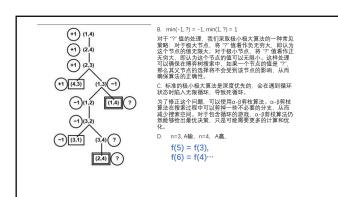
5.9 本题以井字棋(圈与十字游戏)为例练习博弈中的基本概念。定义 X,为恰好有 n 个 X 而没有O的行,列或者对角旋的数目。同样O,为正好有A O 的行,列或者对角旋的数目。效用函数给 X_1 = 1 的机局+1. 给 O $_2$ = 1 的机局+1. 所有其他终止状态效用值 为 O,对于非终止状态,使用线性的评估函数定义为 E $val(s) = 3X_1(s) + X_1(s) - (3O_2(s) + (3O$ $O_1(s))$. 估算可能的井字棋局数。 a. 台界可能的分子供のMX - 考虑対象性、 台出从空棋盘开始的深度为 2 的完整網穿椅(即,在棋盘上一个 X 一个 Ø 的棋局)。 c. 标出深度为 2 的棋局の評估高数据。 d. 使用极小极大算法标出深度为 1 和 0 的棋局的倒推值,并根据这些值选出最佳的起 e. 假设结点按对lpha-eta剪枝的最优顺序生成,圈出使用lpha-eta剪枝将被剪掉的深度为 2 的 a) 9! b) c) d) 中心位置 e)

5.8 考虑图 5.17 中描述的两人游戏。 A B 1 2 3 4 1 2 3 a. 根据如下约定画出完整博弈树: b. 给出每个结点倒推的极小极大值(也标记在圆圈里)。解释怎样处理"?"值和为 什么这么处理。 c. 解释标准的极小极大算法为什么在这棵博弈树中会失败,简要说明你将如何修正 它,在(b)的图上画出你的答案。你修正后的算法对于所有包含循环的游戏都能 给出最优决策吗? d. 这个 4-方格游戏可以推广到n个方格,其中n>2。证明如果n是偶数A一定能赢, 而 n 是奇数则 A 一定会输。



循环检测:在执行剪枝之前,进行循环检测以避免陷入无限循环。当在搜索过程 中遇到一个状态时,可以检查该状态是否已经在当前路径中出现过。如果检测到 循环状态的出现,可以中断当前路径的搜索,并返回一个适当的值,以避免进入 无限循环。

前、 前枝条件:在进行剪枝时,可以根据当前状态的信息和已经计算得到的值,判断 是否可以提前终止该分支的搜索。在处理循环状态时,可以根据循环检测的结果 来判断是否进行剪枝。

如果循环检测结果表明当前状态已经在当前路径中出现过,可以认为进一步搜索 该分支是无意义的,因为会导致重复计算和循环状态的进入。在这种情况下,可 以直接进行剪枝,即不再深入搜索该分支,并返回一个适当的值。

以直接进行男权,即个特殊人授索该方支,并返回一个。如当的值。如果循环检测结果表明当前状态未出现在当前路径中,说明该状态是首次遇到的,可以继续进行搜索并使用剪枝技术。在此过程中,可以根据已经计算得到的值和评估函数的结果。确定是否可以进行剪枝。 综上所述,剪枝技术可以通过结合循环检测和剪枝条件来处理循环状态。循环检测可以防止进入无限循环,而剪枝条件可以根据循环检测结果判断是否进行剪枝,从而减少搜索空间和重复计算。这样可以提高算法的效率,并避免在处理循环状态时陷入无限循环的困境。

 \triangle $n_2 = max(n_3, n_{31}, \dots, n_{3b_3})$ $n_1 = min(max(n_3, n_{31}, \dots, n_{3b_3}), n_{21}, \dots, n_{2b_2})$ Ď MAX $n_1 = \min(l_2, \max(l_3, n_3, r_3), r_2)$ 图 5.18 是否剪掉结点 n_j时的情形 $min(l_j, n_j, r_j)$ $min(l_2, l_4, \ldots, l_j)$ d. $max(l_3, l_5, \dots, l_k)$

5.13 请给出α-β剪枝正确性的形式化证明。要做到这一点需考虑图 5.18。问题为是否要剪 掉结点 n_j , 它是一个 MAX 结点,是 n_1 的一个后代。基本的思路是当且仅当 n_1 的极小极大值可以被证 明独立于n,的值时,会发生剪枝。 a. n_1 的值是所有后代结点的最小值: $n_1 = \min(n_2,$ $n_{21}, \, \cdots, n_{2\,b2}$)。请为 n_2 找到类似的表达式,以得到用 n_j 表示的 n_1 的表达式。 b. 深度为i的结点 n_i 的极小极大值已知, l_i 是在结 点 n, 左侧结点的极小值 (或者极大值)。同样, r_i 是在 n_i 右侧的未探索过的结点的极小值(或者 极大值)。用 I_i 和 r_i 的值重写 n_1 的表达式。 c. 现在重新形式化表达式,来说明为了向 n₁ 施加 图 5.18 是否剪掉结点 n_i时的情形 影响, n_i 不能超出由 l_i 值得到的某特定界限。 d. 假设 n_j 是 MIN 结点的情况,请重复上面的过程。

