观而言，策略-价值对齐程度较高的模型会产生较低的散度项，而策略网络与价值网络不一致则会导致较高的散度。

## 5 实验

本节首先展示 AlphaZero 中存在策略-价值错位和价值函数不一致性。随后，我们评估 VISA-VIS，并考察它在多大程度上缓解了这些现象。

**环境：**我们考虑三种已求解、对称的双人零和博弈，其复杂程度依次递增：井字棋、4×4井字棋和四子棋。我们强调，研究已求解博弈对我们的分析至关重要。为了恰当地衡量 AlphaZero 价值泛化误差的范围，我们需要一个合适的真实值，以便与 AlphaZero 的预测进行比较。已求解的博弈满足了这一需求。在每种博弈中，结果分别可取 $z = 1$、0、和 $-1$，对应胜局、败局和平局。

**训练细节：**按照第3.1节中的算法，AlphaZero 经过固定数量的时间步训练。AlphaZero 的网络由一个 ResNet [He 等人的 2016] 骨干网络组成，并设有分别用于策略和价值预测的独立输出头。VISA-VIS 的训练过程细节均相同，但采用第4节中概述的修改。包括超参数在内的其他细节见附录A。

### 5.1 可枚举博弈树中的 AlphaZero

我们首先系统地研究 AlphaZero 在可枚举博弈树——井字棋中的表现。由于井字棋的状态空间很小，我们可以针对每个合法状态评估 AlphaZero 的预测，从而全面了解 AlphaZero 的行为。

训练结束后，我们通过两种方式验证每个 AlphaZero 模型的表现：首先，在五个随机种子下，让每个模型与预言机对手进行 1，000 场对局，并根据游戏规则 ($z \in \{1, 0, -1\}$) 为每场对局计分。为求完整，我们同时纳入 AlphaZero 先行和后行的对局。接着，我们遍历合法状态空间，计算 AlphaZero 的价值预测与真实博弈树价值之间的均方误差，以此验证其价值预测的准确性。这项分析的结果见图2。

从图2a中的训练曲线可以清楚地看出，AlphaZero 很快就收敛到了强大的博弈策略——在最优对弈下，预期结果是平局——这并不令人意外。另一方面，价值预测分析的结果却相当出人意料。图2b显示，尽管 AlphaZero 能够进行最优对弈，其价值预测却并非总是准确。事实上，当误差

