

# 基于人工智能的五子棋搜索算法

□沈大旺

**【内容摘要】**五子棋作为人工智能开发研究对象之一,能够有效通过五子棋博弈解决很多实际问题,同时因为五子棋内涵丰富,极具趣味性,也被作为计算机算法分析的研究对象之一。在人工智能开发过程中用五子棋作为对象,采用多种算法实现人工智能模拟人类思考推算过程,既能够达到博弈效果,又可以进一步完善人工智能模拟系统。为此,本文立足于人工智能五子棋博弈现状,针对其算法和人工智能开发系统,探究如何在新形势下逐步提升人工智能检索算法,促使人工智能系统算法进一步接近人类智慧,在五子棋程序数据结构、评分规则和胜负判断中展示人工智能五子棋检索算法的整个内容。

**【关键词】**人工智能;五子棋;检索算法

**【作者简介】**沈大旺(1981.11~),男,广东高州人;茂名职业技术学院讲师,硕士;研究方向:软件技术、人工智能

自从电子计算机研制成功以后,诸多学者先后投入到人工智能开发中,以期通过电子计算机模拟人类思维进一步挖掘人类智慧,所以在 Claude Shannon 和图灵提出棋类博弈程序后,五子棋就成为科学界人工智能研究的重点之一,新时代中人工智能成为一门新的学科,机器博弈也是人工智能分支之一,很多人工智能开发的实际问题能够在五子棋博弈中得到解决,从而逐步优化五子棋搜索算法,进一步推动人工智能系统的开发。

## 一、人工智能五子棋检索算法研究现状

基于人工智能的五子棋搜索算法是将五子棋作为研究对象,旨在通过五子棋博弈模拟人脑思考过程,达到优良的博弈效果,依据现阶段的人工智能系统研发实验表明,人工智能五子棋程序开发中有多种算法,随着这些算法在新时代的逐步改进,五子棋的程序和博弈过程更加接近人脑思考,现阶段人工智能五子棋程序开发中,重点在于五子棋棋局状态表示、棋局估值函数设计和检索算法中的检索深度、开局以及残局库的使用等,这些技术若是能够达到理想结果,就能进一步促使机器博弈大规模地完善人工智能系统,从而加快人工智能系统的开发,更好地进行人类智慧的研究、模拟和扩张,使得工业向 4.0 时代更进一步。目前人工智能五子棋检索算法的开发,主要是在 Windows 中应用 Visual C++ 6.0 塑造智能五子棋博弈原型系统,借助三维数组法描述存储棋盘的状态,并应用加权值算法展示棋局优势评价,采用  $\alpha$ - $\beta$  剪枝搜索作为最底层的检索算法,然后通过启发规则选择符合棋局发展现状的检索算法。基于人工智能的五子棋搜索算法实现以及系统分析结果显示,采用这种开发和检索算法,比对以往的五子棋程序更加符合人类思维,有效地减少了不能取胜的空间检索过程,缩短了检索时间,有效提升了人工智能五子棋检索算法。从目前电脑游戏中五子棋游戏种类和软件数量的增加,就可以看出现阶段人工智能五子棋搜索算法正在逐步改进,例如谷歌的 AlphaGo 博弈技巧优

势在于实现了深度卷积神经网络与训练,应用好计算机博弈技术、蒙特卡罗算法并在实战中进行检索锻炼,能持续性改进传统的检索算法。

## 二、人工智能五子棋检索算法文献综述

因为使用五子棋进行人工智能以及检索算法的检查,能够在博弈中发现问题,并进一步改进五子棋检索算法,所以不少学者对五子棋博弈进行了深入分析。许南山等(2006 年)针对人工智能五子棋提出了多线程、分布式并行的检索算法,以进一步加快检索速度以及加深检索深度,但缺点还是比较突出的,资源消耗太大不利于开发和研制,违背了集约原则;王长飞(2009 年)则是在智能五子棋算法的设计实现中论述了优化五子棋算法的思路,但是忽略了视野影响和后台利用时间,还有开局库以及残局库等等,也缺乏实践性价值;闫海艇结合 UML 系统框架进行五子棋程序的研发中,介绍如何在 UML 框架实现五子棋设计,但是优化问题没有进行详细分析;董慧颖(2017 年)在多种搜索算法的五子棋博弈算法研究中,介绍了多种检索算法,但是在改进和优化问题中,缺乏明确性论述。因此,目前我国对智能五子棋博弈的关键技术、检索算法等都是从不同角度展开论述和研究,并没有专门从一点着手,综合性探究如何实现人工智能五子棋检索算法的改进。此外,还需要进一步深入分析怎么深化检索深度,以及选择合适的深度进行最优化检索。以下则是针对人工智能五子棋检索算法等的综合性介绍。

(一) 博弈树搜索。虽然博弈树比较复杂,但是从五子棋的特点来讲,这种方法不需要进行较深层次的检索,换言之,博弈树应用不会引发复杂性的问题,因为五子棋规则比较简单,其博弈树能够非常快速地检索。

