Nomoku五子棋游戏

——程序工作原理与设计说明

林理露 -1120141826-07111403-15624951826

刘洪毅 -1120141827-07111403-18800176313

朴泉宇 -1120141831-07111403-15624951831

谷飒 -1120141946-07111403-15624951946

1、引言

人工智能（Artificial Intelligence），英文缩写为AI。它是一门关于理解人类智能内在机制，并在机器上予以实现的新的技术科学。人工智能是目前人类科学大厦中最为活跃、最富有魅力的组成部分之一。在科技日益发达，信息量迅速膨胀，同时人们想象力日渐丰富、实现力不断增强的今天，人类对智能机器的需求更为迫切，人工智能技术的发展对社会的进步具有重要的意义。

本课程从人工智能的各个方面展开讲解，主要分为机器学习、符号主义、联结主义、行为智能、进化计算与群智能等方面，全方位地向我们介绍了人工智能的方法与原理。这次大作业是这门课程的配套实验，旨在要求学生参与到实际项目的开发工作中来，加深对人工智能基本方法和基本原理的理解。

2、人员分工

通过对大作业要求的研究，以及对我们所学知识掌握程度的评估，确定小组成员分工如下：

林理露： alpha-beta 算法——根据棋局适应度及MIN-MAX树进行快速搜索决策下一步位置。

刘洪毅：人工神经网络——用来评价当前棋局的适应度

朴泉宇：遗传算法（GA）——用来搜索最优的神经网络

谷飒： 文件整理及文档编辑

3、程序设计说明

Nomoku五子棋游戏主要利用人工神经网络得出评价值，来进行alpha-beta剪枝（加入min-max），搜索下一步要走的棋，循环进行alpha-beta剪枝，最终得到比赛结果。

人工神经网络输入层为棋盘大小，中间层为1/2个棋盘大小，输出一个评价值（当前状态下的棋局适应度）。

最终得到的比赛结果进行强化学习。这里，我们采用遗传算法进行学习。染色体为一个人工神经网络的所有权值序列，一个染色体的适应度为其对应的神经网络经过多次比赛而得到的胜率。

4、程序工作原理

4.1、AI框架

本小组所选题目为基于五子棋游戏，应用神经网络，alpha-beta剪枝，MIN\_MAX决策树，GA遗传算法，强化学习，建立人工智能。意于通过实践，加深学习人工智能基础知识，实践掌握基本人工智能是想方法，以下为本小组具体程序工作原理与设计说明。

本小组意于学人工智能，所以五子棋的UI及其他与人工智能无关部分由外部程序piskvork.exe提供，该程序亦可使本小组程序在网上对战，人机对战，双AI对战等。网上对战应基于Gomocup，本小组的外部程序亦是由Gomocup提供。Gomocup是人工智能玩五子棋的比赛。(piskvork.exe下载网页<http://gomocup.org/download-gomocup-manager/)>

注意生成的exe不能直接运行，需要从piskvork设置中选择棋手处选择生成的exe文件，然后再开始对局即可和AI对弈，同时也可以使其与gomocup网站上的其他AI进行对弈。

本小组将使用神经网络作为评价函数，评价一个棋盘状态的适应度；使用alpha-beta剪枝来快速做出下一步的决定；使用机器学习中强化学习的方式来学习神经网络，并应用GA遗传算法细化强化学习方法。由于三者间存在着同步问题，故本程序采用了多线程的解决方案，将GA算法运行在另一个线程中，通过操作系统中的PV信号量同步方法，来进行线程间的消息同步。

4.2、各部分设计

4.2.1、神经网络

4.2.1.1、结构

本小组神经网络使用三层神经网络，分别为输入层，中间层，输出层。

输入层输入棋盘状态，所以有width\*height个神经元;

中间层本小组设定为(1.0 / 2)\*width\*height个神经元；

输出层输出当前棋局的适应度，1个神经元；

各层间全连接，所有连接上有一个权值与其对应；

整合函数为加权求和函数；

激活函数为线性函数y=x；

4.2.1.2、功能

包含初始化神经网络，根据棋局状态计算其适应度，根据自身神经网络计算自身胜率（模拟下棋，下N盘，根据胜场计算胜率）作为GA搜索中整个神经网络的适应度，为GA从外部文件导入已搜到的神经网络最优值，为GA导出已搜到神经网络最优值到外部文件。

4.2.2、遗传算法GA

4.2.2.1、算法思路

① 将现在的神经网络（包括第一次随机生成、以及对一定大小棋盘学习过了的神经网络）的权值序列（权值序列为）作为个体，并默认为最好情况。

② 首先随机产生32个个体作为种群，计算他们的适应度。此处的适应度为，单拿出这个神经网络权值序列带入到神经网络进行10局棋局，然后调用上文的胜率统计函数统计出的胜率。

③ 根据统计出的胜率（适应度），用轮盘赌的方式选出32个最好的子代作为父代。计算完成之后，将适应度最高的个体与当前最好的个体比较，若适应度更高，则替代当前最好个体，并调用上文的最佳神经网络记录函数，在文件中输出并记录。

④ 将父代按照0.9的重组概率，顺次地两两重组。重组的位置随机产生。

⑤ 将重组后得到的32个个体，按照0.05的概率进行单点变异。变异位置随机产生。

⑥ 最终得到的32个个体，在迭代次数范围里，返回第②步。

备注：以上的循环过程，每个子代都要进行10局棋局，故一次迭代就要进行(1 + 32)\*10 = 330局棋局。这是训练的过程。在进行一定的训练之后，应用对应棋盘大小的已经训练完成的神经网络，即可进行展示。