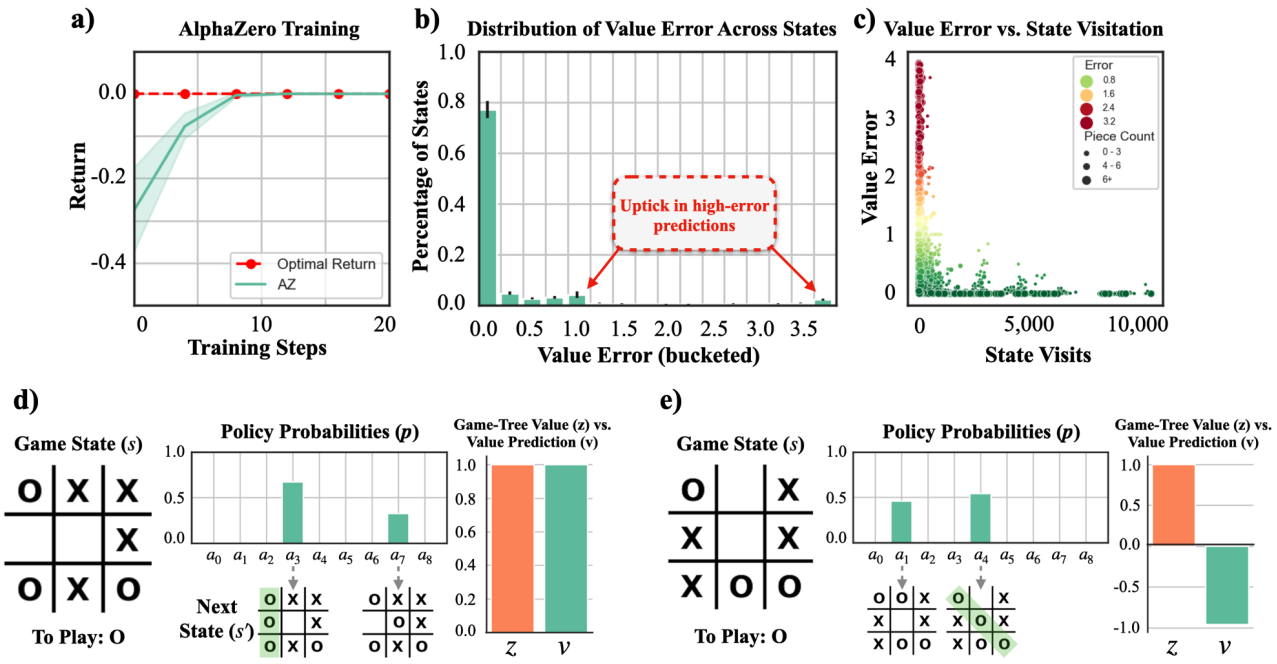


图2：在可枚举博弈树中评估 AlphaZero。**a)** AlphaZero 在训练期间的表现。对于简单的游戏，AlphaZero 会迅速收敛到最优策略。**b)** 尽管对弈表现最优，对 AlphaZero 的价值函数性能进行穷尽分析后发现，它会做出许多错误且误差很大的预测（相对于每个状态的真实博弈树价值而言）。**c)** 将价值误差表示为状态访问次数的函数后可以看出，AlphaZero 的价值函数无法泛化到访问频率较低的状态和未见状态。**d)** 一个定性示例，其中 AlphaZero 的策略与价值预测均正确。**e)** 一个展示策略-价值失配的定性示例——AlphaZero 的策略预测选择了获胜的一步，但其价值预测却不正确（预测会输）。

对 AlphaZero 价值预测的误差进行量化后，可以看到在 $e > 1.0$ 处出现了明显的误差峰值——这表明 AlphaZero 预测结果为负，但真实的博弈树价值为胜（反之亦然）；在 $e > 3.5$处还出现了另一个误差峰值——这表明真实的博弈树价值为胜（反之亦然），而 AlphaZero 却非常确信地预测结果为负。

将误差表示为训练期间 AlphaZero 对每个状态的访问次数的函数（见图2c）后，可以清楚地看出，AlphaZero 的价值预测在访问频率较低或未被访问过的状态上误差最高，这表明 AlphaZero 未能实现泛化。此外，如图2e所示的定性示例（其中包含 AlphaZero 的策略预测及其相应的价值估计）表明，在许多情况下，尽管价值预测极不准确（明明有获胜机会，却预测会输），AlphaZero 的策略预测仍然正确（选择了获胜的一步）。

总而言之，这项分析表明：(i) AlphaZero 的价值预测不一致，且泛化能力较差；(ii) AlphaZero 的价值预测与策略预测不一致（即 AlphaZero 存在策略-价值失配）。

## 5.2 消融实验：VISA、VIS 和 VISA-VIS

接下来，我们将研究所提出的方法 VISA-VIS 在多大程度上能够纠正 AlphaZero 易受泛化误差和策略-价值失配影响的问题。我们对经过训练的 VISA-VIS 模型进行相同的测试时分析，并且为了评估我们方法中各个组件的贡献，分别对 VISA 和 VIS 进行消融实验。除 AlphaZero 和 VISA-VIS 外，我们还报告了一个从随机初始状态开始训练的 AlphaZero 模型的结果（每次训练 rollout 都从一个随机的合法非终止状态开始）。尽管使用随机初始状态进行训练无法扩展到更复杂的游戏，但在这一设置下，它为价值误差提供了一个下界（即，当允许 AlphaZero 多次访问所有状态时，其表现能达到多好）。

