

## 五子棋算法报告

思路整理:这个五子棋程序最开始我是没有什么想法的,毕竟跟以前学的东西还是有着很大差别的。因为以前的,无论是计算器还是什么 c++ 程序,都是一种确定的结果,但是这个五子棋却要控制胜率,需要随着自己下的棋走不同的地方,是一种未知的程序,勉强可以算是一种智能了,虽然比较简单。我想到的第一种方法就是直接设定程序,首先保证不会输,就是可以一直堵着玩家的棋路,这个还是很容易设定的。在此条件下再想办法将电脑的棋路连起来。但是这样编程难度就稍微增大了一些。经过调试,我发现我的程序虽然可以一定程度上连起来,但是也常常会出现一些很容易让玩家发现的套路,所以最后的结果常常就是维持一个不胜不败的结局。第二种方法就是在网上学习到的,计算每一个位置的优劣。优劣则是用这个位置可以实现某种胜利方法的可能来判断的。不过这种方法的劣势就是需要计算的步骤太多了,因此算法的优劣就很重要,而且我还没有弄明白如何让计算机判断获得胜利的方法的好坏。

遇到的问题:首先就是我的最初的想法,只能维持平局,获胜几乎不可能,但是也不会赢,因此此方法可以进行练手,但却无法最终成功。然后第二种方法则是无法让计算机进行棋谱的分析判断,也就是最重要的,让计算机拥有“智能”的那部分。还有一个问题,也是以后需要解决的,就是胜率控制的问题。这也是计算机智能需要解决的一个重要问题。难道是让计算机在某些情况下故意走一些错的步骤吗?我觉得还是有些问题的。希望可以在网上或者书中找到一些解决方法。

胜率数据:正如上面所说的,我的第一种设想失败了,因此一直控制在一种不胜不败的状态...希望正在进行的第二种方法可以带来一些改变。

反思展望:五子棋的规则是五子相连即可胜利。可以解析为若四字相连而且两边没有对方棋子即可必胜,可以衍生出十字法,构造三角法,先堵后赢法等等,都是力求用最少的步数连成最多的连子,这样可以更快构造出四子相连的局面。我觉得这也可以用算法进行考虑。此外,我个人觉得,是不是输入的套路越少,五子棋的难度就越低?这是不是就是控制五子棋难度的方法?五子棋最看重的就是各步的优劣,也就是说,可以实现电脑必胜的局面,至少也可以维持不败的情况。另外,玩家与电脑依次行动,步骤错误判断也应该考虑。若是五子棋想要多功能一些,则可以设置悔棋,提示等等选项,让电脑自己对战自己。另外,整体的步骤就是设置一个模块,构建一个游戏平台,其他的就是程序需要做的事情了。当然,无论如何,算法都是最重要的部分。