

基于 α - β 剪枝算法的五子棋游戏设计与实现

车志宏¹, 吕峰²

(1. 河北省财政厅一体化系统运维中心, 石家庄 050050; 2. 河北财华信息技术有限公司, 石家庄 050050)

摘要: 五子棋是一种两人对弈的策略类游戏, 极易上手, 深受人们喜爱。在此设计并研发了一种基于 α - β 剪枝智能搜索算法的人机对弈五子棋游戏, 在原游戏趣味性和娱乐性的基础上增加了智能性与挑战性。经测试, 该程序具有较高智能, 可以击败有一定基础的游戏玩家, 在休闲娱乐中能够帮助玩家提升棋艺, 具有一定的实用性。

关键词: α - β 剪枝; 搜索算法; 五子棋

1 概述

随着时代的发展, 人工智能已成为国家战略性新兴产业, 吸引了大量优秀人才投身人工智能领域的研究和应用^[1]。作为人工智能领域的重要分支, 人机博弈一直是人们进行科研探索的热门方向, 应用领域十分广泛。人机博弈算法常应用于对抗类的两人游戏中, 例如, 井字棋、象棋、围棋、五子棋等^[2]。传统的博弈树算法通过枚举所有的可能性并进行计算来选择最优解。对于最简单的井字棋(又叫“三子棋”, 棋盘为 3×3)来说, 它的第1步决策有9种, 对手的决策有 $9\times 8=72$ 种, 最后一步的决策有 $9!=362\,880$ 种。对于这个量级的计算, 普通计算机就可以完成, 但是对于五子棋 15×15 的棋盘来说, 显然穷举出所有的决策不太现实。

目前, 在人机博弈的五子棋游戏中, 多使用极小化极大值(Minimax)算法, 该算法是一种深度优先的搜索算法, 相比于传统的深度优先算法需要建立完整博弈树, 极小化极大值算法可以通过指定搜索深度并引入评估函数(Evaluation Function), 在搜索到指定深度时不再往下搜索, 而是对当前深度的各叶子节点使用评估函数评分进行决策。对于五子棋游戏而言, 如果第1步的决策有225种, 那么对方的对应决策就有 $225\times 224=50\,400$ 种, 再往下一层就会有 $225\times 224\times 223=11\,239\,200$ 种决策, 到了第3层时其节点数量的计算已经达到千万级, 因此选择极小化极大值算法并设定搜索层数为3时, 其计算量也是惊人的。一般来说, 五子棋高手能够看到自己落子后5~6步甚至更多步的棋局, 因此如果想进一步提高机器的博弈水平, 就必须增加搜索深度。 α - β 剪枝算法是在极小化极大值算法上改进的搜索策略, 在进行指定深度节点计算时, 不再是逐个计算所有的叶子节点, 而是减少非必要节点的搜索, 节省了算力, 在有限的计算力下能够搜索到更深层次的节点, 有效提高了算法性能和效率^[3]。

研究通过HTML+CSS+JavaScript语言实现了一个基于 α - β 剪枝算法的五子棋人机对弈游戏, 该程序能够根据当前棋局使用 α - β 剪枝算法自动搜索选择最优落子, 并根据游戏结果自动判定胜负, 实现了人机对弈。经测试, 研究实现的五子棋游戏能够战胜具有一定基础的玩家, 具有较高的智能性, 具有一定的挑战性。

2 算法原理

2.1 极小化极大值算法

极小化极大值算法是一种深度优先的搜索算法, 且满足零和博弈, 即双方利益总和为0, 在一方获得利益的同时, 另一方就会失去利益^[4]。因此, 在所有可选择的节点中己方都会选择令自己优势最大化或者对手优势最小化的选择。双方的博弈过程可以视作一颗决策树, 若当前层为己方的决策状态, 那么己方必然会选择收益最大的节点, 该层被称为MAX层; 若当前层为对方的决策状态, 那么在决策时己方必然会选择对方收益最小化的路径, 该层被称为MIN层。由此, 在构建的决策树中, 可以将树中的各节点区分为MAX节点(MAX层中非叶子节点)、MIN节点(MIN层中的非叶子节点)及算法中指定的第 N 层的叶子节点。通过评估函数对各叶子节点进行评分, 然后自底向上对各节点进行赋值, MAX节点取其子节点中最大的值作为当前节点的值, MIN节点取其子节点中最小的值作为当前节点的值。算法流程如下。