**策略-价值失配** 首先，我们根据公式（8）的定义，计算各方法在所有环境状态下表现出的平均策略-价值失配。如图3所示，VIS（因此 VISA-VIS 也）显著提升了模型策略与价值预测的一致性：与 AlphaZero 相比，策略-价值失配降低了 50%；与 AlphaZero Random 相比，则降低了 60% 以上。这些结果也很好地说明了价值预测准确度与策略-价值失配在本质上的区别。以 AlphaZero Random 为例，尽管它能直接观察所有游戏状态，却仍表现出较高的策略-价值失配。这是因为仅凭价值估计并不能说明全部情况——AlphaZero 的价值函数仍可能与其策略不一致。

**价值误差** 本次消融实验的价值误差结果如图4所示。由此可见，VISA、VIS 和 VISA-VIS 均显著提升了 AlphaZero 的价值预测质量。具体而言，与 AlphaZero 相比，VISA-VIS 将具有中等价值误差的状态占比降低了近 10%，并且仅对 0.6% 的状态产生高价值误差预测（比 AlphaZero 少），这一结果与能够观测所有状态的 AlphaZero 版本（AZ Random）所取得的完美分数极为接近。

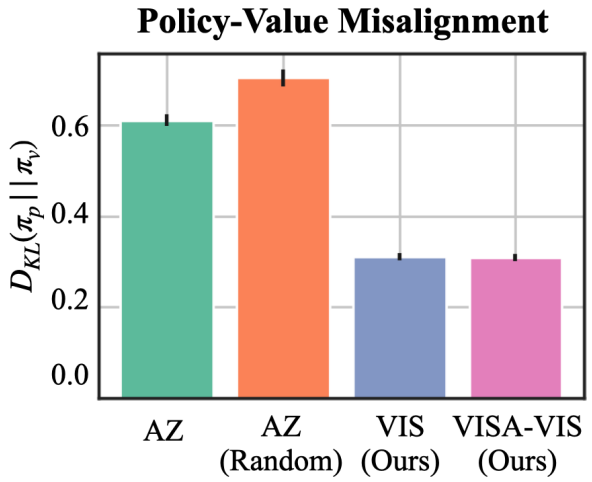


图3：根据公式 8 中介绍的信息论度量，各方法的策略-价值失配程度（越低越好）。与 AlphaZero (AZ) 相比，VIS 将策略-价值失配降低了 50% 以上；与 VISA 结合后（VISA-VIS），仍能保持一致性。

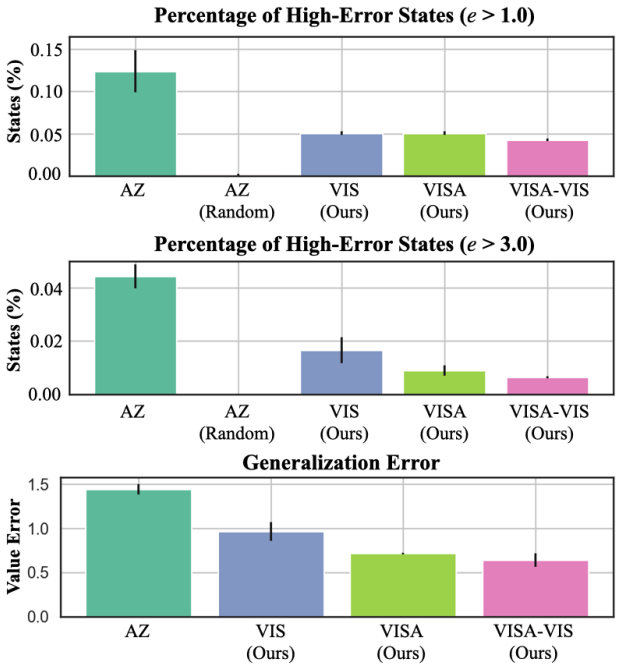


图4：各方法的价值误差（越低越好）。VISA-VIS 显著降低了中等（$e > 1.0$）和高（$e > 3.0$）误差价值预测的比例，并将 AlphaZero 价值网络的平均泛化误差降低了 50% 以上。

