

一种中国象棋机器博弈剪枝策略的改进方法

蔡 灿

(北京工业大学 北京 100124)

摘要: 机器博弈一直被认为是人工智能领域最具有挑战性的课题,以中国象棋机器博弈中的应用为例,为了在有限时间内实现更深层次的博弈树搜索,本文研究了一种改进的剪枝策略。在传统 α - β 搜索算法的基础上,通过引入迭代深化和历史表等辅助剪枝手段,舍弃无价值的节点搜索,有效减少搜索节点数目,使剪枝更早发生以缩短搜索时间。实验表明这种改进策略的搜索算法的剪枝效率明显提高,在相同时间内搜索深度比原来的搜索算法更高,搜索的节点数也更多。

关键词: 中国象棋;机器博弈;剪枝策略;搜索算法

中图分类号: TP18 **文献标识码:** A **国家标准学科分类代码:** 502.20

Improved pruning strategy of Chinese chess machine game

Cai Shen

(Beijing University of Technology, Beijing 100124, China)

Abstract: Machine game has always been considered to be the most challenging task in the field of artificial intelligence, taking the Chinese chess machine game as an example, in order to achieve a deeper level of game tree search in a limited time, an improved pruning strategy is proposed. Based on the traditional α - β search algorithm, by introducing iterative deepening, history table and other auxiliary means of pruning, discard worthless node search, effectively reduce the number of nodes, make the pruning occurs earlier to shorten the search time. Experimental results show that the improved strategy search algorithm significantly improve the efficiency of pruning, at the same time, the search depth is higher than the original search algorithm, the number of nodes is also increased.

Keywords: Chinese chess; machine game; pruning strategy; search algorithm

1 引言

机器博弈一直被认为是人工智能领域最具有挑战性的课题^[1]。基于算法的博弈从某种意义上说可以理解作为一种树形结构,以及对树展开遍历查询。香农被认为是软件博弈理论的开拓者,他创造性地提出了两类博弈树查询方法:1)对博弈树所有节点进行遍历查询;2)经过选择,对博弈树一部分节点进行查询。在1950年香农的软件博弈理论发表以来,计算机运算能力得到了指数级提升,利用博弈理论对国际象棋进行软件模拟的研究越来越多。计算机棋类博弈能力也越来越高。20世纪50年代末,第一个完整的采用上述方法2)选择查询的象棋程序被开发出来。1997年,美国IBM公司推出了一款基于并行计算,利

用博弈树理论进行人机对弈的国际象棋博弈程序——“Deep Blue”^[2];同年5月11日,“Deep Blue”以3.5:2.5击败了世界排名第一的国际象棋选手Garry Kasparov,标志着机器博弈进入了一个新时代。而对中国象棋的计算机博弈的研究要远远落后于国际象棋,20世纪80年代首先在台湾开展并逐步发展开来,如中山大学的“纵马奔流”、东北大学的“棋天大圣”、上海计算机博弈研究所的“象眼”等^[3]。作为具有千年历史的棋类益智游戏,中国象棋在规则上与国际象棋差异较大,独成一体,其复杂程度远高于国际象棋,目前国内对中国象棋博弈算法的研究较少。本文通过梳理归纳国内基于中国象棋博弈研究成果,借鉴国际上关于棋类博弈算法研究的相关经验,对中国象棋传统 α - β 搜索算法^[4]进行分析的同时,探讨博弈剪枝策

收稿日期:2015-12

略能否在搜索深度、节点选择上提高效率。通过推演设计出一个完整的用于中国象棋博弈的软件实现理论。该软件具有一定的自主出着能力,可部署在PC机上,并可通程序对战实验,验证本文剪枝策略能够有效提高博弈算法效率。

2 剪枝策略

中国象棋采用回合制,是一种零和博弈游戏。当使用棋类搜索算法时,要同时考虑己方、对方下一步可能采取的着法,因此棋类对弈在博弈系统软件模拟可被抽象成包含从第一着到最后一着所有可能的节点树结构。博弈树片段如图1所示。

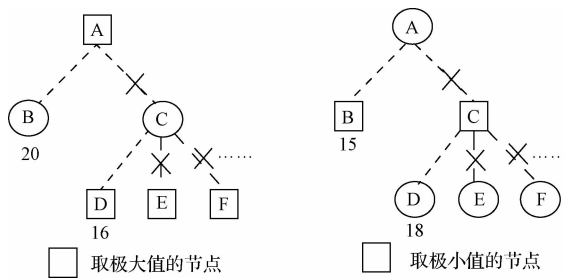


图1 α - β 剪枝示意

默认上层节点的取值取决于下层节点中的最大值。如图1(a)所示,A节点为博弈树的根节点,其一个分支节点B的值设定为20,则根节点A取值不能低于20,在节点B之下没有其他子节点。此时再看A节点另一个分支节点C,C节点之下存在子节点,那么把C节点作为一个单独的树来分析,假设C节点的一个极大值子节点D对应数值为16,则C节点的值不会超过16,小于B节点。对于A节点来说,B节点对其取值影响的权重明显高于C节点,因此在考虑A节点时,只考虑B节点的取值,对于C节点及其子节点都不再进行搜索。这种舍弃子节点搜索的方法被称为 α 剪枝,可以用式(1)表示:

