arXiv:2603.17075v1 [cs.LG] 17 Mar 2026

CircuitBuilder：通过强化学习从多项式构造电路

Weikun K. Zhang[*†] **Rohan Pandey**[*†] **Bhaumik Mehta**[†] **Kaijie Jin**[†] **Naomi Morato**[†]

Archit Ganapule[†] **Michael Ruofan Zeng**[†‡] **Jarod Alper**[†]

摘要

受自动证明生成和 Valiant 的 VP 与 VNP 猜想启发，我们研究如何使用加法门和乘法门发现能够计算多项式的高效算术电路。我们将这一问题表述为单人游戏：强化学习智能体尝试在固定数量的运算内构建电路。我们实现了 AlphaZero 式训练循环，并比较两种方法：采用蒙特卡洛树搜索的近端策略优化（PPO+MCTS）和软演员-评论家（SAC）。在两个变量的目标上，SAC 的成功率最高；而 PPO+MCTS 可扩展至三个变量，并在更难的实例上持续改进。这些结果表明，多项式电路综合是一个简洁且可验证的研究环境，适合研究能够自我改进的搜索策略。

1 引言

寻找计算多项式的最高效方法，是代数复杂性理论中的一个基础问题（例如，参见 B¨urgisser 等人（1997年）；Shpilka 和 Yehudayoff（2009年））。这个任务被称为算术电路问题，其内容是寻找由加法门和乘法门组成的序列，以尽可能少的运算计算目标多项式 $f(x_1, \ldots, x_n)$。这个问题不只是理论上的练习，它是 P 与 NP 问题在代数领域的对应问题，其中的复杂度类

- VP——可由多项式规模电路计算的多项式族，以及
- VNP——系数可在多项式时间内计算的多项式族，是 NP 在代数领域的对应物，

代表了高效计算的极限（Valiant，1979年）。理解某些多项式族（如永久式）的电路规模，可能会推动 VP 与 VNP 问题取得巨大进展（Valiant，1979年）。

此外，寻找最小电路是搜索简短数学证明的一种较简单形式；数学证明通过一系列逻辑推导，从公理得出定理。算术电路问题是理想的试验场，可用于检验哪些强化学习方法可能适用于自动证明生成。通过系统研究各种算法在搜索算术电路方面的有效性，我们希望深入了解证明搜索问题（Hubert 等，2025）。

然而，这类电路的搜索空间极其庞大。对于包含 k 个中间节点的电路，可选的下一步运算数量以 $O(k^2)$的速度增长，导致指数级增长，使得对复杂多项式进行穷举搜索并不切实际。历史上，诸如用于单变量多项式的霍纳方案，或用于初等对称多项式的递归结构等高效构造，都是通过人类直觉发现的（见第 2.1 节）。在这项工作中，我们探究机器学习智能体能否自主发现这些以及其他高度高效的计算结构。

*共同贡献。
†华盛顿大学
‡联系邮箱：Michael R. Zeng ¡zengrf@uw.edu¿

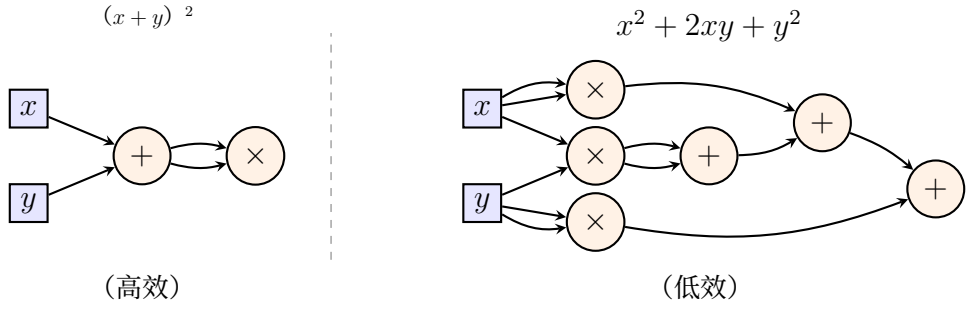


图 1：多项式 $x^2+2xy+y^2$ 的两种算术电路。

受 AlphaZero（Silver 等，2018年）掌握复杂双人游戏的成功，以及 AlphaProof（Hubert 等，2025）自动证明生成的启发，我们将算术电路构造建模为单人马尔可夫决策过程（MDP）。在这一环境中，智能体从输入变量和常数出发，必须选择一系列代数运算，最终得到目标多项式。我们衡量成功的主要指标是智能体准确计算给定多项式的能力：既要实现最优目标——最小门数，也要能够成功泛化到未见过的多项式。

我们的贡献是对这一问题的两种不同架构方法进行比较研究：

- CircuitBuilder（近端策略优化 +蒙特卡洛树搜索）：一种强化学习智能体，结合使用近端策略优化（PPO）和蒙特卡洛树搜索（MCTS），以引导智能体在庞大的搜索空间中进行探索，并应对稀疏的奖励信号。
- 软演员-评论家（SAC）：一种离策略演员-评论家方法，用于应对稀疏奖励信号带来的挑战；稀疏奖励信号常常阻碍电路发现任务中的训练进程（Haarnoja 等，2018）。