- (1) 自顶向下构建指定深度的决策树。
- (2) 使用评估函数计算各叶子节点的值。
- (3) 自底向上给各MAX节点和MIN节点赋值。
- (4) 从根节点选择其子节点中值最大的分支作为下

作者简介: 车志宏(1978—), 男, 学士, 研究方向为财政业务软件研发; 吕峰(1991—), 男, 硕士, 研究方向为人工智能、机器学习。



一步行动策略。

由此，可以看出极小化极大值算法需要搜索完所有的叶子节点，自底向上对所有的节点进行赋值，从根节点选择最合适的分支作为下一步行动策略。由于极小化极大值算法需要先构建出完整决策树后，再对各节点进行评分，所以在搜索深度增加时，其计算量会按照几何倍数增长，很难通过增加搜索深度提高性能。在其基础上改进的 α - β 剪枝算法很好地解决了这个问题，其不需要事先构建完整决策树，而是边计算边构建决策树，并通过剪枝的方式在保证准确性的同时减少子节点的扩展，因此在一定计算量的前提下， α - β 剪枝算法能够搜索更深层数，提高了准确性。

2.2 α - β 剪枝算法

α - β 剪枝算法是基于极小化极大值算法的改进算法。极小化极大值算法需要遍历所有的叶子节点后自底向上为各节点赋值。 α - β 剪枝算法的中心思想是减少博弈树扩展，减少内存使用，减少计算量，增加决策深度，从而提高决策的准确性。 α - β 剪枝算法只能按照深度优先的规则进行博弈树的创建。与极小化极大值算法不同的是， α - β 剪枝算法不需要遍历所有的叶子节点，而是在节点搜索中引入了 α 剪枝和 β 剪枝，在进行子节点搜索时受到其兄弟节点值的影响，从而判断是需要继续扩展还是进行剪枝处理（即不再进行搜索）。

首先，介绍 α 剪枝， α 剪枝示意图如图1（图1中圆形节点代表MAX节点，矩形节点代表MIN节点）所示。在对MAX节点A的子节点进行搜索时，假设已经确定B节点的值4，然后，继续搜索C节点，如果已经计算出C节点子节点D的值为7、E的值为3，且C节点为MIN节点，那么C的值一定是 ≤ 3 ，又因为A节点为MAX节点，在所有子节点中取最大值时一定不会选择C节点，所以在搜索完C节点子节点E后，对C节点其余子节点的搜索已没有意义，可以进行剪枝，这个过程就是 α 剪枝。

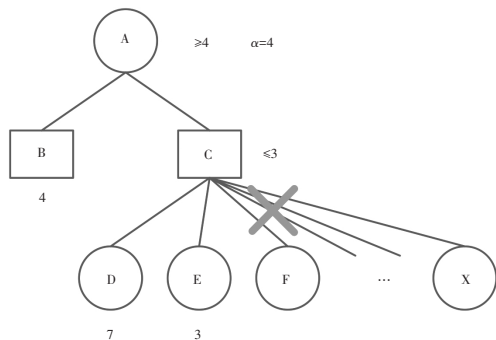


图1 α 剪枝示意图

在对MIN节点进行搜索时，同样有这样的情况发生， β 剪枝示意图如图2（图中圆形节点代表MAX节点；矩形节点代表MIN节点）所示。在对MIN节点A的子节点进行搜索时，假设已经确定子节点B的值为4，然后继续搜索C节点子节点，如果已经计算出C节点子节点D的值为2、E节点的值为5，且C节点为MAX节点，那么C的值一定是 ≥ 5 ，又因为A节点为MIN节点，在所有子节点中取最小值时一定不会选择C节点，所以在搜索完C节点子节点E后，对C节点的其余子节点的搜索已经没有意义，可以进行剪枝，这个过程就是 β 剪枝。

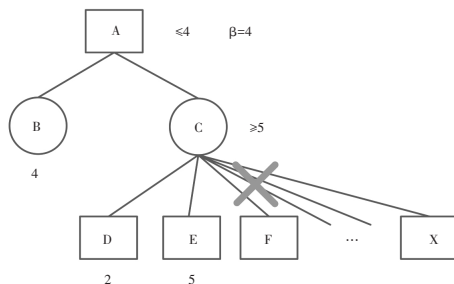


图2 β 剪枝示意图

通过以上过程可以看出，在创建决策树的过程中，使用 α 剪枝或者 β 剪枝，可以在保证算法准确性的同时，剪掉一些不必要的搜索过程，提高了搜索效率。算法流程可描述如下。

