重庆理工大学学报 自然科学
Journal of Chongqing University of Technology(Natural Science)

一种结合策略价值网络的五子棋自博弈方法研究

刘 溜, 张小川, 彭丽蓉

引用本文:
刘 溜, 张小川, 彭丽蓉. 一种结合策略价值网络的五子棋自博弈方法研究[J]. 重庆理工大学学报（自然科学）, 2022, 36(12): 129-135.

重庆理工大学学报(自然科学)
Journal of Chongqing University of Technology(Natural Science)
2022 年 第 36 卷 第 12 期
Vol. 36 No. 12 2022
doi: 10.3969/j. issn. 1674-8425(z). 2022. 12. 015

“机器博弈”专栏(专栏主编:张小川 重庆理工大学 教授)

一种结合策略价值网络的五子棋自博弈方法研究

刘 溜[1],张小川[1],彭丽蓉[2,3],田 震[4],万家强[1],任 越[1]

(1. 重庆理工大学 两江人工智能学院, 重庆 401135;
2. 重庆理工大学 人工智能系统研究所, 重庆 400054;
3. 重庆工业职业技术学院 人工智能与大数据学院, 重庆 401120;
4. 重庆市南开两江中学校, 重庆 401135)

摘 要:针对传统蒙特卡洛树搜索算法存在“难以在节点的探索和利用之间做出平衡;难以聚焦重要搜索分支”等问题,提出使用策略价值网络完成棋局评估与落子着法生成,将策略价值网络与蒙特卡洛树搜索相结合。策略价值网络指导搜索树的展开,搜索结果用以持续更新网络参数,形成一种自博弈方法,在多轮自博弈中实现算法的迭代优化。实验表明:相较于各种经典搜索算法,所提算法在平均落子时间上降低了约 95%,平均对局胜率达到 80% 以上。

关 键 词:蒙特卡洛树搜索;深度神经网络;五子棋计算机博弈;自博弈

中图分类号:TP301 **文献标识码**:A **文章编号**:1674 - 8425(2022)12 - 0129 - 07

0 引言

在人工智能的发展历程中,人们一直尝试赋予计算机“思考”的能力。为此,科学家尝试通过给计算机编程、设计博弈系统、运行棋类博弈游戏,期望实现计算机模仿人类下棋[1]。那么,如何构建计算机博弈系统呢? 首先,设计合适的数据结构,以表达棋局局面信息;其次,数字化规则,设计合法的落子着法;最后,构造一个全局性的博弈策略,以期发现能使己方收益最大化的着法。

计算机博弈系统可以通过构造博弈树来完成博弈行为。19 世纪 50 年代,香农提出了计算机博弈的核心思想:通过构建完整的博弈树,搜索当前局面的最佳着法[2]。但在实际博弈过程中,却难以构造一颗理想的博弈树。主要原因是对于博弈中的某个局面,可行的着法数通常较大,而且随着博弈进程推进,这个数字还会呈指数级增加。而发现能决定胜负的着法,理论上来讲,最好就是穷

收稿日期:2022 - 10 - 19
基金项目:国家自然科学基金项目(60443004)
作者简介:刘溜,男,硕士研究生,主要从事机器博弈、机器学习研究,E-mail:2236142970@ qq. com;通讯作者 彭丽蓉,女,副教授,主要从事计算机博弈、软件工程研究,E-mail:28011734@ qq. com。

本文引用格式:刘溜,张小川,彭丽蓉,等. 一种结合策略价值网络的五子棋自博弈方法研究[J]. 重庆理工大学学报(自然科学),2022,36(12):129 - 135.

Citation format:LIU Liu, ZHANG Xiaochuan, PENG Lirong, et al. Research on a self-play method of Gobang combined with a strategic value network[J]. Journal of Chongqing University of Technology(Natural Science),2022,36(12):129 - 135.