$$\text{Value}(A) = \text{Max}(20, \text{Min}(16, E, F)) \quad (1)$$

如图1(b)所示,情况与图1(a)恰好相反,但原理一致,因此右侧剪枝过程的表达式如式(2),被称作 β 剪枝。

$$\text{Value}(A) = \text{Min}(15, \text{Max}(18, E, F)) \quad (2)$$

将 α 剪枝和 β 剪枝与极大极小算法的思想结合起来,就得到了 α - β 搜索算法。

3 剪枝策略的改进

α - β 剪枝算法是博弈树搜索的基础技术,得到了广泛的应用^[5-6],但 α - β 剪枝也有其固有的局限性,遍历搜索所有节点是得到博弈树最佳节点路径的保险方法,同时在广度不变情况下,搜索深度与路径优化程度正相关。由此可见,着法优化程度越高,花费时间也必然越高。在任一局面,中国象棋大约有40种走法,如果此时双方推演接下来的着数,则推演几步,生成节点数S为以40为底数,双方

推演步数的和为指数,即:

$$\text{Sum}(S) = 40^{\text{双方推演步数和}} \quad (3)$$

上述搜索是实验室理想情况,根据下棋人经验、规则的不同,实际上在不同局面某些着法不太可能采用的,如开局时“九宫”内的棋子基本上不会移动,对应博弈树上的节点在前期局面中也可以舍弃。

3.1 历史表

研究表明,在最理想的情形下建立的博弈树的节点个数为 w^d ,其中 w 是博弈树的分枝因子, d 是最大搜索深度^[7]。这大约是极小极大搜索法建立的节点数的平方根的二倍^[8]。可见如何调整待展开的着法序列,是提高搜索效率的关键^[9]。由于 α - β 剪枝算法与节点的排列顺序是强相关的,历史表可通过之前历史着法对当前所有节点,按着法从好到坏进行排序,节省搜索时间。历史表默认搜索树中某个节点上的一个好的着法,对于其他节点可能也是好的,好的着法即为可以引发剪枝的走法或者是其兄弟节点中最好的走法^[10]。因此历史表法会建立一张存储表,用以存储之前查询到的较好着法,并对这些着法按好坏权重进行排序。在某一局面,通过从好到坏搜索历史表,发现之前所存储的某一着法适用于当前局面,则认为这一着法是可实施的。通过这种对比搜索,可以节省大量节点查询的数量和时间,提高博弈算法效率。

具体方法是对好的着法进行打分,打分基于两项原则:1)认为搜索深度较高的着法其分值比浅层深度的着法好;2)认为博弈树搜索深度越大,位于搜索深度枝干部位的着法差异度越大,而对应的根节点着法被多次搜索的可能性也越大。基于这两点,历史表可采用矩形数组存储模式,每一个存储点同时存储本局位置和下一着预计位置,而数组先后顺序则通过给着法打分来确定。假设数组每个存储点初始值均为0,而该点每被作为好着法搜索一次,则给其历史得分增加 $2^{d \cdot \text{path}}$ 。通过叠加,将分值最高的着法放在第一,次高的放在第二,以此类推,则一局象棋中,所有的着法都得到了排序。

3.2 迭代加深

迭代加深是一个不断求精的过程,每一步搜索的时间是指定的,在固定时间内,先搜索一层,遍历同一层所有节点,完成后进入下一层搜索,目的是在固定的时间内尽可能多地增加搜索深度。在象棋博弈当中,局面的不同阶段,对应着法的可能性数量差别很大,对应的不同阶段做同一深度搜索所花费的时间区别也很大。一盘棋局中,同样的时间,局面后期可以搜索8~9层的博弈树分支,如果棋子较少的残局甚至可以搜索到几十层,而棋局中期仅能搜索到5~6层。如果从时间角度考虑,人类在处理相同局面时同样存在中前期每步耗时较长,后期耗时较短的情况。综上所述,迭代加深的实现关键在于时间控制和迭代过程中的有用信息的保存。程序实现中,需要在一开始建立计时函数,设定每次迭代的时间,迭代开始前清空历史表内原有信息,迭代开始后计时函数作为计时器开始计