我们在此前未见过的多项式上评估这些方法，并证明深度搜索和结构化学习能够找回最优或近乎最优的电路。我们还研究了高效采样具有多个最优电路的多项式的策略，以改进训练和测试。我们的结果表明，基于学习的方法能够为代数复杂性带来新的见解，并为实现递归式自我改进的数学发现智能体提供一条可扩展的路径。

2 背景

2.1 算术电路

我们依据 B¨urgisser 等人（1997年）总结一些关于算术电路的事实。设 $\mathbb{F}$ 为一个域，或更一般地说，是一个交换环。设 $X=\{x_1,\ldots,x_n\}$ 为形式变量。一个系数取自 $\mathbb{F}$、变量为 X 的**算术电路** 是一个有限有向无环图（DAG），其顶点（称为“门”）的入度为 0 或 2。入度为 0 的门是输入门，并以变量 X标记。入度为 2 的门是加法门 / 乘法门，并分别以 $+$ 或 $\times$标记。其中一个指定的门被指定为输出门。请注意，B¨urgisser 等人（1997年）允许输入门取值为可能为负的常数。本文仅讨论输入门只含变量的情形（其中一个变量可视为常数“1”），其他常数值留待未来工作。

每个门 v 都归纳地计算一个多项式 $f_v\in\mathbb{F}[X]$。输入门表示单项式 x_1、.....、x_n。如果 v是一个 $+$/ $\times$门，且其输入邻居为 u、w，那么 $f_v=f_u+f_w$（分别为 $f_v=f_u\cdot f_w$）。若输出门处的多项式等于 f，则该电路计算多项式 f。电路的复杂度是门的数量，而深度是从输入到输出的最长有向路径的长度。语法次数归纳定义如下：对于常数输入，$\deg(v)=0$；对于变量输入，$\deg(v)=1$；在 $+$-门处，$\deg(v)=\max\{\deg(u),\deg(w)\}$；在 $\times$-门处，$\deg(v)=\deg(u)+\deg(w)$。它给出了 f_v总次数的上界。两个计算多项式 $x^2+2xy+y^2$的不同算术电路见图 1。

许多多项式都可以由规模较小的算术电路计算，因为中间子表达式可以重复利用。一个最初的例子是霍纳方案（Horner，1819）。给定一个单变量多项式 $p(x) = \sum_{i=0}^{d} a_i x^i$，可以通过递减递推式 $t_d = a_d,\qquad t_i = a_i + x \cdot t_{i+1} (i = d-1、\ldots\ldots、0)$，来计算 p，从而 $p(x) = t_0$。这一递归等价于以下嵌套表达式

$$p(x) = a_0 + x\big(a_1 + x(a_2 + \cdots + x(a_{d-1} + xa_d)\cdots)\big).$$

霍纳法恰好使用 d 个加法门和 d 个乘法门，与朴素电路形成对比：朴素电路会为每一项重新计算幂 x^i。

一个多变量示例是初等对称多项式。在 n 个变量中，k 次初等对称多项式为 $e_k(x_1, \ldots, x_n) = \sum_{1 \le i_1 < \cdots < i_k \le n} x_{i_1} \cdots x_{i_k}$，并满足递推关系 $e_k(x_1, \ldots, x_n) = e_k(x_1, \ldots, x_{n-1}) + x_n \cdot e_{k-1}(x_1, \ldots, x_{n-1})$。该递推关系将 e_k 表示为两个先前已计算出的、定义在 $(n-1)$ 个变量上的多项式。

算术电路问题是代数复杂性理论的核心。VP 类由一族多项式构成，这些多项式可由规模关于参数呈多项式增长的算术电路计算（Valiant，1979年）。Valiant 的 VNP 类是 NP 的代数类比。随后，Valiant 指出，计算永久式多项式

$$\operatorname{per}_n (x_{i,j}) = \sum_{\sigma \in S_n} \prod_{i=1}^{n} x_{i,\sigma(i)}, \quad (i,j) \in [n] \times [n]$$

是 VNP 完全的（Valiant，1979年）。

2.2 AlphaZero 风格的搜索与策略优化

AlphaZero 的成就，在于它能够精通双人完全信息博弈（如围棋、国际象棋和将棋）（Silver 等，2018）。AlphaZero 将神经网络与蒙特卡洛树搜索框架结合，使智能体仅通过自我对弈就能掌握游戏玩法，无须接受任何超出游戏规则之外的人类反馈或输入。该方法的核心是一个同时输出策略（所选走法的概率分布）和价值估计（对给定状态下结果的预测）的单一神经网络，二者共同引导树搜索。