尽所有可行着法。在强实时、高对抗的博弈进程中,计算资源的有限性决定了穷举法基本是不可能的方法。

传统蒙特卡洛树搜索摒弃穷举思想,结合了模拟采样的随机性和树搜索的准确性,对各分支的探索权重进行评估,使计算资源集中在重要分支上。但此算法采用随机走子策略模拟对局,在没有被访问足够多的次数的情况下,不能够对关键树节点进行可靠评估。导致此算法效率很低,一定时间内即使是在中等复杂度的博弈中也难以给出优质决策。

为了克服上述缺点,本文提出通过结合深度神经网络改进蒙特卡洛树搜索算法,设计了一种五子棋计算机博弈深度强化学习算法;其中,通过神经网络拟合局面评估函数和落子概率分布的方法,引导蒙特卡洛树搜索方向,提升搜索速度;同时,基于自博弈强化学习训练框架,实现了神经网络的迭代优化,加强搜索强度。

1 传统蒙特卡洛树搜索

蒙特卡洛是一种统计实验方法,利用事件发生频率近似事件发生概率。蒙特卡洛树搜索通过蒙特卡洛方法逐步遍历和扩展博弈树:树中节点的遍历是基于对局面估值的贪心利用;树外节点的扩展则是通过随机策略进行多次快速走子模拟,用胜负结果近似局面评估[3]。如图 1 所示,蒙特卡洛树搜索算法共有 4 个步骤:选择、扩展、模拟、回溯。

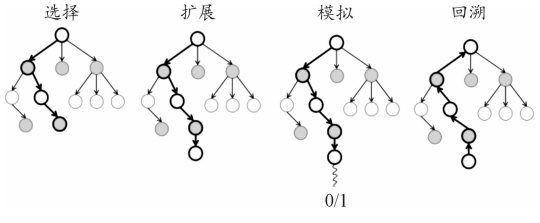

图 1 蒙特卡洛树搜索的四步图

步骤 1 选择:以当前局面为根节点,完成从根节点到达叶子节点之间的路径选择。通过引入上限置信区间算法(UCB),进行节点选择的同时考虑平衡探索次数和局面估值[4]。

$$UCB = V_n + C\sqrt{\frac{2\log\sum_{i=1}^{n} T_i}{T_n}} \tag{1}$$

式中:V_n 为节点估值,T_n 为节点探索次数,C 为平衡因子,C 越大则越偏向探索次数少的节点,反之则偏向局面价值高的已探索节点。

步骤 2 扩展:当到达某个未被探索过的叶子节点时,列举其下所有可行着法。否则进入模拟阶段。

步骤 3 模拟:采用随机策略,模拟双方对弈直到游戏结束产生胜负结果。

步骤 4 回溯:将胜负结果从叶子节点层层向上回溯到根节点,更新途经节点的探索次数以及估值。

2 结合神经网络的蒙特卡洛树搜索

结合深度神经网络和蒙特卡洛树搜索,提出一种自博弈学习方式,从零开始学习五子棋。

2.1 策略价值网络生成棋局着法

人类博弈时,首先需评估局面预测己方胜负,其次选择有利落子。本文模拟人类思考过程,构建策略价值网络返回可行落子概率以及局面评分。

2.1.1 策略价值网络的输入特征及结构设计

局面表示是计算机理解博弈过程的先决条件。在深度学习理论中,这一步也被称为特征提取。有学者将人类知识融入特征提取中,比如 DeepMind 曾融入围棋中“气”“征子”等概念,帮助 AlphaGo 理解围棋[5],却最后掣肘了 AlphaGo 棋力提升。

提取了落子分布以及决策方信息。如图 2 所示,使用矩阵表示棋盘,用 1 代表此位置已有落子,0 表示暂无落子,全 1 或者全 0 的棋盘代表此时轮到执黑方或执白方落子。

