

# 基于博弈树算法的五子棋游戏的设计与实现

曹晓燕 王 辰

(南京工业职业技术大学 计算机与软件学院, 江苏 南京 210023)

**摘 要:** 针对当前五子棋游戏软件存在的自主学习能力有限、用户体验感差等问题, 笔者设计实现了一个基于博弈树搜索模型的五子棋游戏系统。在深入分析棋局局面搜索基本原理的基础上, 设计了五子棋棋局局面最优搜索与评估策略, 研究了基于平衡权值策略的博弈树搜索评估理论模型, 详细讨论了五子棋游戏系统的设计过程, 给出了五子棋游戏软件的具体实现。本文实现的五子棋游戏系统操作简洁、自主学习能力强且用户体验感好, 研究工作具有重要的现实意义。

**关键词:** 博弈树; 五子棋; 权值平衡; 游戏系统

中图分类号: TP317 文献标识码: A 文章编号: 1003-9767(2021)17-031-03

## Design and Implementation of Gomoku Based on Game Tree Search Algorithm

CAO Xiaoyan, WANG Chen

(School of Computer and Software Engineering, Nanjing Vocational University of Industry Technology, Nanjing Jiangsu 210023, China)

**Abstract:** Aiming at the problems of limited autonomous learning ability and poor user experience in Gobang game software, a Gobang game system based on game tree search model is designed and implemented. Based on the in-depth analysis of the basic principle of Gobang situation search, the optimal search strategy of Gobang situation is designed, the game tree search model based on weight balance strategy is studied, the design process of Gobang game system is discussed in detail, and the specific implementation of Gobang game software is given. The Gobang game system implemented in this paper has simple operation, strong autonomous learning ability and good user experience, the research work has important practical significance.

**Keywords:** game tree; Gomoku; weight balance; game system

### 0 引言

五子棋是一种出棋规则简单、内在逻辑性强的棋类游戏, 是深受人们欢迎的智力游戏之一<sup>[1]</sup>。五子棋游戏的实现以局面评估函数的优化设计为核心, 并通过不同的最优搜索算法搜索棋局局面。由于五子棋局面复杂度高、评估难度大, 因此其最优局面搜索和评估函数的设计比较难<sup>[2]</sup>。目前, 五子棋人机博弈游戏大多存在自主学习能力有限、用户体验感差的问题, 如何设计实现操作简洁、自主学习能力优、趣味性强的五子棋游戏软件是当前研究的热点问题<sup>[3]</sup>。

在五子棋人机博弈系统的研究中, 人机对弈问题的关键是博弈树搜索算法和棋局局面评估策略, 搜索和评估算法的性能对游戏系统的博弈水平有着重要影响<sup>[4-5]</sup>。在搜索算法方面, 常用的随机式搜索算法存在着适用性差、最优解搜索效率低的问题, 而剪枝式搜索算法虽然降低了数据冗余度, 但存在剪枝判断算法需要依靠设计者经验的问题<sup>[6]</sup>。在局面评估策略方面, 当前的研究通过引入神经网络来提高评估效果,

但是存在训练时间复杂度高、样本质量敏感的问题<sup>[7]</sup>。因此, 研究基于博弈树搜索模型的五子棋游戏软件的实现具有重要的现实意义。

本文针对五子棋游戏系统设计中常用搜索算法和局面评估算法的局限性, 研究基于权值平衡策略的博弈树搜索与评估理论模型。通过详细讨论五子棋游戏系统的设计策略, 给出了系统的具体实现过程, 开发了游戏界面简洁、用户体验感好、自主学习能力优的五子棋人机博弈游戏系统。

### 1 基于权值平衡的博弈树模型研究

博弈树算法是人机对弈研究中最常用的一种算法, 通过模仿人类对弈时人脑的思维模式判断当前选择的对弈策略对当下棋局是否有利, 并进一步评估当前对弈策略对接下来棋局的影响。当经过若干对局之后, 就可以形成多个棋局局面的状态集合。对弈的过程也是局面状态不断切换的过程, 其局面状态转换方程为:

$$T_0 = T(0) \quad (1)$$

**作者简介:** 曹晓燕(1986—), 女, 江苏南通人, 硕士研究生, 讲师。研究方向: 智能优化、移动应用开发技术研究。

(C)1994-2021 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

$$T_{n+1}=T_n * P_{n+1} \quad (2)$$

式中,  $T_0$  表示五子棋棋局局面的初始状态,  $T_{n+1}$  表示对弈完第  $n+1$  步之后的棋局局面状态,  $P_{n+1}$  表示第  $n+1$  步的下子策略。