蒙特卡洛树搜索 蒙特卡洛树搜索算法是一种强大的搜索算法，它通过采样基于价值的轨迹，逐步构建搜索树。它利用上置信界（UCB）在探索与利用之间取得平衡（Kocsis & Szepesv´ari，2006）。对于一个状态—动作对（s，a），有

$$\mathrm{UCB}(s,a) = \frac{Q(s,a)}{N(s,a)} + c\sqrt{\frac{\ln N(s)}{N(s,a)}},$$

其中，$Q(s,\ a)$ 是总价值，$N(s,\ a)$ 是该动作的访问次数，$N(s)$ 是父节点的访问次数，而 c是一个探索常数，用于在利用高价值动作与探索访问次数较少的节点之间取得平衡。

该算法分为四个阶段：选择阶段，使用 UCB 遍历树；扩展阶段，添加一个新节点；模拟阶段，估算新节点的价值；以及反向传播阶段，沿访问路径更新统计数据。

电路构建具有顺序性（每次选择一个门），这与蒙特卡洛树搜索的树结构天然契合，因此非常适合在算术电路的组合搜索空间中进行搜索。

近端策略优化 近端策略优化（近端策略优化）是一种基于策略梯度的强化学习方法；其核心思想是通过防止策略在单次更新中发生剧烈变化来稳定训练。近端策略优化的核心公式是裁剪代理函数（Schulman等人，2017年）

$$L^{\mathrm{CLIP}}(\theta) = \mathbb{E}\left[\min\left(r_t(\theta)\hat{A}_t, \mathrm{clip}(r_t(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t\right)\right],$$

其中 $r_t(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{\text{old}}}(a_t|s_t)}$ 是新旧策略之间的概率比率，$\hat{} A_t$ 是估计的优势，而 ϵ 是裁剪超参数，用于防止更新幅度过大而产生不利影响。

在我们的工作中，我们将PPO实现为神经网络的训练算法。这取代了AlphaZero原有的策略迭代方案实现。PPO的训练稳定性在我们的场景中尤其重要，因为电路构建的奖励景观较为稀疏，并且对较大的策略变化十分敏感。

因此，通过PPO训练的神经网络提供策略先验和值估计，用于引导蒙特卡洛树搜索；而蒙特卡洛树搜索生成的改进动作分布则作为网络选择最佳动作的训练目标。

软演员-评论家 SAC是一种离策略演员-评论家方法，在标准强化学习目标函数中加入熵奖励，以鼓励智能体继续进行探索。SAC采用双Q网络来减少过高估计偏差，并结合可学习的温度参数 α，以平衡智能体对奖励与探索的重视程度（Haarnoja 等，2018）。SAC目标函数为

$$J(\pi) = \sum_t \mathbb{E}\left[r(s_t, a_t) + \alpha H(\pi(\cdot|s_t))\right],$$

其中，H 是策略的熵。智能体的目标是在最大化奖励的同时保持策略的随机性。在我们的工作中，电路构建的奖励较为稀疏，因此智能体可能要经过许多步骤后才能判断自己是否走在正确的方向上，而SAC的熵正则化有助于在这种情况下维持探索。尽管SAC最初是为连续动作空间设计的，我们将其调整到了离散设置中（Christodoulou，2019）。

3 方法论

3.1 环境与状态编码

我们将算术电路构建建模为单人马尔可夫决策过程（MDP），其中状态是以有向无环图（DAG）表示的部分构建电路，动作对应于选择一种运算和两个已有节点，转移则确定性地向图中添加一个新节点。智能体的目标是尽可能少地经过若干步，达到某个计算出的节点与目标多项式相匹配的状态。

形式化地，令 $\mathcal{G}_t = (V_t, E_t)$ 表示步骤 t 时的电路图，初始节点 $V_0 = \{x_0, \ldots, x_{n-1}, 1\}$ 由输入变量和一个常数组成。在每一步，智能体选择一个动作 $a_t = (\star, v_i, v_j)$，其中 $\star \in \{+, \times\}$ 和 $v_i, v_j \in V_t$，从而生成一个新节点 v_{new}，该节点计算$v_i \star v_j$。转移将图更新为 $V_{t+1} = V_t \cup \{v_{\text{new}}\}$。

每个节点 $v \in V_t$ 由一个 4 维特征向量表示，该向量由表示节点类型（输入变量、常数或运算结果）的 3 位独热编码和一个标量值组成；有向边连接操作数节点与其结果节点，并添加自环以进行消息传递。目标多项式不以符号形式表示，而是采用紧凑的电路编码。此编码将运算类型独热编码、边选择独热编码以及根据参考动作序列得出的最后生成节点指示器拼接起来。动作空间也经过展平，因此每个动作$a_t = (\star, v_i, v_j)$都映射到唯一的整数索引。每一步都会屏蔽无效动作（引用尚未创建的节点）。

3.2 游戏棋盘生成

为生成结构化训练数据，我们构建一个“游戏棋盘”有向无环图（DAG），枚举在 $V_0 = \{x_0, \ldots, x_{n-1}, 1\}$ 个算术运算内可从种子节点到达的所有多项式。每一步都通过加法和乘法组合已有节点对。某个节点出现的步骤对应于计算该多项式所需的最少门数。在此图中，电路对应于从 V_0 中的根节点到目标多项式的一条路径，而最优电路则定义为此类最短路径。