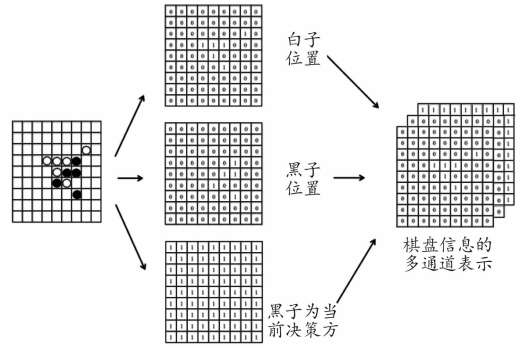

图 2 棋局信息表示实例

网络起始为3层公共卷积网络,分别使用32、64、128个3×3的卷积核,均使用ReLU激活函数,如图3所示。随后分成policy和value 2个输出:policy输出端,先用4个1×1的卷积核降维,紧接一个全连接层,最终使用softmax函数输出棋盘上每个位置的落子概率;value输出端,先用2个1×1的卷积核进行降维,再连续接2个全连接层,最后使用tanh函数输出[-1,1]之间的局面评分。

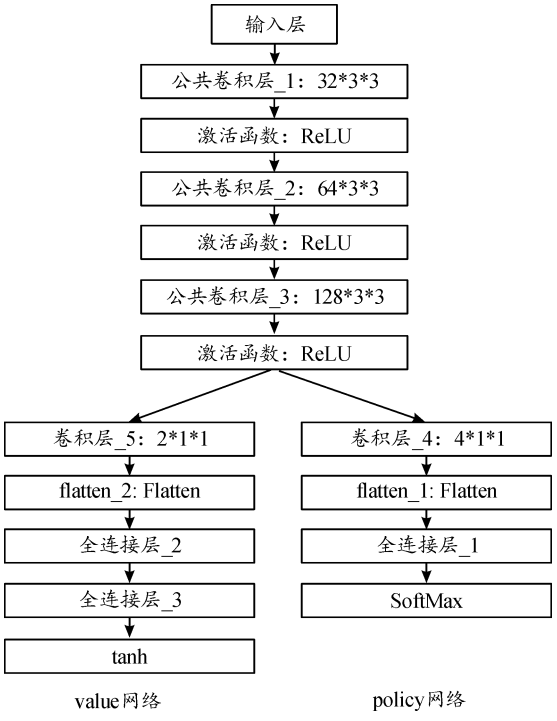

图3　策略价值网络结构设计框图

2.1.2　策略价值网络的损失函数

策略价值网络的输入是棋局描述s,输出是落子概率估计p以及胜负估计v。自博弈的过程中,会探索各种局面s、实际落子概率π以及胜负结果z,收集(s,π,z)数据用于后面的网络更新。训练目标是让策略价值网络输出的p和v更加接近经过蒙特卡洛树搜索采样模拟后得到的π和z[6]。如式(2)所示,本文将联合损失作为该网络优化的目标函数。

l_{v-z}代表局面胜负估计v和实际蒙特卡洛树搜索返回值z的均方误差,根据策略梯度下降算法,需要最小化目标策略的评估函数$J(\pi_\theta)$的相反数$-J(\pi_\theta)$,即p和π的交叉熵误差,其中θ表示策略价值网络参数。

$$l = l_{v-z} - J(\pi_\theta) = (z-v)^2 - \pi^T \log p \quad (2)$$

为了缓解网络过拟合的问题,考虑引入正则化。引起过拟合的主要原因在于模型复杂,参数过多而正则化的基本思想是引入惩罚项以减少参数量级。

最终的损失函数为:

$$l = (z-v)^2 - \pi^T \log p + c\|\theta\|^2 \quad (3)$$

2.2　自博弈中蒙特卡洛树搜索提升着法策略

人类通过推演落子后的棋局变化评估落子价值。而蒙特卡洛树搜索可凭借强大的采样能力,用模拟统计结果近似落子概率,达到棋局推演的目的。如图4所示,可以将蒙特卡洛树搜索作为一个策略提升器,校准策略价值网络输出的落子概率。