- (1) 开始构建决策树。
- (2) 递归进行深度优先搜索，如果当前节点是叶子节点，则使用评估函数计算其值。
- (3) 计算出当前叶子节点的值后，向上传递更新 α 、 β 的值，如果符合剪枝条件，则进行剪枝处理。
- (4) 从根节点中选择其子节点中值最大的分支作为下一步行动策略。

3 程序设计与实现

研究设计实现的人机对弈五子棋游戏提供了简洁美观的界面，在人机对弈过程中，计算机使用 α - β 剪枝算法自动计算最佳落子点，在计算机或玩家落子后能够自动根据当前棋盘状态判定胜负。根据功能需求将该程序划分为界面显示及落子显示模块、人机对弈模块和胜负判定模块的3个主要功能模块如图3所示。

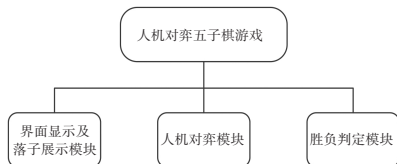


图3 主要功能模块

(1) 界面显示及落子展示模块。程序中使用HTML5的Canvas元素进行棋盘绘制,以及根据不同执子人的落子在对应位置绘制棋子样式,五子棋游戏界面如图4所示。画布背景设置为白色,棋盘线为黑色,双方棋子分别为黑色填充的黑子和白色填充带边框的白子,界面简洁美观。游戏中计算机执黑子,玩家执白子,单击开始游戏按钮后,计算机黑子先手在棋盘天元位置落子并绘制黑子样式,在玩家落子时根据玩家鼠标点击的位置自动计算坐标位置绘制白子样式,玩家落子后如果游戏没有结束,则由计算机自动计算落子点,并交替绘制棋子。每次落子后由胜负判定模块判定是否有五子连珠棋型,如果有,则游戏结束,落子者胜利。

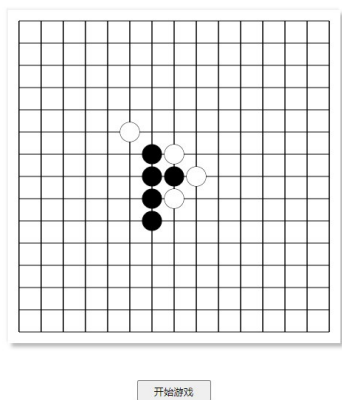


图4 五子棋游戏界面

(2) 人机对弈模块。主要负责评估函数的设计,以及根据当前棋盘布局,通过 α - β 剪枝算法处理计算最佳落子点。在评估函数中,针对当前棋子的分布设定了不同的得分值,计分规则如表1所示。表1中提到的部分五子棋棋型分布的专业术语在表2中有对应的图例介绍^[5],按照表1中的计分规则对各叶子节点的棋局进行评分。程序将计算机落子的节点记为MAX节点,将玩家落子的节点记为MIN节点,例如,在进行积分统计时,如果棋盘中计算机有活四棋型出现,那么计分为3 800,如果玩家有活四棋型出现,则计分为-3 800。由此,基于零和博弈原则,计算机在进行落子选择时,为保证评分较高会选择在高积分落子点落子的同时,阻止玩家形成高积分棋型。

表1 计分规则

棋型	分值
连五	99 999
活四、眠四	3 800
活三、眠三	680
活二、眠二	110
单子	20

表2 部分棋型图例

棋型	说明	图例示意图 (未全部列出)
连五	5颗同色棋子紧挨连成一线	或
活四	4颗同色棋子紧挨连成一线,且两端无对方棋子	
眠四	存在五连点,可形成连五但非活四	或或
活三	差一子可以形成活四	或或
眠三	差一子可以形成眠四	或或
活二	差一子可形成活三	或或
眠二	差一子可形成眠三	或或
单子	在一个方向的五个连续的位置只有本方一个棋子	

(3) 胜负判定模块。针对15×15的五子棋棋盘,共计有572种五子连珠的可能,程序已经将所有的五子连珠情况初始化为胜利棋型数组,每次落子后会当前棋型进行判定,如果和任一胜利棋型一致,则判定落子方胜利,游戏结束。

4 结语

研究设计实现一款基于 α - β 剪枝智能算法的人机对弈五子棋游戏,通过不断优化,在设定搜索深度为5时,计算机可以战胜有一定基础的玩家,且程序计算落子速度较快。将智能算法与五子棋相结合,使游戏更具有挑战性。目前,虽然研究实现的五子棋游戏能够战胜一般游戏玩家,但是仍可以进一步优化,例如,在评估函数中根据棋型进一步优化评分标准,加入针对“四三胜”“活三先手”等棋型的计分等。还可以根据设定搜索深度的不同设置关卡级别,例如,设定搜索深度为3时对手玩家比较友好,设定搜索深度为5时对手有一定基础的玩家比较友好等。这些也是下一步研究的(下转第153页)