深度	C4-main	C4-pretrain
0	3	1
1	12	2
2	174	9
3	10,862	96
4	8,949	6,548

表1：C4 电路板的电路深度分布。

指标	C4-main（多变量）	C4-pretrain（单变量）
节点	20,000	6,656
边	31,746	20,966
根节点	3	1
具有多条最优电路的节点	18,966 (94.83%)	6,592 (99.04%)
单个节点的最大最优回路数	164	92
单个节点的最大总回路数	1,596	15,296

表 2：C4 游戏棋盘结构和路径效率统计。最优电路是从根节点到节点的最短路径。

并非有向无环图中的所有多项式对训练都同样有用。如果某个多项式在有向无环图中存在多条不同的最短路径（即多条最优电路），我们就将其定义为有趣。以这些多项式为目标，能促使智能体学习有意义的决策，而不是死记硬背一套唯一确定的步骤序列。

表1报告了不同电路深度下的节点分布。大多数节点位于深度 3 和 4，反映出可达多项式数量迅速呈组合式增长。表 2 总结了这些图的结构属性。在双变量（C4-main）和单变量（C4-预训练）棋盘中，绝大多数节点都存在多个最优电路，这表明环境中存在相当大的决策自由度。最终，搜索空间随C呈指数级增长。具有 n 个变量、复杂度为 $\leq C$ 且长度为 $\mathbb{F}_p$ 的不同电路数量，上限为 $\exp(\Theta(C \log(n + p + C)))$。

3.3 近端策略优化 +蒙特卡洛树搜索

我们的主要智能体 CircuitBuilder 将基于图的状态编码与 Transformer 解码器相结合，以生成策略和价值估计。图神经网络编码器处理电路有向无环图。它使用带有残差连接和层归一化的 GCNConv 层，并通过全局平均池化聚合各节点嵌入，得到单个图嵌入向量。目标多项式采用紧凑的独热编码方案进行编码，该方案拼接了参考动作序列中的操作类型指示符、边选择指示符和最后生成节点指示器。随后，通过一个线性层将其投影到相同的嵌入维度。接着，Transformer 解码器对这两种嵌入进行注意力计算：可学习的输出标记充当查询，而图嵌入和目标多项式嵌入则堆叠起来构成解码器记忆。解码器的输出传递给策略头，策略头生成动作空间中各动作的 logits，并将其限制在有效动作范围内；价值头则输出当前状态价值的标量估计。

监督预训练 我们首先使用从游戏棋盘上已知最优电路中提取的（状态、下一步动作）对进行训练，其中电路的每个中间步骤都会成为一个训练样本。模型通过最小化动作预测的交叉熵损失和价值预测的均方误差进行训练。这为强化学习微调前的策略提供了良好的初始化。

近端策略优化微调 经过监督预训练后，我们使用 PPO 对模型进行微调。我们使用广义优势估计（GAE）来计算低方差的优势估计值。

$$\delta_t = r_t + \gamma V(s_{t+1}) - V(s_t), \qquad \hat{A}_t = \delta_t + \gamma\lambda \hat{A}_{t+1}.$$

其中， γ 是折扣因子， λ 控制偏差-方差权衡。我们优化的完整 PPO 损失函数为

$$\mathcal{L}_{\text{PPO}}(\theta) = -\mathbb{E}\left[\min\left(r_t(\theta)\hat{A}_t, \text{clip}(r_t(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t\right)\right] + c_v\text{MSE}(V_\theta(s_t), \hat{R}_t) - c_e\mathbb{E}[\mathcal{H}(\pi_\theta(\cdot|s_t))].$$

这三项分别是裁剪代理目标、价值函数损失，以及鼓励探索的熵奖励。我们采用课程学习策略：训练从多项式复杂度 1 开始；当智能体在滑动窗口内的成功率超过阈值时，便提高复杂度。

蒙特卡洛树搜索集成 在 PPO 数据收集期间，蒙特卡洛树搜索可选择性地引导动作选择。我们采用伯努利混合方案：

$$a_t = \begin{cases} a_t^{\text{MCTS}}, & z_t = 1 \\ a_t^{\pi}, & z_t = 0 \end{cases}, \quad z_t \sim \text{Bernoulli}(p_{\text{mix}}).$$

以 p_{mix} 的概率，智能体会交由蒙特卡洛树搜索规划器决策；否则，它会根据自身策略进行采样。我们的蒙特卡洛树搜索使用神经网络的价值头评估叶节点（采用 AlphaZero 风格），以学习得到的价值估计取代随机模拟。它会返回访问次数最高的动作。这使策略网络在训练期间能够接触到质量更高的轨迹，从而比纯策略采样提高样本效率。