4.2.2.2、GA算法功能

本算法为典型的GA算法，运用GA的种群、重组、变异、迭代的特性，以胜率为适应度，能够较快速地进行收敛的同时，多次、多段的随机保证了种群多样性，得到较优的神经网络。

4.2.3、Alpha-Beta剪枝

4.2.3.1 算法思路

若棋盘为空，则向棋盘中心落子

若棋盘中已经有棋子，则开始alpha-beta剪枝

2步思考

由于五子棋的特殊性，棋盘中的将要落下的棋子有极大概率将会存在于已经存在的棋子的邻域内，为了提高搜索效率，同时降低搜索复杂度，本AI采用B\_TO\_EXPAND标记，来进行邻域扩展，在第一步Alpha搜索时，将本次邻域扩展的结果保存至v\_expand\_a的vetor中，即此得到了我方可能下的可能的落子处。

接下来进行第二步思考，即考虑所有对手可能的落子处，即进行Beta搜索，使用同样的规则和方法进行邻域扩展，将本次邻域扩展结果保存至v\_expand\_b中

评价

在得到所有对手可能的落子处后，接下来就是对局面的评价了，此时调用人工神经网络提供的评价函数，传入此时棋局的临时状态(假设我方在a处落子,对方在b处落子的情况下)，返回值越大，代表对我方越有利。

剪枝

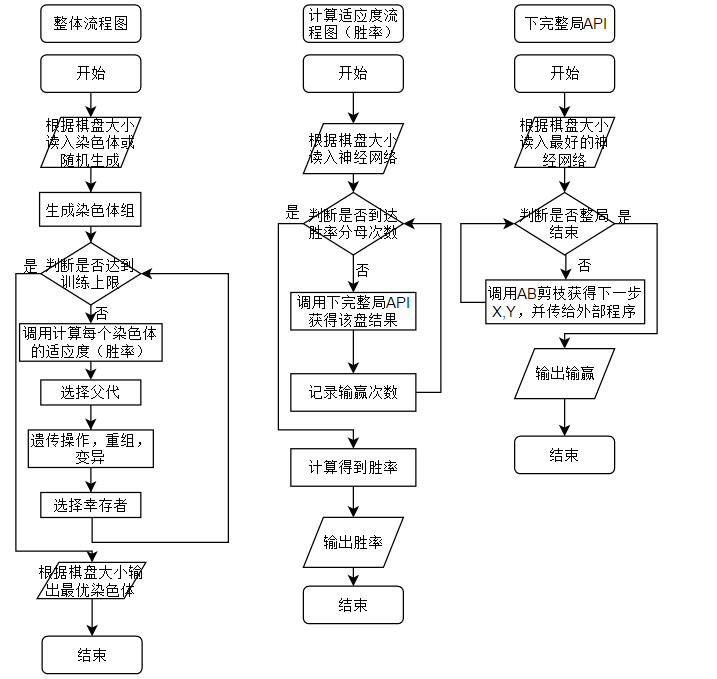
剪枝是Alpha-Beta搜索的核心，在2步思考的过程中，评价函数对于beta搜索的结果，仅仅保存min值，而对于alpha搜索的结果，仅仅保存max值，而当min < max时，代表着当前节点再扩展下去也是无用的，不可能影响到最终的max值，便可以直接舍弃。于是此处进行剪枝。剪枝的引入，进一步的提高了搜索的效率。

4.2.3.2 功能

Alpha-Beta搜索，是最接近底层的工作，是performance的实现，是下棋操作的直接执行者，为GA，神经网络对落子操作的提供了基础。

4.3、流程图

因为本程序可以自适应的学习，及下棋的过程就是学习的过程，所以主过程为GA过程。



5、不足之处

本人工智能的训练时间很短，相关神经网络权值，遗传算法优化程度有限，故仍远不及人类五子棋水平，和其他AI相比也有一定的差距。

6、结语

“事非经过不知难”。从第一次例会确定成员分工，到这个项目的最终完成，每一个微小的进展，都凝聚了我们小组全体成员的心血。为了保质保量完成任务，我们小组成员各司其职，各负其责，有的奋战在图书馆和网络上检索资料，有的通宵达旦地编写、调试程序，项目的每一部分，每一个点都有专人负责。开发过程中，我们开了三次例会，对项目开展进度严格把控，确保保质保量完成任务。遇到问题大家及时交流沟通，集思广益，充分发挥集体智慧。所有人都真真正正地切身参与到这次的项目开发工作中来。

“雄关漫道真如铁，而今迈步从头越”。“人工智能导论”这门课程的学习已经结束了，通过这门课的配套大作业，我们小组每个人都加深了对AI的理解，对人工智能的基本原理和基本方法有了更深的理解，而这些，都是听课，看书所不能得到的；同时，小组每位成员都参与到代码编写工作之中，通过实际项目开发，提高了编程能力；通过老师深入浅出的讲解，对自己所学科目产生了浓厚的兴趣，并下定决心一定要把这门课学好；通过小组成员的分工合作，我们提高了团队协作能力，并且知道了如何准确无误地表达自己的观点。

“学习意味着改变”。课程学习已经结束了，我们学到的也只是AI的基本方法和基本原理。但是，通过时间不长的学习，我们却明白了学习的目标，掌握了学习的方法，也对人工智能有了入门级的认识，从而为以后的学习和工作打下了坚实的基础。

在此要衷心地感谢刘老师一学期的付出。刘老师尽职尽责，上课时总能把晦涩难懂的原理用幽默风趣的语言讲出来，极大地提高了同学们的学习积极性，克服了同学们的畏难情绪；对于同学们提出的问题，刘老师从来没有回避过，而是深入细致地分析，充满耐心地地讲解，直到同学真正理解掌握为止。谢谢刘老师,谢谢为这门课付出的各位助教！