在泛化方面，我们发现 VISA-VIS 将平均价值泛化误差降低了逾 50%（从平均值 1.44 降至 0.64）。值得注意的是，对于高误差预测和泛化误差，我们发现 VISA 与 VIS 的性能提升具有互补性——VISA 和 VIS 都会减少高误差状态的数量，但二者结合时价值误差的降幅最大（泛化误差也同样如此）。

表1：对抗性检测状态。

| Game | 对抗状态 | 平均值误差 |
|---|---|---|
| 4×4井字棋 | 10,241 | 2.63 |
| 四子棋 | 52,768 | 2.36 |

### 5.3 扩展规模

我们通过在复杂度显著更高的游戏中进行评估，展示了该方法的可扩展性。这项分析的结果见图 5。

**对抗性状态检测** 在更复杂的环境中，不可能枚举整个状态空间并逐一评估每个状态。因此，我们对评估策略作出两项调整，以支持更复杂的游戏。首先，我们主要关注残局状态。残局状态使我们能够针对 AlphaZero 最重要的决策——单个着法就可能决定游戏结果——同时仍能从博弈树中提取真实结果，用于验证 AlphaZero 的预测。

其次，受先前关于 AlphaZero 对抗样本的研究启发，Wang 等人， 2022], 我们定义了一种对抗策略，用于找出 AlphaZero 会做出不一致预测的状态。具体而言，在测试时，使用公式（4）计算搜索引导的动作概率后，我们强制 AlphaZero 采取被赋予最低概率的动作 $a_t = \min_a[\pi(a|s)]$。由于 $\pi$ 编码了从状态 $s$采取动作 $a$ 的预期值，因此我们实际上是在迫使 AlphaZero 走出可选的最差一步。在每次推演中，这种对抗策略都会引导 AlphaZero 进入状态空间中那些最可能在训练期间被搜索忽略的子集。事实证明，这种简单的对抗策略能够有效生成 AlphaZero 评估不佳的状态。在我们的每种环境中，我们都使用上述对抗检测策略，让一个经过训练的 AlphaZero 模型与自身对弈 100，000 场，并收集 AlphaZero 做出高误差（$e > 1.0$）值预测的所有独特残局状态。在 4×4井字棋中，对抗检测器在三**个分别训练的 AlphaZero 模型中每一个**都找出了超过 10，000 个获得高误差价值估计的状态。该检测器在四子棋中也找出了超过 50，000 个状态。这些统计数据汇总于表1。

在图5a所示的4×4井字棋对抗分析结果中，我们看到，常规 AlphaZero 的价值函数一致性较差——这些状态的平均价值误差为 2.63——且存在显著的策略-价值失配——平均 KL 散度为 $D_{KL}(\pi_p||\pi_v) = 1.41$。我们提出的 VISA-VIS 方法解决了这两个问题，将平均价值误差降低