3.3.1 MCTS引导的专家迭代

在每个时间步 t，蒙特卡洛树搜索都由当前的策略价值网络引导，并生成访问次数分布 π_{MCTS}。为平衡回合前期的探索与后期的利用，动作选择采用逐渐衰减的温度参数 τ 进行调节：

$$\tau(t) = \tau_{\text{final}} + (\tau_{\text{init}} - \tau_{\text{final}})\max\left(1 - \frac{t}{t_{\text{decay}}}, 0\right),$$

其中 t_{decay} 定义了退火计划。
我们的方法在架构上的一个关键区别，在于近端策略优化重要性采样比率的公式。标准的AlphaZero实现会训练策略头，使其最小化相对于蒙特卡洛树搜索访问次数的交叉熵损失。在我们的近端策略优化公式中，代理目标依赖于概率比率

$$r_t(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{\text{old}}}(a_t|s_t)}.$$

至关重要的是，分母 $\pi_{\theta_{\text{old}}}(a_t|s_t)$ 是数据收集时网络自身策略所赋予的概率，而非根据蒙特卡洛树搜索访问次数得出的概率。如果分母采用蒙特卡洛树搜索概率，比率 $r_t(\theta)$ 将保持在接近 1 的水平，从而导致策略梯度几乎为零。

通过使用网络内部的 logits 计算基准比率，蒙特卡洛树搜索严格充当数据质量增强器——寻找更短、更高效的电路路径——而裁剪PPO目标则为网络参数提供稳定且有意义的更新，使其近似这种改进后的行为。此外，广义优势估计（GAE）的目标使用网络价值头 $V_\theta(s)$进行自举，而非使用蒙特卡洛树搜索价值估计，从而确保在近端策略优化更新期间持续降低方差。

3.4 软演员-评论家

SAC方法与CircuitBuilder共享同一个GNN-Transformer主干网络，但将单个价值头替换为双Q头，输出每个动作的Q值。SAC采用离策略方式，将转移存储在经验回放缓冲区中，并通过Polyak平均执行软目标网络更新。有关完整的更新过程，请参见附录A中的算法1（OpenAI，2018年）。

我们采用与近端策略优化相同的课程学习策略，并增加了一项机制：当成功率低于较低阈值时，复杂度也可以降低，从而避免智能体在超出当前能力的目标上停滞不前。

4 结果

我们在配备 NVIDIA Tesla T4 GPU（16 GB）的亚马逊云科技 EC2 云实例上训练了 PPO+MCTS 和 SAC。两个模型都使用带有 GCNConv 层的图神经网络编码器来处理电路有向无环图，并将其与目标多项式嵌入融合，然后输出策略 logits 和价值估计。对于两个变量和三个变量的情况，我们分别在 $C = 5$和 $C = 6$ 处、跨越 $\mathbb{F}_5$ 对固定复杂度目标进行评估；每个检查点使用近贪心蒙特卡洛树搜索（$\tau = 0.1$）进行 1000 个留出回合的测试。

对于目标多项式 f，如果智能体生成的算术电路的输出多项式等于 f，我们就判定成功率 。表 3 报告了近端策略优化 + 蒙特卡洛树搜索在 200 次训练迭代、批量大小为 256 时的评估指标。

表 3：近端策略优化+蒙特卡洛树搜索在 $\mathbb{F}_5$ 上的评估（每个检查点进行 1000 个回合，使用带有 $\tau = 0.1$的近贪心蒙特卡洛树搜索）。成功率以百分比表示；熵为迭代 200 时的策略熵。

变量	Task	迭代 50	迭代 100	迭代 150	迭代 200	熵
$n = 2$	$C = 5$成功率（%）	29.40	26.50	27.40	**34.90**	3.28
	$C = 6$成功率（%）	26.00	27.30	27.60	**35.80**	3.21
	$C = 5$平均奖励	2.47	2.22	2.34	3.08	—
	$C = 6$平均奖励	2.19	2.30	2.31	3.20	—
$n = 3$	$C = 5$成功率 (%)	17.70	20.90	26.50	**27.30**	3.65
	$C = 6$成功率 (%)	16.40	**23.50**	20.90	19.90	3.72
	$C = 5$平均奖励	1.34	1.58	2.21	2.30	—
	$C = 6$平均奖励	1.18	1.93	1.67	1.53	—

对于 $n = 2$ 个变量，PPO+蒙特卡洛树搜索在第 200 次迭代时，在 $C = 5$ 和 $C = 6$上均达到 $\sim$35% 的成功率；熵从 $\sim$3.3 略微下降至 $\sim$3.2，表明策略逐渐变得更明确，但并未崩溃。
扩展到 $n = 3$明显更难：目标空间从 $7^2 = 49$ 增长到 $7^3 = 343$ 个系数。在$C = 5$，智能体的表现仍然单调提升至 27.3%，但 $C = 6$ 在迭代 100 时达到峰值（23.5%）后开始下降，这表明在更大的搜索空间中存在训练不稳定性。$n = 3$ 的更高熵（$\sim$3.7 对比 $\sim$3.2）反映了其对扩展后动作空间的更大不确定性。