(二) 启发式估值。这种估值的要求非常高,在开局棋子相近或者是数目相对较少的时候,棋局中会有一些关键点,这些关键点的抢占和反占比较重要。

(三) 棋盘状态的表示。需要从界面和数据结构,针对棋

局在计算机中的表示方法和差生合法作为规则,博弈搜索和评估函数而言,棋盘状态表示不仅与视野有关,而且还决定着棋局走向。

### 三、棋盘情况的估值

针对五子棋的博弈特点来看,需要对棋盘情况进行区域检索和类别划分,针对双方可以落棋但是不能取胜的点,应该缩小检索空间,从而提高反应速度。目前,棋盘状态的表示采用位置表示以及矩阵表示,能将很多棋盘的运算转变为处理器的逻辑指令,极大地提升运算速度,用矩阵进行智能五子棋表示,基本上使用二维数组法,用数字表示棋盘上各个点的位置,目前也就三维数组法,但还在应用开发中,还远远达不到实际需求。三维数组法是分别从四个方向进行估值,然后进行加权求和,实现对复杂组合模型的估算,常见的估算函数有加权估算法,这个估算方法的公式为估值 =  $Qa \times A + Qb \times B + Qc \times C + Qd \times D$ ,其中  $Qa \sim Qd$  是各个因素的系数权重,取值现阶段还没有得到固定值,都是进行估值猜测,然后通过模拟分析将较强的组合保留下来,进行实际博弈和对棋。还有就是模板估值,这种估值方法的核心,是将棋局进行拆分,看作是几种模板的组合,这是棋类人工智能程序设计中优良的估值方法,相当于查表选择一个比较优秀程序或者是棋手进行大量的对局训练,逐步增强判断胜率或评估棋力,逐步改进模板的估值。

### 四、博弈树搜索算法搜索

这种检索算法是最为普遍的一种算法,也是计算机博弈的核心,不断地改进和创新,是人工智能五子棋检索比较活跃的算法和关键技术。因为,实际上计算机难以对所有棋局进行模拟,为了尽可能快速地进行搜索,需要寻求合适的算法依据棋局实际情况进行内容检索,进而快速完成棋局下一步乃至多步的模拟,促使当前棋局保持在最佳的对弈状态。故而需要从当前状态寻求怎样在最短时间内达到最佳状态,这个过程中最佳路径的第一步就成了关键,为达到最佳着法,第一步棋是重中之重,检索算法影响着博弈程序的优劣,基本检索算法中包含了优先检索、极小检索以及启发式检索,使用估值函数针对博弈树中的每一个局面进行加权估算后,就能在极大和极小检索中寻求最佳的博弈树路线。博弈树检索算法中,基本上都是在博弈树中进行检索,每棵博弈树的甲方胜利局面值为 30,000,则乙方胜利局面值为 -30,000,和局则为 0,所以当轮到甲走的时候,就会自动地选择节点值最大的走法,乙方则是会偏向于子节点值最小的走法,故而中间节点会出现以下的计算步骤:第一,若是此节点所对应的局面归甲方下子,则节点值是所有子节点中值最大的那个,如果节点所对应的下子方是乙方,则选择子节点中最小值的走法。第二,子节点值判断是采用最大、最小进行极值判断,因此博弈树中检索最好的走法应该是以极大、极小值为主,借助估值函数对特定局面进行评分,进而快速地在博弈树中为机器选择出最佳的走法。

但是现阶段极大极小检索过程中存在着一定的数据冗余,需要通过删除冗余数据,缩小检索空间,因此需要用到  $\alpha - \beta$  剪枝搜索,这个检索算法用 C# 语言可以描述成以下几个

个程序逻辑过程:

```
int AlphaBeta( int depth, int alpha, int beta)
{
    if( depth == 0) {
        return Evaluate();
    }
    GenerateLegalMoves();
    while( MovesLeft() ) {
        MakeNextMove();
        val = - AlphaBeta( depth - 1, - beta, - alpha);
        UnmakeMove();
        if( val >= beta) {
            return beta;
        }
        if( val > alpha) {
            alpha = val;
        }
    }
    return alpha;
}
```

这就是整个剪枝算法的过程,在这个过程中剪枝算法的效率与子节点扩展排序具有较强的相关性,所以在每一步算法中都要得到最好的节点扩展顺序,这还需要进一步深入研究,才能在实际使用中不断地改进  $\alpha - \beta$  剪枝算法。虽然,目前  $\alpha - \beta$  剪枝算法和单纯的极大极小算法对比是有较大的改进,但依旧还需要联合其他检索算法,才能在检索深度加深的时候,获取较快的反应速度,所以不少学者将眼光瞄向了遗传算法以解决当前  $\alpha - \beta$  剪枝算法中存在的问题。

### 五、结语

综上所述,目前应用在人工智能五子棋棋技中的算法有置换表技术检索算法与  $\alpha - \beta$  剪枝算法等,两者相互合作,引入迭代加深深化检索深度和应用局部搜索方法选定合适深度进行检索,可以进一步提升程序棋技,且在此基础上使用 Monte Carlo 方法和深度学习方法,还能使得人工智能系统的棋艺更进一步。

### 【参考文献】

- [1] 郑培铭,何丽. 基于计算机博弈的五子棋 AI 设计[J]. 电脑知识与技术, 2016, 12(33): 80 ~ 81, 90
- [2] 王云霞. 智能五子棋博弈算法研究[J]. 江苏技术师范学院学报, 2013, 19(2): 62 ~ 66
- [3] 朱龙梅. 浅论人工智能启发式搜索策略的研究[J]. 电子设计工程, 2013, 21(16): 61 ~ 64
- [4] 董慧颖,王杨. 多种搜索算法的五子棋博弈算法研究[J]. 沈阳理工大学学报, 2017, 2
- [5] 黄继平,张栋,苗华. 六子棋智能博弈系统的研究与实现[J]. 电脑知识与技术, 2009, 9
- [6] 张颖. 六子棋计算机博弈及其系统的研究与优化[D]. 重庆大学, 2008, 4