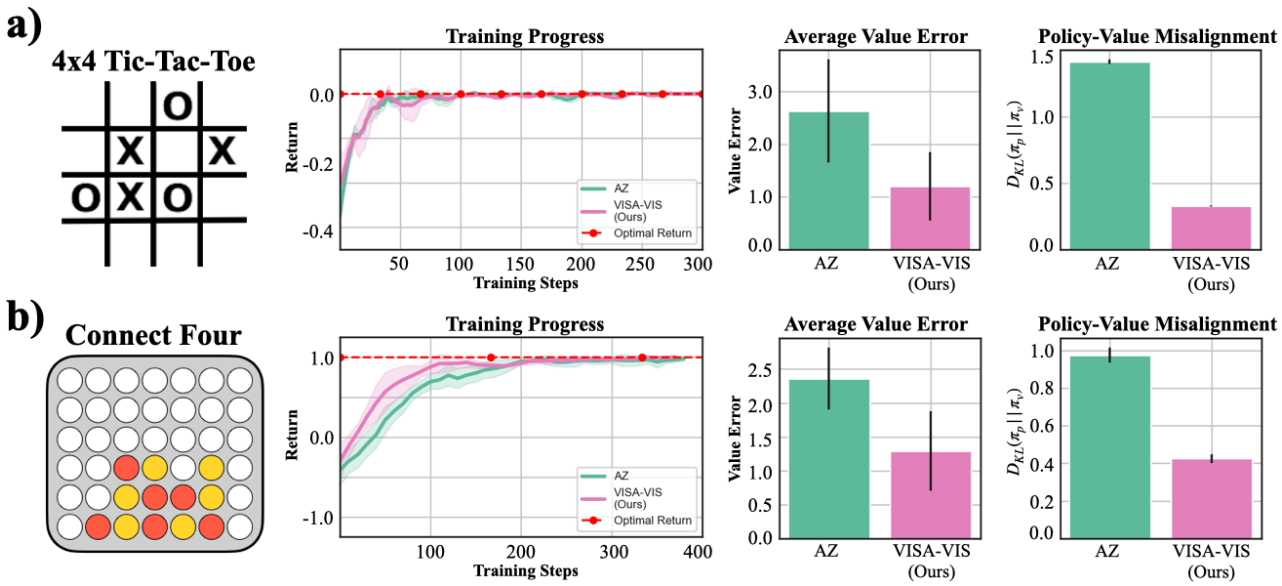


图 5：AlphaZero 与 VISA-VIS 在更复杂环境中的表现。**a)** 在 4×4井字棋中，对抗状态分析发现了 10,000 多个 AlphaZero 表现不佳的状态（价值误差高、策略-价值失配严重）。VISA-VIS 大幅纠正了 AlphaZero 在这些状态中的错误，使平均价值误差降低了 55%，策略-价值失配降低了 76%。此外，在训练过程中，AlphaZero 和 VISA-VIS 的收敛速度相近，这表明 VISA-VIS 达到了这一效果，却没有相应增加样本复杂度。**b)** 在四子棋中，对抗分析得出了定性相似的结果。AlphaZero 再次表现出价值函数不一致性——对抗状态中的平均价值误差为 2.36；同时也存在策略-价值失配——平均 KL散度为 0.975。VISA-VIS 同样降低了这些误差：在相同的对抗生成状态中，其平均价值误差为 1.29，策略-价值失配得分为 0.425（分别降低了 45% 和 56.5%）。此外，就样本复杂度而言，VISA-VIS 的平均收敛速度快于 AlphaZero（在四子棋游戏中）。这揭示了 VISA-VIS 的另一项优势——对称增强缩短了学习最优策略所需的时间。

55% 至 1.20，并将策略-价值失配降低 76%，降至 $D_{KL}(\pi_p||\pi_v) = 0.327$。

图5b展示了我们在四子棋中的对抗分析。在这里，我们发现了与4×4井字棋案例定性相似的结果。我们的对抗检测器能够识别出超过50,000个会导致AlphaZero产生高误差预测的状态。与之前一样，VISA-VIS纠正了这些高误差预测。具体而言，这些对抗状态使AlphaZero的平均价值误差达到2.36，平均策略-价值失配得分达到0.975。VISA-VIS的表现更佳，平均误差为1.29，平均策略-价值失配得分为0.425——与AlphaZero相比，分别降低了45%和56.5%。

总而言之，这些结果不仅表明策略-价值失配和价值函数不一致性是存在于多种环境中的普遍现象，也证明了我们的方法VISA-VIS能够可靠地同时应对这两种现象。

**样本复杂度** 我们还绘制了 AlphaZero 和 VISA-VIS 在训练过程中的博弈表现。与 3×3 情况一样，在最优对弈下，4×4井字棋的预期结果是平局（$z = 1$）。如图5a所示，我们发现证据表明，AlphaZero 和 VISA-VIS 的收敛时间大致相同。总体而言，VISA-VIS 在实现上述策略-值对齐和值函数一致性改进的同时，并未牺牲样本复杂度。