表 4：SAC 在 $\mathbb{F}_5$ 上的评估（每个检查点进行 1000 个回合，采用贪心策略）。成功率以百分比表示；熵为迭代 200 时的策略熵。

变量	Task	迭代 50	迭代 100	迭代 150	迭代 200	熵
$n = 2$	$C = 5$成功率（%）	48.30	55.60	53.20	**57.80**	1.85
	$C = 6$成功率（%）	45.10	**49.70**	48.30	46.30	1.88
	$C = 5$平均奖励	3.41	4.14	3.90	4.37	—
	$C = 6$平均奖励	3.09	3.52	3.39	3.19	—
$n = 3$	$C = 5$成功率 (%)	5.40	5.00	6.20	**10.70**	2.23
	$C = 6$成功率 (%)	4.30	4.30	5.60	**9.40**	2.22
	$C = 5$平均奖励	−0.99	−1.03	−0.89	−0.45	—
	$C = 6$平均奖励	−1.10	−1.09	−0.95	−0.58	—

表 4 报告了 SAC 在相同的 200 次训练迭代、批量大小为 256 时的评估指标。对于 $n = 2$，SAC 表现尚可，并在迭代 200 时取得最佳结果：在 $C = 5$ 目标上的成功率为 57.8%，在 $C = 6$ 上为 46.3%。平均奖励始终为正，且在训练过程中总体有所提升，这表明 SAC 能够在二变量设定中学习有效的符号构造策略。正如预期的那样，在 $C = 6$ 上的表现仍低于 $C = 5$，这反映出搜索更复杂目标的难度更大。
然而，对于 $n = 3$，性能大幅下降。即使在迭代 200 时，SAC 在 $C = 5$ 上的成功率也只有 10.7%，在 $C = 6$上也只有 9.4%；尽管有所改善，平均奖励仍为负值。与 $C = 5$和 $C = 6$之间相对较小的差距相比，从 $n = 2$到 $n = 3$的大幅下降表明，增加变量数量是一个更重要的

困难来源。这表明，未来的改进应着重于更好的探索、更强的表征，以及能够更有效地扩展到更高维多项式空间的训练策略。

讨论与未来方向根据更新后的结果，在二变量设定中，SAC 的表现显著优于 PPO+蒙特卡洛树搜索。这表明，在当前设置下，SAC 更善于将训练经验转化为成功的符号构造策略。其平均奖励也随着训练不断提高，显示出稳步的学习进展；不过，熵的下降表明，随着训练推进，探索变得愈发集中。值得注意的是，与 SAC 相比，采用学习型树搜索的 PPO+蒙特卡洛树搜索扩展到三个变量时更加平稳，这表明随着问题维度增加，通过蒙特卡洛树搜索进行显式规划可能会变得愈发重要。

$C = 5$ 与 $C = 6$ 之间仍存在差距，这反映了随着电路复杂度增加，符号搜索空间迅速扩大的现象。即使只增加一次运算，可达表达式的数量也会大幅增加，使智能体更难在固定时域内稳定地找到成功的构造。未来的工作包括将训练扩展到复杂度更高的目标（$C \geq 7$）、研究适用于更难实例的更强探索策略，以及在多个复杂度级别上引入课程学习。另一个重要方向是改进多项式和电路表示，使智能体能够在搜索空间扩大时更有效地泛化。

我们的实验仍然聚焦于相对较小的电路，包含三个变量，目标复杂度最高为六个门。尽管这一设定远小于代数复杂性理论所研究的范围，但它提供了一个可控的环境，用于检验强化学习能否学会有意义的电路构造策略。本阶段的目的是了解小型符号问题中的学习动态，然后再扩展到更大的电路、更多变量以及更丰富的多项式族。

致谢

本项目是 UW Math AI Lab 的一部分。我们感谢 UW eScience School 提供计算资源。中央处理器和图形处理器计算部分使用了 UW eScience School 和 UW IT 提供的亚马逊云科技额度，另一部分则使用了由 UW Student Technology Fee Committee 资助的 UW Research Computing Club。部分代码库是在 GitHub Copilot 的帮助下生成的。

参考文献

Peter B¨urgisser、Michael Clausen 和 M. Amin Shokrollahi。代数复杂性理论。在 *Thomas Lickteig* 的合作下，《Grundlehren Math. Wiss.》第 315 卷，柏林：Springer，1997年。ISBN3-540-60582-7。

Petros Christodoulou。《离散动作设置下的软演员-评论家》，2019年。网址：https://arxiv.org/abs/1910.07207。

Tuomas Haarnoja、Aurick Zhou、Pieter Abbeel 和 Sergey Levine。《软演员-评论家：采用随机演员的离策略最大熵深度强化学习》，2018年。网址：https: //arxiv.org/abs/1801.01290。

William George Horner。XXI. 一种通过连续逼近求解任意阶数值方程的新方法。《伦敦皇家学会哲学汇刊》，(109):308–335，1819年12月。ISSN 0261-0523。doi: 10.1098/rstl.1819.0023。网址：https://doi.org/10. 1098/rstl.1819.0023。