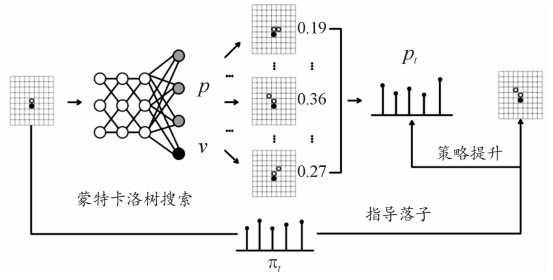

图4　策略提升器—蒙特卡洛树搜索图

一局完整的自博弈,从某一棋盘状态s_1,经历$s_2,s_3,\cdots$一直到结束状态s_T。每一个s_t下,都会执行一次完整的蒙特卡洛树搜索,搜索过程中使用策略价值网络进行辅助,最终返回s_t下不同位置的落子概率π_t,自博弈的真实走棋根据落子策略π_t决定。到达s_T后,得到这一局结果z,一般是-1、0、1之一,分别对应输、平和赢。对于自博弈的每一步,都要保存一个三元组(s_t,π_t,z_t)作为一个训练样本。自博弈中最关键的一步就是在s_t下,如何执行蒙特卡洛树搜索,并返回不同位置的落子概率。改进的蒙特卡洛树搜索具体分为以下3个阶段:选择阶段、扩展模拟阶段、回溯阶段。通过反复执行这3个步骤,逐渐建立一棵树,然后根据树中存储的统计信息得到落子的概率分布。

搜索树中的节点记录棋局状态,节点访问次数$N(s,a)$表示在状态s下采取着a法的频率;累计行动价值$W(s,a)$表示(s,a)局面动作在采样模拟中获得的总收益;平均行动价值$Q(s,a)$是$W(s,a)$与$N(s,a)$的比值;先验概率$P(s,a)$由策

略价值网络给出,表示对落子概率的估计[7]。

2.2.1 节点选择

节点选择是指从合法动作集中选择一个着法。关于节点选择策略,受 UCB 式(1)的启发,设置变量 $U(s,a)$ 用以平衡节点的探索和利用。

$$U(s,a) = Q(s,a) + c_{\text{quct}}p(s,a)\frac{\sqrt{\sum_{b\in Actions}N(s,b)}}{1+N(s,a)} \tag{4}$$

式(4)包括两部分:$Q(s,a)$ 代表着法 a 在 s 下获得的平均价值,是对局面评分贪心利用;$P(s,a)$ 是策略价值网络给的着法指导;公式最右边是计算在 s 下选择 a 的频率比重;c_{quct} 是超参数,用来调整探索和利用的比例。策略开始会偏向选择次数少的节点,这是对探索的鼓励避免错过价值高的着法,随着模拟次数越来越多,策略会偏向那些 $Q(s,a)$ 高的节点,将计算资源集中在最佳探索分支上。

2.2.2 节点扩展和模拟

依据节点选择策略,不断进行树内节点选择,直到遇到一个从未探索过的叶子节点 s_l。如果 s_l 是终止节点则直接进入节点回溯阶段,否则需要将 s_l 下所有的分支节点添加到搜索树中,并分别保存采取落子动作后进入的新局面。

同时策略价值网络会给出 s_l 下的每个合法落子概率 p_l 以及局面评分 v_l,p_l 存储在 s_l 到各个分支节点的边中。由于各分支节点都是新加入到搜索树中,需要将对应边中的其他变量 $N(s,b)$,$W(s,b)$,$Q(s,b)$ 均作初始化处理,随后进行随机模拟直到游戏结束。

2.2.3 节点回溯

此阶段完成从叶子节点 s_l 到根节点 s_0 路径上的参数更新。这是蒙特卡洛树搜索能集中计算资源于重要分支的原因。假设 s_l 的父节点为 s_t,s_t 到 s_l 之间的落子动作为 d_t,策略价值网络给 s_l 的评分为 v_t。则从 s_l 回溯到 s_t 的参数更新如式(5)—(7)所示。

