

基于置换表技术的中国象棋搜索算法研究

黎利辉

(广西民族师范学院 数学与计算机科学系,广西 崇左 532200)

【摘要】中国象棋的基础搜索——Alpha-Beta 算法的剪枝过程对搜索节点的排序顺序依赖很大,当搜索顺序的排列为最差情况时,该算法基本上不能实现剪枝。搜索过程其实会出现很多重复的节点,利用哈希表的思想,把以前搜索过的节点保存起来,这样在搜索某一节点之前,先到哈希表里去查找以前是否搜索过,如果以前搜索过,则直接返回哈希表里保存的局面评估值;如果没有,则采用正常的 Alpha-Beta 算法进行搜索。通过实现可知当搜索层次大于 5 层时,改进后的算法比 Alpha-Beta 算法在搜索节点数量和时间上都有很大的优化。

【关键词】中国象棋;置换表;哈希表;博弈树搜索

0 引言

在中国象棋的人机博弈的研究中,局面搜索算法是核心,中国象棋的博弈过程中,针对某一局面,平均着法达到 60 步,故减少搜索的节点,提高搜索的速度和节省搜索过程中内存开销,就成了中国象棋搜索研究的一个终极目标。Alpha-Beta 搜索算法是人机博弈的一个基本算法,但是该算法在搜索过程中,存在大量的冗余^[1]。本文即在这一基本算法的基础上,加入置换表技术和 Hash table 技术,以提高搜索速度。

1 Alpha-Beta 搜索算法

如图 1 所示,结点下面的数字为该节点所代表的局面评估值。在左图中,节点 B 的值为 18,节点 D 的值为 16,因为 C 节点要取其子节点的最小值,故可以判定节点 C 的值将小于或者等于 16,而 A 节点的值为 B 节点和 C 节点的最大值,这样无论 E 节点和 F 节点的值为多少,都不影响 A 节点的取值,因为 A 节点此时肯定会取 B 节点的值 18,这样将节点 D 的后继节点减去称 Alpha 剪枝。观察图 1 的右半部的极大极小树, A 节点要取 B 节点和 C 节点的最小值, C 节点要取其子结点的最大值,而 D 节点的值为 18,故 C 节点的值最小为 18,这个值已经大于 B 节点的值了,所以无论 E 节点和 F 节点的值为多少,都不影响 A 节点的取值,所以将节点 D 的后继节点减去称 Beta 剪枝。

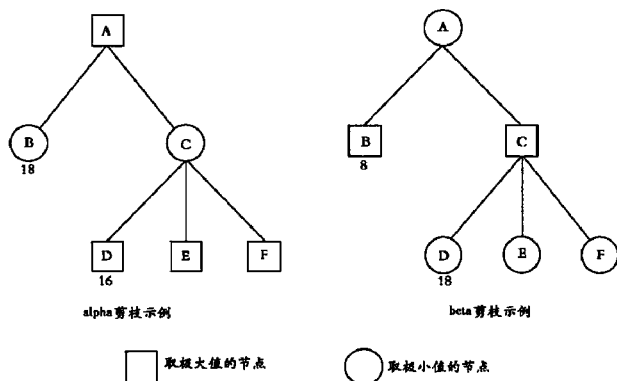


图 1 Alpha-Beta 剪枝过程

原始的 Alpha-Beta 搜索算法略显繁琐,需要在奇数层进行 Alpha 剪枝,在偶数层进行 Beta 剪枝,利用负极大值搜索的思想,可以统一在任何一层进行 Beta 剪枝。

2 置换表的思想

Alpha-Beta 搜索算法的剪枝过程对搜索节点的排序顺序依赖很大,当搜索顺序的排列为最差情况时,其搜索效果与极大极小值相同^[2]。

其实在搜索的过程中,随着搜索层次的加深,会出现一些节点是原来搜索过的,如果能直接利用已经搜索过的节点局面评估值,而不重新搜索一次,这样相当于进行了很大程度的剪枝^[3]。我们可以利用一个数组将已经搜索过的节点保存起来,在搜索新的节点时,先到数组中检查以前是否已经搜索过本节点,如果是,则直接用数组里保存的局面评估值,如果没有,则继续正常搜索,这就是置换表(Transposition Table)的思想。