ˇ Thomas Hubert、Rishi Mehta、Laurent Sartran、Mikl´os Z. Horv´ath、Goran Zuˇzi´c、Eric Wieser、Aja Huang、Julian Schrittwieser、Yannick Schroecker、Hussain Masoom、Ottavia Bertolli、Tom Za- havy、Amol Mandhane、Jessica Yung、Iuliya Beloshapka、Borja Ibarz、Vivek Veeriah、Lei Yu、Oliver Nash、Paul Lezeau、Salvatore Mercuri、Calle S¨onne、Bhavik Mehta、Alex Davies、Daniel Zheng、Fabian Pedregosa、Yin Li、Ingrid von Glehn、Mark Rowland、Samuel Albanie、Ameya Vel- ingker、Simon Schmitt、Edward Lockhart、Edward Hughes、Henryk Michalewski、Nicolas Son- nerat、Demis Hassabis、Pushmeet Kohli 和 David Silver。通过强化学习实现奥林匹克级形式化数学推理。《自然》，2025年11月。ISSN 1476-4687。doi: 10.1038/ s41586-025-09833-y。网址：https://doi.org/10.1038/s41586-025-09833-y。

Levente Kocsis 和 Csaba Szepesv´ari。基于多臂老虎机的蒙特卡洛规划。载于 Johannes F¨urnkranz、Tobias Scheffer 和 Myra Spiliopoulou（编），《机器学习：*ECML 2006*》，第 282–293 页，柏林、海德堡，2006年。Springer Berlin Heidelberg。ISBN 978-3-540-46056-5。

OpenAI。软演员-评论家（SAC），2018年。网址：https://spinningup.openai.com/en/latest/algorithms/sac.html。访问日期：2026-02-09。

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。

Amir Shpilka 和 Amir Yehudayoff。算术电路：近期研究成果综述及未解决问题。《理论计算机科学基础与趋势》，5(3-4)：207–388，2009年。ISSN 1551-305X。doi：10.1561/0400000039。

David Silver、Thomas Hubert、Julian Schrittwieser、Ioannis Antonoglou、Matthew Lai、Arthur Guez、Marc Lanctot、Laurent Sifre、Dharshan Kumaran、Thore Graepel、Timothy Lillicrap、Karen Si-monyan 和 Demis Hassabis。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12月。ISSN 0036-8075，1095-9203。doi: 10.1126/science.aar6404。URL https://www.science.org/doi/10.1126/science.aar6404。

L. G. Valiant。代数中的完备性类。载于第十一届 *ACM* 计算理论年度研讨会论文集，STOC ’79，第 249–261 页，美国纽约州纽约市，1979年。计算机协会。ISBN 9781450374385。doi: 10.1145/800135.804419。URL https://doi.org/10.1145/800135.804419。

L.G. Valiant. 计算永久式的复杂度。《理论计算机科学》，8(2):189–201，1979年。ISSN 0304-3975。doi: https://doi.org/10.1016/0304-3975(79) 90044-6。URL https://www.sciencedirect.com/science/article/pii/ 0304397579900446。

A 算法

算法1离散软演员-评论家

需要： 经验回放缓冲区 $\mathcal{B}$、策略网络 π_θ、双Q网络 $Q_{\theta,1}$，$Q_{\theta,2}$、目标网络 $Q_{\bar{\theta},1}$，$Q_{\bar{\theta},2}$，温度参数 α，蒙特卡洛树搜索系数 λ_{mcts}，软更新率 τ

对于每个更新步骤**执行**

采样小批量 $\{(s_t, a_t, r_t, s_{t+1}, d_t, \pi_t^{\text{MCTS}}, m_t)\} \sim \mathcal{B}$

// 在有效动作上计算软价值目标

$V_{\bar{\theta}}(s') \leftarrow \sum_{a \in \mathcal{A}_{\text{有效}}(s')} \pi_\theta(a|s') \left(\text{最小值}_i Q_{\bar{\theta},i}(s', a) - \alpha \text{对数} \pi_\theta(a|s')\right)$

$y_t \leftarrow r_t + \gamma(1 - d_t) V_{\bar{\theta}}(s_{t+1})$

// 更新双Q网络

$\mathcal{L}_Q \leftarrow \text{均方误差}(Q_{\theta,1}(s_t, a_t)\ y_t) + \text{均方误差}(Q_{\theta,2}(s_t, a_t)\ y_t)$

// 更新有效动作上的策略

$\mathcal{L}_\pi \leftarrow \mathbb{E}_{s_t}\left[\sum_{a \in \mathcal{A}_{\text{有效}}} \pi_\theta(a|s_t)\left(\alpha \text{对数} \pi_\theta(a|s_t) - \text{最小值}_i Q_{\theta,i}(s_t, a)\right)\right]$