如果 s_l 是终止节点,v_t 将不会使用策略价值网络给出的胜率评分值,直接使用胜负结果值:-1、0、1 之一;在回溯的过程中,每经过一个节点,v_t 值要进行取反操作,因为博弈树中模拟的是双方博弈的过程,一方的收益对于另外一方则是损失。

$$W(s_t,d_t) = E(s_t,d_t) + v_t \tag{5}$$

$$N(s_t,d_t) = N(s_t,d_t) + 1 \tag{6}$$

$$Q(s_t,d_t) = \frac{W(s_t,d_t)}{N(s_t,d_t)} \tag{7}$$

2.2.4 实际落子选择

将当前局面视为搜索树的根节点 s_0,经过选择、扩展和模拟、回溯,结束一次蒙特卡洛树搜索。而每一次真实落子需要根据经过多次蒙特卡洛树搜索,这也是自博弈特别消耗计算资源的原因。多次采样之后,根节点 s_0 下所有边存储的信息都得到了更新,利用这些信息可以计算 s_0 下的落子概率。如式(8)所示,在 s_0 下采取落子动作 a 的概率为:

$$\pi(a\mid s_0) = \frac{N(s_0,a)^{\frac{1}{\tau}}}{\sum_{c\in Actions}N(s_0,c)^{\frac{1}{\tau}}} \tag{8}$$

受模拟退火算法的启发,设置温度参数 τ 控制蒙特卡洛树搜索收敛到重要分支。最开始 τ 设置为 1,此时落子概率分布较均匀,偏向选择访问次数少的落子动作;随着自博弈持续进行,τ 值逐渐减小为 0,此时访问次数多的落子更加容易被选择。最后选择 $\pi(a\mid s_0)$ 最大的对应着法 a 当作实际落子动作。

3 算法性能对比实验

3.1 实验环境

本实验平台为 Windows 10,运行内存为 8.0 GB,显卡配置为 NVIDIA GeForce GTX 1050Ti,主体代码使用 python 语言实现,神经网络搭建使用 pytorch 深度学习开发框架。

3.2 实验对照算法

为了使实验结果更加可信,本文引入另外 2 种经典搜索算法用作效果对照。

极大极小值是一种适用双人零和博弈的搜索思想。核心是分别站在各博弈方进行搜索,其中一方在可行着法中选择自身利益最大化的落子着法,而对方就要尽可能阻止对方选择该最佳落子[8]。一般极大极小值算法需要通过构造一个局面评估函数,计算出对己方最大、对方最小的着法。分析发现,局面评估函数常常与棋子数、棋子分布及其关键棋型 3 个因素密切相关[9]。本文设

计局面评估函数时,先尝试对不同棋型赋予一个先验权值,再根据棋子位置,赋予另一个先验值[10]。式(8)中 x,y,z 分别表示处于棋盘上"核""边""角"位置上双方棋子数目之差,而 N_1、N_2、N_3 分别表示"活四""冲四""活三"3 种关键棋型数目,如图 5 所示。

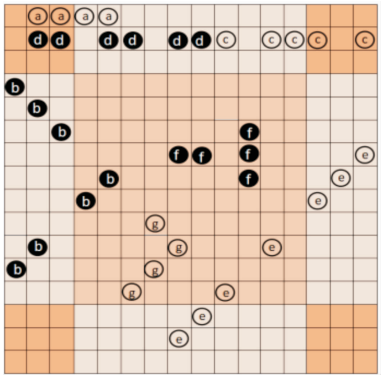

图 5　五子棋重要棋型以及位置分布示意图

$$f(x,y,z,N_{i=1,2,3}) = 3x + 2y + z + 100N_1 + 50N_2 + 20N_3 \quad (9)$$

Alpha-Beta 剪枝算法本质上是一种对极大极小值算法的优化。其核心思想是记录中间节点 Min 的最小值 α,节点 Max 的最大值 β 及其局面估计值 Value,以达到对搜索树裁剪的目的[11]。在 Alpha-Beta 算法中,通常将对 Max 节点层的剪枝称为 Alpha 剪枝,对 Min 层的裁剪称为 Beta 剪枝。如图 6 所示,无论采用哪种剪枝,其核心都是需要比较兄弟节点 *Value* 值、父节点 α 值或者 β 值。