3 哈希表来实现置换表搜索

因为在搜索的过程中,节点数量非常大,如果将每一个节点都与数组中的元素一一对应,内存开销太大,不可能实现。为了解决这一问题,可以利用哈希表技术。每一个搜索节点对应哈希表中一个节点,但是反过来,哈希表中的一个节点并不一定只对应一个搜索节点,这就是哈希冲突。在同一次搜索过程中,哈希冲突的可能性并不高,在实际的应用中这是一个可靠的办法。根据中国象棋搜索的特点,定义如下哈希数组。

```
struct HASHITEM {
    int64 checksum; // 用以验证表中数据是否是要找的局面
    int depth; // 搜索深度
    enum{exact, lower_bound, upper_bound} entry_type;
    double eval; // 所代表的节点的评估值
} hashtable[HASH_TABLE_SIZE];
```

对要搜索的每一节点,计算出它的一个哈希值(hashIndex,通常是一个 32 位数对哈希表大小取模)以确定此局面在哈希表中的位置。计算另一个哈希值 checksum,来检验表中的数据项是否是所要的那一项。64 位的哈希值 checksum 发生冲突的几率很小,几乎可以将其冲突忽略。即使因为哈希冲突导致没有获得原来搜索过的节点,也不会影响博弈系统的运行,因为还可以采用基础的 alpha-beta 算法进行搜索。

在对某一局面搜索之前,先查看哈希表项 hashtable[hashIndex],如果 hashtable[hashIndex].checksum == Checksum,并且 hashtable[hashIndex].depth 大于或者等于最大搜索深度减去当前层数,就返回 hashtable[hashIndex].eval 作为当前局面的估值。当对一个局面搜索完成后,将 Checksum、当前层数和估值结果保存到 hashtable[hashIndex]当中,以备后面的搜索使用。置换表与 alpha-beta 搜索协同工作的实现机制如下面的类 C 语言伪代码所示。

```
Intalphabeta(int depth, int alpha, int beta)
{
    int value, bestvalue = -INFINITY;
    inthashvalue;
    if(lookup(depth, &alpha, &beta, &hashvalue))
        return hashvalue;
    for(each possibly move m) // 对每一个可能的走法
    {
        make move m; // 产生子节点
```

(下转第 66 页)

※基金项目:广西教育厅科研项目(201106LX652)。

作者简介:黎利辉(1980-),男,湖南汨罗人,广西民族师范学院数学与计算机科学系教师,主要研究方向为无线传感器网络、图形图像和人工智能等。

从表 3 所示数据可以看出,随着固体润滑剂的加入,样品的润滑能力得到了提升。接着,我们还进行了样品加压情况下润滑能力的测试。具体试样过程是:

润滑脂抗压性能试验:按照一般的测试摩擦系数变化率的方式进行测试,但需在测试是在重物上方施加一个额外的垂直于地面的压力,压力 100N,加压 3~5 次,每次持续 5 分钟左右。期间须观测标尺读数变化。其结果如表 4:

表 4 润滑脂抗压性能试验

	摩擦系数变化率
样品 1	57
样品 2	62
样品 3	65

由以上试验结果可以看出,在润滑脂中适当加入一些添加剂可以比较有效的改善润滑脂的抗压性能。

3 配方试验

不同组分比例下的性能测试:

为了获知不同组分比例对样品性能的影响,按以下比例进行分组测试,并对其结果进行对比分析。具体分组如下:1.基础油 75%,稠化剂 25%,添加剂 5%;2.基础油 70%,稠化剂 20%,添加剂 10%;3.基础油 70%,稠化剂 15%,添加剂 15%(其中添加剂部分二硫化钼含量保持 2%)。分别对三组样本进行了摩擦系数减轻,高速转动残留测试。测试结果如下:

表 5 润滑脂配比测试-润滑性能

	摩擦系数变化(%)
样品 1	62
样品 2	64
样品 3	66

表 6 润滑脂配比测试-吸附能力

	残留重量
样品 1	0.4
样品 2	0.62
样品 3	0.67

由表中结果可以得知,随着基础油的减少和添加剂加量的增加,样品的吸附能力有所提高,而润滑性能则有随着稠化剂的减少而增加的趋势。

由以上试验结果,再结合各组分的市场价格进行分析,在保证润滑和吸附性的前提下,决定使用以下的比例最为产品的实际生产方案:基础油 70%,稠化剂 18%,添加剂 12%。