4.2 模拟运动定位实验

在篮球运动中,运动员发起快攻有很多种进攻路线,其中包含了带球或无球从中线跑至三分线附近,呈 45° 向篮筐切入。此时如果是在带球的情况下,则运动员可以选择持球上篮;如果是无球跑位到这个位置,则继续等待队友传球,配合队友上篮攻框得分。运动员快攻实验定位结果如图2所示。

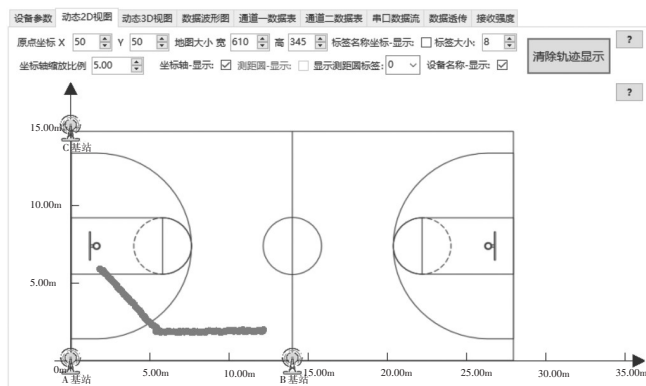


图2 运动员快攻实验定位结果

5 结语

室内运动定位通过UWB技术对运动员进行实时定位,充分利用UWB的高带宽、超强抗干扰能力、穿透力强等优势,保证了定位的精度和稳定性,符合对动态目标进行实时定位的要求。针对UWB测距算法进行推导分析,并提出了一种改进的双向测距算法。通过实验

对每个算法进行对比,由实验结果得知,双边双向测距算法可以将误差控制到2 cm内,满足了室内运动定位的设计要求,并且可以应用于多种室内运动场合,能够帮助教练员了解运动员跑位习惯及战术是否贯彻落实。

参考文献

- [1] 程霞. 无线通信技术的发展趋势阐述[J]. 中国新通信, 2022, 24(2): 8-10.
- [2] 魏青, 彭睿. 定位技术及室内定位方案探讨[J]. 广东通信技术, 2014, 34(5): 57-63.
- [3] 杜鑫, 朱文亮, 文西芹, 等. UWB超宽带室内定位技术[J]. 数字技术与应用, 2021, 39(12): 23-25.
- [4] 赵红梅, 赵杰磊. 超宽带室内定位算法综述[J]. 电信科学, 2018, 34(9): 130-142.
- [5] MACIEJ C. A prototype of static IR beacon-receiver positioning system based on triangulation method[J]. Measurement, 2018, 128: 149-159.
- [6] ZHANG X, XI X, SONG Z, et al. Performance analysis of spread spectrum through wall radar[J]. IEEE transactions on magnetics, 2014, 50(11): 1-4.
- [7] NEIRYNCK D, LUK E, MCLAUGHLIN M. An alternative double-sided two-way ranging method[C]//IEEE. 2016 13th Workshop on Positioning, Navigation and Communications. New York: IEEE, 2016.

(上接第133页)

方向。

参考文献

- [1] 李睿晶, 房超, 陈凯. 新时代我国人工智能发展回顾与展望[J]. 科技智囊, 2023(1): 14-21.
- [2] 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, 2021.

(上接第146页)

nition[J]. IEEE transactions on pattern analysis & machine intelligence, 2015, 37: 1904-1916.

- [10] 陈书贞, 任占广, 练秋生. 基于改进暗通道和导向滤波的单幅图像去雾算法[J]. 自动化学报, 2016, 42(3): 455-465.

- [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] 闫俊名. 双人零和博弈问题中策略搜索算法的研究[D]. 大连: 大连理工大学, 2021.
- [5] 郑健磊, 匡芳君. 基于极小极大值搜索和Alpha Beta剪枝算法的五子棋智能博弈算法研究与实现[J]. 温州大学学报(自然科学版), 2019, 40(3): 53-62.

- [11] WANG C, ZHU H, FAN W, et al. Single image rain removal using recurrent scale-guide networks[J]. Neurocomputing, 2022(7): 467.
- [12] 王林, 张鹤. Faster R-CNN模型在车辆检测中的应用[J]. 计算机应用, 2018, 38(3): 666-670.