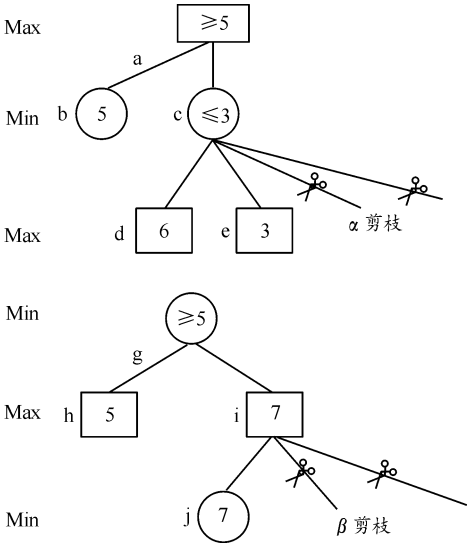

图 6　Alpha-Beta 剪枝过程示意图

从理论上讲,在棋盘中任意一个空白位置都是合理的落子选择点。但是,在五子棋博弈中,落子会相对集中于某个区域,因此,区域内的搜索价值更大。为此,本文引入变尺度思想,将棋盘离散化不同价值区域,对不同区域再赋予 Alpha-Beta 剪枝搜索的不同深度值。

本文用搜索效率和博弈水平 2 个标准,综合比较了极大极小值搜索,Alpha-Beta 剪枝,传统蒙特以及改进蒙特卡洛树搜索 4 种算法的性能[12]。表 1 展示了实验中用到的具体算法类型及重要参数。

同时本文为了验证搜索空间大小对搜索算法的影响,分别设计 9×9、11×11、15×15 三种尺寸的棋盘,作为验证环境。

表 1　算法类型及重要参数

算法序号	算法类型及重要参数
算法 1	极大极小值算法(不加评估函数)
算法 2	极大极小值算法(附加评估函数)
算法 3	传统 Alpha-Beta 剪枝算法(搜索深度=4)
算法 4	传统 Alpha-Beta 剪枝算法(搜索深度=5)
算法 5	区域搜索限定的 Alpha-Beta 剪枝算法(搜索深度=4)
算法 6	结合置信区间上界的蒙特卡洛树搜索(模拟次数=1 000)
算法 7	结合置信区间上界的蒙特卡洛树搜索(模拟次数=2 000)
算法 8	结合置信区间上界的蒙特卡洛树搜索(模拟次数=3 000)
算法 9	结合神经网络的改进蒙特卡洛树搜索

3.3　搜索效率对比

构建和遍历一棵搜索树所耗费的时间即平均落子时间,是体现算法搜索效率的重要指标[13-14]。如图 7 所示,各算法在不同棋盘上,平均落子时间随对弈步数的变化情况。

相较于经典搜索算法,改进后的蒙特卡洛树搜索在搜索效率上有了大幅度的提升,平均落子时间仅仅只有极大极小值算法的 1/196,Alpha-Beta剪枝算法的 1/55,传统蒙特卡洛树搜索的 1/28。此外这种改进算法在不同大小的棋盘下的落子决策时间无明显变化,说明其具有相当的鲁棒性。