在对弈过程中, 每步棋后都可能会产生若干种棋局局面, 各局面状态按照自上而下的原理可以组合为树状。其中, 树的根节点表示棋局局面的初始状态, 子树层则表示参与对弈的两方按照走棋规则分别对应的状态层。若根节点为第一层, 则甲方的局面状态集合为偶数层的所有结点, 乙方的局面状态集合则为奇数层的所有结点。

以该树型结构为基础, 五子棋人机对弈的过程可以描述为甲方做出走棋决定(对应 A 结点)后, 乙方则据此生成两种及以上的可能(对应 B 结点和 C 结点), 在此基础上甲方则会产生更多的可能(对应 D 结点、E 结点、F 结点以及 G 结点)。

为了方便局面的搜索与评估, 考虑为每个结点增加权重值, 通过权重值的动态变化判读局面的优劣。假设结点 D 和结点 F 对应的策略是弱势策略, 结点 E 和结点 G 对应的策略是优势策略, 在对弈过程中乙方选择了 B 结点对应的走棋策略。由于在设定中结点 D 对应的弱势策略, 则选择 D 结点时该当前棋局局面的综合权重值应该减少, 该权重值为乙方的第一步决策提供依据, 经判断后应该选择 C 结点为最优结点。同理, 在进行下一次选择时权值会根据所选结点的优劣动态调整均值。通过多次去搜索以及权重值的调整, 优势结点 E 和结点 G 所在路径的权重值将会大于弱势结点 D 和结点 F 所在路径的权重值, 并以此引导选择向优势结点倾斜, 并最终确定走棋的最优策略。

## 2 五子棋游戏的设计与实现

### 2.1 五子棋游戏的总体设计

以研究的权值平衡博弈树搜索与评估算法模型为基础, 设计实现了人机对弈五子棋游戏系统。该系统由棋局局面表示模块、参数设置模块、监听器模块、局面搜索模块、局面评估模块以及胜负判定模块组成。其中, 设计开发的五子棋游戏系统最优走法求解的流程如图 1 所示。

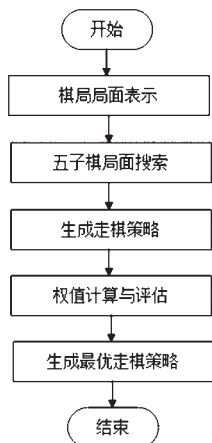


图 1 五子棋游戏最优走法求解流程示意图

### 2.2 五子棋局面表示的设计与实现

五子棋棋盘上的信息需要以一定的结构将其表示出来, 根据棋盘的物理结构考虑使用  $15 \times 15$  的横纵数列表示游戏走棋的位置信息, 从上到下和从左到右分别使用 15 根线条表示, 横纵线条交叉处为走棋位置, 该结构使用二维数据来实现。具体来说, 当二维数组的数值设置为 0 时, 则表示在该位置所走棋子为黑子; 当二维数组的数值设置为 1 时, 则表示在该位置所走棋子为白子。通过这样的设计, 可以方便地通过二维数组的下标值获取棋子位置信息。五子棋棋子绘制的核心代码如下:

```

#define len 15
if (chess[j][i] == 1)
    printf("●");
else if (chess[j][i] == 2)
    printf("○");
if (j == 0 && i == len - 1)
    printf("┐");
else if (j == len - 1 && i == 0)
    printf("└");
else if (j == len - 1 && i == len - 1)
    printf("┘");