时。如果一次迭代完成后计时器并未到时,则继续进行下一次迭代,如果此时计时器已超过设定的迭代时间,则停止迭代并返回结果。

4 实验结果及分析

将历史表、迭代加深与基本 $\alpha\beta$ 算法结合,并与基本的 $\alpha\beta$ 算法作比较,对它们的性能进行测试。在同样的条件下,当搜索深度分别从 1~5 时,得到这 3 种方法的搜索节点数和所用的搜索时间分别如表 1 和表 2 所示。用于实验的机器的配置为 Intel(R)Core(TM) i5-4590CPU @3.30 GHz,4 GB 内存。

表 1 搜索节点数比较

搜索深度	1	2	3	4	5
$\alpha\beta$	35	284	8 387	108 563	1 500 761
$\alpha\beta+$ 历史表	33	155	2 645	12 547	217 936
$\alpha\beta+$ 迭代加深	34	265	8 350	103 508	1 398 513

表 2 搜索时间比较 /ms

搜索深度	1	2	3	4	5
$\alpha\beta$	1	8	37	238	10 320
$\alpha\beta+$ 历史表	1	7	18	184	1 118
$\alpha\beta+$ 迭代加深	1	8	24	208	6 387

从表 1 和表 2 中可以看出,当进行浅层搜索时,迭代加深算法在搜索节点数和搜索时常上与基本 $\alpha\beta$ 算法基本相同,随着深度增加,迭代加深算法相较于其他算法有一定的综合优势,但其剪枝效果从数字上看并不突出。此时可以假设,迭代加深算法的剪枝效果是优于其他算法的,只是因为相同时间内搜索的节点更多,导致从总节点数目上与基本 $\alpha\beta$ 算法接近。在随后的程序对战实验中,推测得到了证实。实验共测试 10 局,每局每 5 s 完成一着,迭代加深的算法赢得了 7 场胜利,3 场平局,基本 $\alpha\beta$ 算法未赢一局。这充分说明了迭代加深算法搜索深度远高于基本 $\alpha\beta$ 算法,剪枝效果也更强,综合比较棋力更强。

此外,也可以从图中看出 $\alpha\beta$ 剪枝算法与节点排列顺序之间的强相关,历史表能够有效减少 $\alpha\beta$ 算法的搜索时间,且随着层数增加减少量变大,如第二层搜索时间减少了 1 s,第 3 层缩短了一倍多,第 5 层历史表算法的搜索时

间已减为原来的大约 1/10,节点数也急剧减少,这也说明了 $\alpha\beta$ 算法对着法展开顺序的敏感性。

5 结论

为了在有限时间内实现更深层次的博弈树搜索,剪枝算法舍弃了大量无价值节点的搜索,同时针对 $\alpha\beta$ 算法的缺点,采用辅助剪枝手段迭代加深方法以及历史表的引入使剪枝更早的发生,减少了程序的反应时间。实验证明这种改进策略的搜索算法的剪枝效率得到了明显提高。

参考文献

- [1] 傅调平,张奥狄,马滨强. 机器博弈海战兵棋推演系统的设计实现[J]. 计算机仿真, 2015, 32(3):14-18.
- [2] 谭世民. 人工智能技术在中国象棋博弈中的应用与发展[J]. 电脑迷, 2014(15):37.
- [3] 邢胜. 基于数据库自学习的中国象棋研究[D]. 保定:河北大学, 2009.
- [4] 岳金朋,冯速. 中国象棋 Alpha-Beta 搜索算法的研究与改进[J]. 北京师范大学学报:自然科学版, 2009, 45(2):156-160.
- [5] 裴祥豪. 基于剪枝策略的中国象棋搜索引擎研究[D]. 保定:河北大学, 2009.
- [6] 周开军,周鲜成,申立智,等. 基于多变量属性分类的图像形态滤波方法研究[J]. 仪器仪表学报, 2015, 36(8):1735-1743.
- [7] 张聪品,刘春红,徐久成. 博弈树启发式搜索的 $\alpha\beta$ 剪枝技术研究[J]. 计算机工程与应用, 2008, 44(16): 54-55.
- [8] 王骄,王涛,罗艳红,等. 中国象棋计算机博弈系统评估函数的自适应遗传算法实现[J]. 东北大学学报:自然科学版, 2005, 26(10): 949-952.
- [9] 周开军,周鲜成,申立智,等. 基于多变量属性分类的图像形态滤波方法研究[J]. 仪器仪表学报, 2015, 36(8):1735-1743.
- [10] 张群. 反射棋游戏系统的设计与实现[D]. 大连:大连理工大学, 2009.

作者简介

蔡岫,1989 年出生,硕士研究生,主要研究方向为人工智能、软件工程。

E-mail: cs_undead@126.com