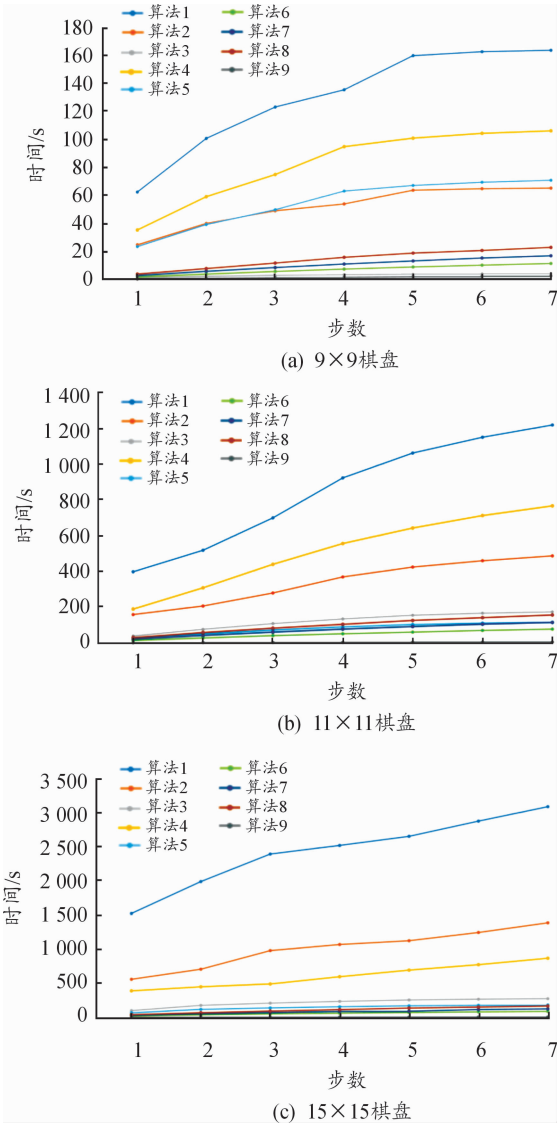

图 7 各算法在不同棋盘上平均落子时间随对弈步数的变化曲线

3.4 博弈水平对比

为了比较各算法间的博弈水平,设计五子棋对局实验。实验中任意 2 种算法之间均设置 100 局比赛,统计各算法的胜负平数据,结果如图 8 所示。对局实验中采用的是比赛标准 15 × 15 的棋盘。

当极大极小值算法并未使用局面评估函数,而是将博弈树完全展开至游戏结束。与其他经典搜索算法相比,对局平均不败率达到了 97%,但是胜率却仅有约 36%。这是由于极大极小值算法的核心是以避免对手获取最大收益,而不是以自己取胜为目标。

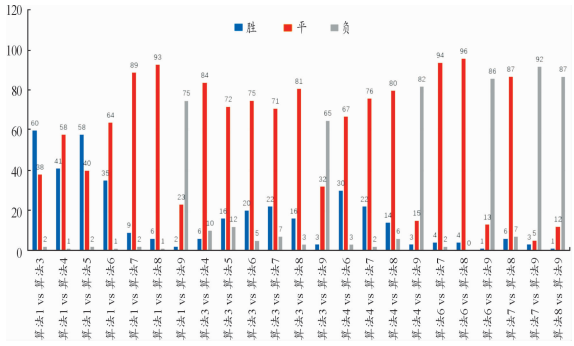

图 8 算法间对局实验结果统计

对于 Alpha-Beta 剪枝算法,实验设置两组搜索深度用于对照,结果显示浅加搜索深度并不能显著提高搜索水平。与其他经典搜索算法相比,搜索深度加 1,胜率仅仅提高了 2%。因为基于人类经验设计的评估函数,并不能保证对每一个局面的评估都来源于真实的胜负反馈。对弈越早期,这种评估上的误差就越大,而且误差还带有积累效应。

在传统蒙特卡洛树搜索的对局中,为了验证模拟次数对其搜索水平的影响,实验设置 1 000 次,2 000 次,3 000 次 3 组参数。通过观察实验结果,发现模拟次数的增加同样不能显著提高其搜索水平。模拟次数增加 1 000 次,平均胜率增加不到 1%,甚至出现增加模拟次数但是胜率反而下降的情况。一定程度上说明了模拟中采用随机落子策略,缺乏先验指导很难收敛到重要分支。

改进的蒙特卡洛树搜索在胜率上均取得了明显的优势。实验结果显示,与极大极小值算法、Alpha-Beta 剪枝算法、传统蒙特卡洛树搜索算法相比,平均胜率分别达到了 75%、84%、87%。

4 结论

以五子棋为研究载体,分析了经典的蒙特卡洛搜索算法的缺点。针对这些缺点,设计出一种结合策略价值网络的蒙特卡洛树搜索,以深度神经网络拟合局面评估函数,用先验落子概率指导蒙特卡洛搜索,形成一种先验知识支持的自博弈方法。与传统的搜索算法相比,改进后的方法无论是在搜索效率,还是博弈水平上,都有较大程度的提高。

参考文献:

[1] 刘贺,张小川,刁志东,等. 一种棋类计算机博弈强化学习智能体的决策依据解释方法[J]. 重庆理工大学学报(自然科学),2021,35(12):140 - 146.