```

### 2.3 权值平衡策略的设计与实现

根据本文所研究的权值均衡博弈算法模型, 在实际实现过程中需要将树结构中输掉比赛的走棋路径的综合权重值自减, 而将赢得比赛的走棋路径的综合权值自加, 单一棋局局面的权值则根据遍历访问次数计算。当黑子胜利时, 每个平衡权值都会增加它相应的调用次数, 以此来增加自己调用的权值比重。同样当白子胜利时, 平衡权值会减去它所调用的来降低这一步骤的权值比重, 以此表明这一步可能影响胜局的比重没有那么大。其中, 棋局局面胜负判断的关键代码如下所示:

```

int win(int x, int y, int word)
{
    int row, col, count;
    row = x, col = y; count = 0;
    while (chess[row][col] == word && col < len)
    {
        count++;
        col++;
    }
    col = y - 1;
    while (chess[row][col] == word && col >= 0)
    {
        count++;
        col--;
    }
}

```

```

}
if(count >= 5)
return 1;

```

## 2.4 局面评估策略的设计与实现

在进行局面评估时,需要新定义 4 个新的函数用来记录横向、纵向、正斜方向以及反斜方向相连的五子棋棋子的数量,并使用胜负判断方法来判定当前棋局局面胜负。在具体实现过程中引入新的变量用于计算分数,当分数超过设定的最大分数值时直接判定胜负,当任意两个方向都有四个子相连,并且没有被完全堵住,即气还有 1 以上时也可以直接判定为胜利。判断的关键核心代码(部分)如下:

```

if ((f1 == 4 && l >= 1 && r >= 1) || (f2 == 4 && u >= 1
&& d >= 1) || (f3 == 4 && ru >= 1 && ld >= 1) || (f4 == 4 &&
lu >= 1 && rd >= 1))
{ return sumscore = MAX; }

```

其中, l、r、u 以及 d 表示 4 个方向气的数量。

## 2.5 五子棋游戏系统测试与分析

为了验证本文所设计实现的五子棋游戏系统的有效性,分别以极大极小值搜索算法、剪枝算法以及本文所研究的权值平衡算法进行了 100 局对战。其中:极大极小值搜索算法的胜率为 26%、负率为 36%、超时无结果为 38%;剪枝算法的胜率为 75%、负率为 14%、超时无结果为 11%;权值平衡算法的胜率为 73% (后 20 场的胜率大道 91%)、负率为 27%,无超时情况发生。

综合以上结果,3 种算法中极大极小值搜索算法的时间复杂度较高,实际走棋过程中不能在有效时间内判别出结果的概率比较大,游戏体验性比较差;剪枝算法的胜率

较高,但时间效率仍然比较低;权值均衡算法胜率相对较高,且随着测试场次的增多胜率明显增强,该算法时间效率较高、游戏体验玩家的体验较好。

## 3 结 语

针对当前人机对弈五子棋游戏开发中博弈树搜索算法和棋局局面评估策略设计难度大、效率低的问题,研究了基于权值均衡策略的博弈树搜索与评估模型,优化了游戏的搜索与评估效率。同时在此基础上设计实现了人机对弈五子棋游戏系统,通过测试验证了所研究算法模型的有效性。未来将在现有研究工作的基础上进一步优化评估函数,不断提高游戏系统的棋力。

## 参考文献

- [1] 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,2020:11.
- [2] 王钦. 基于深度强化学习的五子棋算法研究 [D]. 重庆:重庆大学,2019:8.
- [3] 刘淑琴,刘淑英. 基于博弈树搜索算法的中国象棋游戏的设计与实现 [J]. 自动化与仪器仪表,2017(10):96-98.
- [4] 雷捷维. 基于强化学习与博弈树搜索的非完备信息博弈算法的研究与应用 [D]. 南昌:南昌大学,2020:6-7.
- [5] 孙若莹,官义山,赵刚. 一种新的博弈树迭代向前剪枝搜索 [J]. 沈阳工业大学学报,2017,39(3):304-310.
- [6] 李阳. 中国象棋博弈树搜索算法的研究与实现 [D]. 秦皇岛:燕山大学,2015:12.
- [7] 李学俊,王小龙,吴蕾,等. 六子棋中基于局部“路”扫描方式的博弈树生成算法 [J]. 智能系统学报,2015,10(2):267-272.