4 产品配制

在润滑脂的配制过程中,温度与时间始终是影响最终产品性能的关键性因素。温度过高会导致调配过程中组分中某些轻质物质挥发从而对产品外观与质量,温度过低则稠化剂难以形成真正的胶体状态从而使其在基础油中的分散。同时,对反应时间和冷却速率的把握也很重要。具体的配制过程如下:先在烧杯内加热基础油,并保持到一定温度,然后,加入稠化剂(此时的稠化剂选用的是成品有机酸钠),开始搅拌,速度开到 500 到 1000 转,搅拌一定时间,期间保持加热以使得稠化剂能顺利溶解并分散。然后加入添加剂,保持温度继续搅动。至烧杯内部目测无残留固体时,关闭加热装置,继续搅拌,至观察到开始凝固时停止搅拌,放入冷却装置中冷却,也可置于室内自然冷却。

5 结论

- 1)新型钻杆润滑脂的实根本实验室性能已基本达到现有的钻杆润滑脂的性能,并且在极压性,吸附性上有所改进;
- 2)本品所使用稠化剂与添加剂均无毒副作用,基础油使用的是废机油,因此,在保护环境方面也能有所贡献,符合目前润滑脂的发展趋势;
- 3)本品具有一定的市场空间,随着技术的进步,成本价格的进一步降低,其市场空间也将得到更大的扩展。

【参考文献】

[1]赵玉贞,宗明.润滑脂应用的基础知识[J].
[2]朱廷彬.润滑脂技术大全[M].
[3]刘伟,张天胜.利用废油脂制备高滴点锂基润滑脂的研究[J].润滑油,2005,6,20(3).
[4]杨礼河,孙玉德,范国琛.环境友好润滑油发展概况[C]/2008 中国润滑油技术经济论坛论文专辑,76-77.
[5]杨海宁,姚立丹.国内润滑脂产品及其应用技术的进步简述[J].
[6]孙庆民.金刚石钻进用润滑膏[J].地质与勘探,1983(07).
[7]于殿奎,李演.钻杆润滑脂在钻探生产中的应用[J].探矿工程,2005(10)47.
[8]刘晓春.黑机油复合钻杆润滑脂的研制与应用[J].新疆有色金属,2010(4):16-18.
[9]谢凤,季峰.固体添加剂在润滑脂中的应用研究[J].合成润滑材料,2008,35(4):12-16.
[10]谢凤,朱江.固体润滑剂概述[J].合成润滑材料,2007,34(1):31-32.

[责任编辑:杨玉洁]

(上接第 23 页)//递归搜索子节点

```
value = -alphabeta(d-1,-beta,-localalpha);  
unmake move m;  
if(value >= bestvalue)  
    bestvalue = value; //保存最佳值  
if(bestvalue >= beta)  
    break; // beta 剪枝  
if(bestvalue>alpha)  
    alpha = bestvalue; // 修改 alpha 范围  
}  
store(depth, bestvalue, alpha, beta);  
return bestvalue; // 返回极大值
```

4 实验

基于以上核心算法,利用 JAVA 语言设计了一个实用系统。中国象棋的棋盘用一个 10 行 9 列的二维数组与之对应,而各棋子用对应的数字表示。棋局评估主要从剩余棋子基本值、棋子灵活性、棋子受攻击度、棋子受保护度、棋子位置附加值这几个方面来衡量。分别计算这几种类型的值,然后将它们的和作为棋局的优劣值,供搜索引擎使用。实验主要是比较 alpha-beta+置换表算法与基本的 alpha-beta 算法在不同的搜索层次上搜索结点的个数,通过实验可知从第 3 层开始,

alpha-beta+置换表评估的节点数目要少于 alpha-beta 搜索在同样深度搜索中评估的节点数。而且随着搜索的最大深度的增加,置换表的命中率也不断提高,表明重复的节点所占的比例随深度增加而增加。这表明 alpha-beta+置换表评估的节点数同 alpha-beta 搜索评估的节点数相比的比例越来越低。由于置换表的操作对每一节点都要花费一定的时间,所以在较浅的搜索当中(例如 3 层),由于命中率较低,虽然 alpha-beta+置换表评估的节点数比 alpha-beta 搜索少,但花费的时间仍多于 alpha-beta 搜索。但随着搜索深度的增加,alpha-beta+置换表开始显露出时间上的优势。当搜索的最大深度为 5 时,alpha-beta+置换表的搜索速度达到 alpha-beta 的 2 倍以上。

【参考文献】

[1]王小春.PC 游戏编程[M].重庆:重庆大学出版社,2002.
[2]Knuth D E, Moore R W. An analysis of Alpha-Betapruning [J].Artificial Intelligence, 1975,6:293.
[3]徐心和,王骄.中国象棋计算机博弈关键技术分析[J].小型微型计算机系统,2006,27(6):961-968.

[责任编辑:周天凤]