如果蒙特卡洛树搜索分布可用对数 $\pi_\theta(a|s_t)$**则**

$\mathcal{L}_{\text{CE}} \leftarrow \mathbb{E}_{s_t}\left[\sum_a \pi_t^{\text{MCTS}}(a|s_t) \left(m_t \pi_\theta(a|s_t)\right)\right]$

else

$\mathcal{L}_{\text{CE}} \leftarrow 0$

结束条件判断

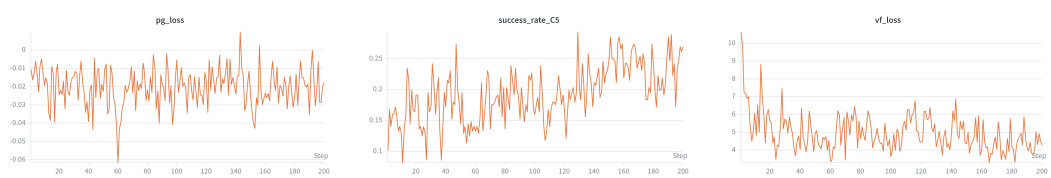
$\mathcal{L}_{\text{total}} \leftarrow \mathcal{L}_Q + \mathcal{L}_\pi + \lambda_{\text{mcts}} \mathcal{L}_{\text{CE}}$

通过最小化 $\mathcal{L}_{\text{total}}$来更新 θ

$\theta \leftarrow (1 - \tau)\bar{\theta} + \tau\theta$

结束循环

B 近端策略优化+蒙特卡洛树搜索训练图



(a) 策略梯度损失 (b) 成功率 (c) 价值函数损失

图 2：复杂度为 5 时的近端策略优化+蒙特卡洛树搜索训练曲线。



(a) 策略梯度损失 (b) 成功率 (c) 价值函数损失

图 3：复杂度为 6 时，近端策略优化+蒙特卡洛树搜索的训练图。

C 软演员-评论家训练图

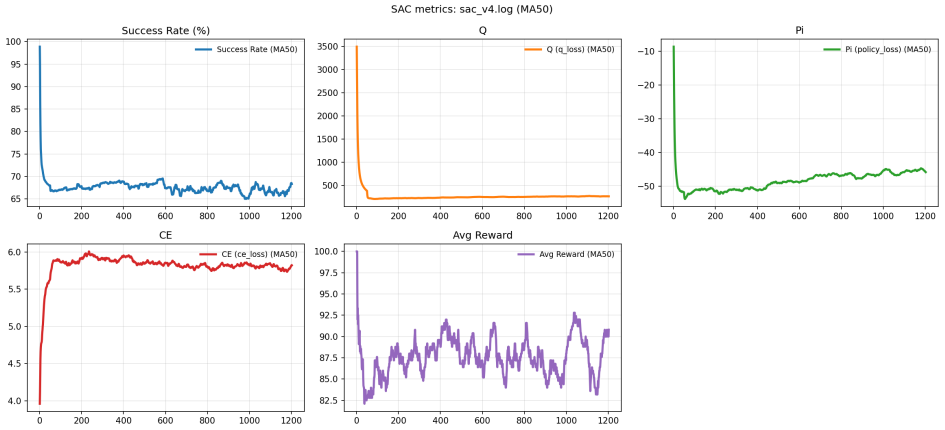


图 4：复杂度等级 1–4 下随时间变化的软演员-评论家训练指标（MA50）

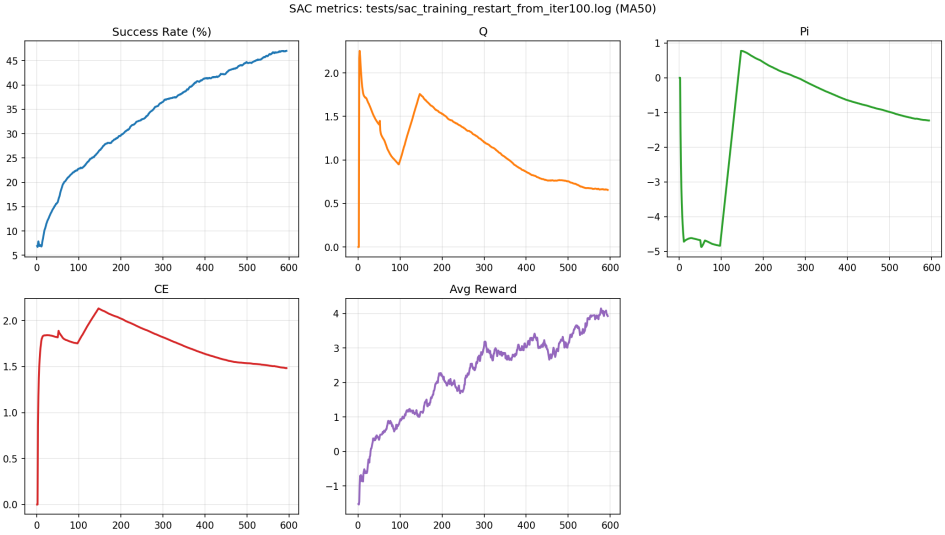


图 5：复杂度 5 下随时间变化的软演员-评论家训练指标（MA50）