在四子棋中，我们发现 VISA-VISA 对样本复杂度有更显著的积极影响。从图5b)可以清楚看出，平均而言，VISA-VIS 比 AlphaZero 更快收敛到最优策略（四子棋中先手玩家的预期结果是获胜）。这一结果表明，在这款更复杂的游戏中，VISA-VIS 的对称数据增强降低了学习最优策略所需的样本复杂度。这进一步凸显了 VISA-VISA 的另一项优势。

## 6 结论

在这项工作中，我们对 AlphaZero 进行了系统研究，发现了两种会损害其鲁棒性的现象——策略-价值失配和价值函数不一致性。至关重要的是，搜索会掩盖这两种现象，这表明尽管它们存在，AlphaZero 仍有可能采取最优行动，而且它们可能会导致 AlphaZero 容易受到对抗性样本的影响。我们提出了 VISA-VIS，它结合了值引导选择（VIS）——一种面向 AlphaZero 的新型动作选择准则——以及值引导对称增强（VISA）——一种针对价值函数相对于对称性的不确定性的新型数据增强方法。实验表明，VISA-VIS 显著提升了 AlphaZero 的策略-价值对齐程度和价值函数鲁棒性。

## 伦理声明

本研究考察了 AlphaZero 的鲁棒性和可靠性。随着人工智能等大规模人工智能系统在现实应用中日益普及，了解其局限性和潜在失效模式十分重要。利用自我对弈进行强化学习的算法尤其容易带来潜在危害（例如，在安全性、可靠性和可解释性方面），因为它们在训练期间不会接触任何由人类提供的数据。我们认为，为了确保此类系统总体上能够提供帮助，并在部署期间减轻无法预见或非预期的危害，必须确保它们能够抵御潜在攻击（例如对抗性输入），并具备整体可靠性。本研究旨在朝这一方向迈出一小步。本研究未使用敏感数据，也未将敏感任务或研究纳入其成果。

## 参考文献


[Forde 等， 2022] Jessica Zosa Forde、Charles Lovering、George Konidaris、Ellie Pavlick 和 Michael L Littman。AlphaZero 学会了哪些概念、在何时何地学会？来自 Hex 游戏的启示。载于 *AAAI* 游戏中的强化学习研讨会，2022年。[Gleave等， 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。*arXiv* 预印本 *arXiv:1905.10615*，2019年。[Goodfellow等， 2014] Ian Goodfellow、Jonathon Shlens 和 Christian Szegedy。解释并利用对抗样本。*arXiv* 预印本 *arXiv:1412.6572*，2014年。[Greydanus等， 2018] Samuel Greydanus、Anurag Koul、Jonathan Dodge 和 Alan Fern。可视化并理解 Atari 智能体。载于国际机器学习会议。PMLR，2018年。[Grupen等， 2022] Niko Grupen、Natasha Jaques、Been Kim 和 Shayegan Omidshafiei。基于概念理解涌现式多智能体行为。载于 *NeurIPS 2022* 深度强化学习研讨会，2022年。[He等， 2016] 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 *IEEE*计算机视觉与模式识别会议论文集，2016年。[Hendrycks等， 2021] Dan Hendrycks、Kevin Zhao、Steven Basart、Jacob Steinhardt 和 Dawn Song。自然对抗样本。载于 计算机视觉与模式识别会议论文集，2021年。[Huang等， 2017] Sandy Huang、Nicolas Papernot、Ian Goodfellow、Yan Duan 和 Pieter Abbeel。针对神经网络策略的对抗攻击。*arXiv* 预印本 *arXiv:1702.02284*，2017年。[Kaplan 等， 2020] Jared Kaplan、Sam McCandlish、Tom Henighan、Tom B Brown、Benjamin Chess、Rewon Child、Scott Gray、Alec Radford、Jeffrey Wu 和 Dario Amodei。神经语言模型的缩放定律。*arXiv* 预印本 *arXiv:2001.08361*，2020年。

