2048AI Report

Fangzhou Luo & Songtao He

1. 算法框架

（1）算法的基本框架为：求最大期望

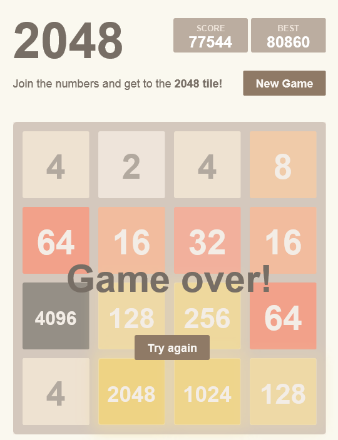
（2）通过递归的方式枚举n步移动后的场面，通过一个启发函数对各个场面进行评估。选择期望评估结果最大的那一个方向作为当前移动方向。当然，在每一层（步）的计算会分成两个部分，首先是计算四个方向移动后的期望值，然后选择最大的传递给上一层。既先平均再取最大值。

（3）为了提高速度，采用四线程并行的方式，同时对四个方向进行评估。虽然这种方法很难做到负载平衡，但是还是可以起到一定的速度提升的。

（4）对于层次数控制，大体分成两个阶段：

（A）当最大值小与4096时，固定迭代深度为3层或4层。这个数值基本上可以保证每一次都能很快的达到4096。

（B）当最大值大于等于4096时，采用迭代加深算法，既初始迭代深度为3或4，统计总的局面数，逐渐增加迭代深度，直到达到某一个阈值。

1. 启发函数
2. 棋盘中空格的数量，乘以系数作为加分。
3. 每一行每一列的最大值是否在两侧，符合要求的个数，乘以系数作为加分。
4. 相邻两个位置的数字相同的对数，乘以系数作为加分。
5. 每一行每一列是否有序，有序的个数，乘以系数作为加分。
6. 将棋盘划分成蛇形，按顺序填入，将此棋盘与原棋盘比较，将差的绝对值求和，再乘以一定的阈值作为减分。
7. 按照蛇形走法，从最大值开始，连续的递减序列的长度，乘以系数作为加分。
8. 在（6）中，一些关键位置进行额外加分。
9. 所有格点距离最大值的曼哈顿距离和格点值的大小的关系，大概就是如果排序后，比较大的数字距离最大的数字越近，得分越高。
10. 以上所有启发函数中的得分都会根据所涉及的格子的数值进行加权。
11. 如果迭代过程中游戏结束，得分全部为0 。
12. 最终的启发函数又（1）至（10）混合而成。
13. 实验结果

可以很快的稳定的达到4096。有一定的概率达到8192 。