[2] 罗俊仁,张万鹏,苏炯铭,等. 计算机博弈中序贯不完美信息博弈求解研究进展[J/OL]. 控制与决策:1 - 28[2022 - 10 - 12]. DOI:10. 13195/j. kzyjc. 2022. 0698.

[3] 张效见. 五子棋计算机博弈系统的研究与设计[D]. 合肥:安徽大学,2017.

[4] 王亚杰,祁冰枝,张云博,等. 结合神经网络的改进UCT在国际跳棋中的应用[J]. 重庆理工大学学报(自然科学),2021,35(7):259 - 265.

[5] SILVER D,HUANG A,MADDISON C J,et al. Mastering the game of Go with deep neural networks and tree search[J]. Nature,2016,529(7587):484 - 489.

[6] 李大舟,沈雪雁,高巍,等. 一种自学习的智能五子棋算法的设计与实现[J]. 小型微型计算机系统,2020,41(6):1169 - 1175.

[7] 张泽阳. 基于强化学习的完全信息博弈理论研究与实现[D]. 西安:西安电子科技大学,2021.

[8] 任航. 基于知识与树搜索的非完备信息博弈决策的研究与应用[D]. 南昌:南昌大学,2020.

[9] 王鸿菲,王静文,李媛. 基于PVS算法的六子棋博弈系统的研究[J]. 智能计算机与应用,2021,11(2):97 - 100.

[10] 李枫,王彦博. 基于专家系统与DAN网络的围棋局面判断算法[J]. 北华大学学报(自然科学版),2020,21(4):556 - 560.

[11] 张小川,刘溜,陈龙,等. 一种非遗藏族久棋项目计算机博弈智能体的评估方法[J]. 重庆理工大学学报(自然科学),2021,35(12):119 - 126.

[12] 李昊. 五子棋人机博弈算法优化研究与实现[D]. 大连:大连海事大学,2020.

[13] 张小川,王宛宛,彭丽蓉. 一种军棋机器博弈的多棋子协同博弈方法[J]. 智能系统学报,2020,15(2):399 - 404.

[14] 曹风云,赵卫华. 基于Java的五子棋博弈平台研究[J]. 重庆工商大学学报(自然科学版),2021,38(2):10 - 15.

Research on a self-play method of Gobang combined with a strategic value network

LIU Liu[1], ZHANG Xiaochuan[1], PENG Lirong[2,3], TIAN Zhen[4], WAN Jiaqiang[1], REN Yue[1]

(1. School of Artificial Intelligence, Chongqing University of Technology, Chongqing 401135, China; 2. Institute of Artificial Intelligence System, Chongqing University of Technology, Chongqing 400054, China; 3. School of Artificial Intelligence and Big Data, Chongqing Industry Polytechnic College, Chongqing 401120, China; 4. Chongqing Nankai Liangjiang Middle School, Chongqing 401135, China)

Abstract: In view of the problems in the traditional Monte Carlo tree search algorithm, such as "difficulty in balancing the exploration and utilization of nodes, difficulty in focusing on important search branches and so on", this paper proposes a strategic value network combined with Monte Carlo to complete chess evaluation and generation of moves in the game. The application of Monte Carlo is guided by the strategic value network, and the search results are used to continuously update network parameters, so a self-play method is formed to realize iterative optimization of the algorithm in multiple rounds of self-play games. In accordance with the final experiment, in contrast with various classic search algorithms, this method reduces the average dropping time of chess pieces by about 95%, and the average game winning rate reaches more than 80%.

Key words: Monte Carlo tree search; deep neural network; computer-based Gobang game; self-play

(责任编辑　王　欢)