[Kos 和 Song， 2017] Jernej Kos 和 Dawn Song。《深入探究针对深度策略的对抗攻击》。arXiv 预印本 arXiv:1705.06452，2017年。[Lan 等人， 2022] Li-Cheng Lan、Huan Zhang、Ti-Rong Wu、Meng-Yu Tsai、I Wu、Cho-Jui Hsieh 等人。《类 AlphaZero 智能体能否抵御对抗扰动？》arXiv 预印本 arXiv:2211.03769，2022年。[Lanctot 等人， 2019] Marc Lanctot、Edward Lockhart、Jean-Baptiste Lespiau、Vinicius Zambaldi、Satyaki Upadhyay、Julien P´erolat、Sriram Srinivasan、Finbarr Timbers、Karl Tuyls、Shayegan Omidshafiei、Daniel Hennes、Dustin Morrill、Paul Muller、Timo Ewalds、Ryan Faulkner、J´anos Kram´ar、Bart De Vylder、Brennan Saeta、James Bradbury、David Ding 等人。《OpenSpiel：用于游戏强化学习的框架》。2019年。[Lin 等人， 2017] Yen-Chen Lin、Zhang-Wei Hong、Yuan-Hong Liao、Meng-Li Shih、Ming-Yu Liu 和 Min Sun。《深度强化学习智能体对抗攻击的策略》。arXiv 预印本 arXiv:1703.06748，2017年。[McGrath 等人， 2022] Thomas McGrath、Andrei Kapishnikov、Nenad Toma˘sev、Adam Pearce、Martin Wattenberg、Demis Hassabis、Been Kim、Ulrich Paquet 和 Vladimir Kramnik。《AlphaZero 中国际象棋知识的获取》。《美国国家科学院院刊》，2022年。[Metz， 2016] Cade Metz。《两步之内，AlphaGo 和李世石重新定义了未来》，2016年。[Neumann 和 Gros， 2022] Oren Neumann 和 Claudius Gros。《多智能体强化学习模型的缩放定律》。arXiv 预印本 arXiv:2210.00849，2022年。[Omidshafiei 等人， 2022] Shayegan Omidshafiei、Andrei Kapishnikov、Yannick Assogba、Lucas Dixon 和 Been Kim。《超越奖励：离线多智能体行为分析的层级视角》。arXiv 预印本 arXiv:2206.09046，2022年。[Selvaraju 等人， 2017] Ramprasaath R Selvaraju、Michael Cogswell、Abhishek Das、Ramakrishna Vedantam、Devi Parikh 和 Dhruv Batra。《Grad-CAM：通过基于梯度的定位，对深度网络进行可视化解释》。发表于国际计算机视觉大会论文集，2017年。[Silver 等，2017a] David Silver、Thomas Hubert、Julian Schrittwieser、Ioannis Antonoglou、Matthew Lai、Arthur Guez、Marc Lanctot、Laurent Sifre、Dharshan Kumaran、Thore Graepel 等人。《通过自我博弈，运用通用强化学习算法精通国际象棋和日本将棋》。arXiv 预印本 arXiv:1712.01815，2017年。[Silver 等，2017b] David Silver、Julian Schrittwieser、Karen Simonyan、Ioannis Antonoglou、Aja Huang、Arthur Guez、Thomas Hubert、Lucas Baker、Matthew Lai、Adrian Bolton 等人。《在没有人类知识的情况下精通围棋》。《自然》，2017年。[Wang 等人， 2022] Tony Tong Wang、Adam Gleave、Nora Belrose、Tom Tseng、Joseph Miller、Michael D Dennis、Yawen Duan、Viktor Pogrebniak、Sergey Levine 和 Stuart Russell。《对抗策略击败职业级围棋人工智能》。arXiv 预印本 arXiv:2211.00241，2022年。

表 2：AlphaZero 和 VISA-VIS 的超参数。

| 训练参数和超参数 | | | |
|---|---|---|---|
| 参数 Name | 值（井字棋） | 值（4×4井字棋） | 值（四子棋） |
| 对局数 | 5e5 | 1。75e6 | 7.5e6 |
| 批次大小 | 64 | 128 | 256 |
| 学习率 | 1e−3 | 1e−4 | 1e−4 |
| 网络深度 | 2 | 4 | 4 |
| 网络宽度 | 128 | 128 | 128 |
| $\lambda$ | 1e−4 | 1e−4 | 1e−4 |

表 3：AlphaZero 和 VISA-VIS 的搜索参数。

| 搜索参数 | | | |
|---|---|---|---|
| 参数 Name | 值（井字棋） | 值（4x4 井字棋） | 值（四子棋） |
| 蒙特卡洛树搜索模拟次数 | 25 | 25 | 50 |
| $c$ | 2.0 | 2.0 | 2.0 |
| 初始 $\tau$ | 1.0 | 1.0 | 1.0 |
| $\tau$ 步（步数） | 5 | 9 | 21 |

## A 训练详情（续）

这里我们提供更多训练细节，包括搜索参数、训练超参数，以及训练 AlphaZero 和 VISA-VIS 时使用的其他学习细节。

**训练参数** 两种算法分别在井字棋、4×4井字棋和四子棋中进行了 500,000 局、1.75M 局、1.75M 局和 7.5M 局的训练。联合策略价值网络由一个 ResNet 骨干网络构成，并设有独立的输出头，用于输出策略和价值预测，如公式 (1) 所示。每个全连接层均以 128 个神经元的宽度进行初始化，网络骨干的深度则根据游戏复杂度进行缩放——井字棋使用 $d = 2$ 个 ResNet 模块，$d = 4$ 用于 4×4井字棋和四子棋。每种游戏的批次大小和学习率也有所不同：(i) 井字棋使用 1e$-3$ 的学习率和 64 的批次大小；(ii) 4×4井字棋使用 1e$-4$ 的学习率和 128 的批次大小；(iii) 四子棋使用 1e$-4$ 的学习率和 256 的批次大小。如公式 (5) 所述，所有游戏均使用系数为 $\lambda = 1$e$-4$ 的值来调整 L2 正则化的权重。

**搜索参数** 回想第3.1节，多个参数会影响 AlphaZero（因此也包括 VISA-VIS）的搜索过程。在训练期间，AlphaZero 在井字棋和 4×4井字棋中每个时间步进行 25 次蒙特卡洛树搜索模拟，在四子棋中每个时间步进行 50 次模拟。对于每次模拟，所有游戏中都使用树内探索参数 $c = 2.0$，通过式（2）所指定的上置信界来引导探索。温度参数 $\tau$用于引导式（4）所指定的动作选择步骤中的探索，所有游戏的初始值均设为 $\tau = 1.0$。在每次训练 rollout 中，经过固定步数后会将 $\tau$ 降低；井字棋、4×4井字棋和四子棋对应的步数分别为 5、9 和 21，这与 Silver 等人 [2017 b]中的做法一致。

**实现** 我们的实现基于 DeepMind 的 OpenSpiel 项目提供的开源 AlphaZero [Lanctot 等人， 2019]。VISA-VIS 扩展了 OpenSpiel 中提供的实现。最重要的实现细节是为每种游戏选择棋盘表示方式。对于每种游戏，棋盘状态都表示为由 3 $HxW$ 个帧组成的堆栈——其中 $H$ 和 $W$ 分别表示棋盘的高度和宽度——用于表示每位玩家的棋子（井字棋的两种变体中双方的圆圈和叉号，以及四子棋中的红色和黄色棋子），并增加一个平面来编码回合信息（即轮到哪位玩家行动）。每种游戏还将游戏规则编码到动作空间中，用于指定每个回合的合法着法——非法着法会被屏蔽，并由价值网络赋予零概率。

**资源** Allexperiments 均在配备 2x18 核 Intel Xeon Skylake 6154 CPU 和一块 16GB 内存的 Nvidia V100 GPU 的集群上运行。