
Lab 1: A* search algorithm for n-puzzle

1 Intro

N-puzzle Problem 数字华容道 (n-puzzle) 一般是指在3x3的九宫格上随意排列1至8共八位数字，通过利用剩余的一个空格进行数字的上下左右移动，最终达到数字按行有序排列的问题。其大致过程如下图所示：

$$\left\{ \begin{array}{ccc} 5 & 3 & 6 \\ 1 & 8 & 4 \\ 7 & 2 & \end{array} \right\} \Rightarrow \left\{ \begin{array}{ccc} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & \end{array} \right\}$$

类似的问题还有最终状态不同的八数码问题，但本质上都可以看作是从初始状态转化为有序终态的搜索问题，搜索空间是整个九宫格各种可能排列的状态。由此，数字华容道同样可以扩展为4阶5阶甚至任意更高阶的问题，越高阶问题的搜索空间越大，也将花费更高的时间和空间开销。

A* Algorithm A* (A Star) 算法是一种很常用的路径查找和图形遍历算法。它有良好的性能和准确度，可以看作是Dijkstra算法的扩展，但由于借助启发函数的引导，A*算法通常拥有更好的性能。对于每个可能的搜索状态，我们将其称作节点，A*算法通过函数 $f(n)$ 来计算每个节点的优先级：

$$f(n) = g(n) + h(n)$$

其中 $f(n)$ 是综合优先级，节点的 $f(n)$ 越小则优先级越大。 $g(n)$ 是节点 n 距离初始节点的代价， $h(n)$ 是节点 n 距离终点的预计代价，也即A*算法的启发函数，通常是A*算法的设计核心。在运算过程中，每次选择优先级最高的节点进行遍历，A*算法用open表和close表来分别表示待遍历的节点和已经遍历的节点。

2 Task

TODO

- 代码：实现A*算法，能够给出任意3阶华容道问题的解，即状态路径
- 报告：运行结果截图
- 附加：数字华容道是否有解；修改A*算法，当问题存在多于 n 个解时，求解前 n 个最好的解；给算法实现扩展至高阶问题（附加题不作考察，仅供引导思考）

Note

- 可以基于给定的python code base（已实现棋盘类、广度优先算法），也可以用任意编程语言从零开始构建，为了和后续实验语言保持一致建议用python
- 给定的code base有提供根据给定随机种子随机生成的可解华容道问题，建议多尝试几种随机生成的初始问题，或者自己输入的问题
- 关于实验分数评定：是否正确实现了A*算法、能否正常运行给出完整解，完成目标即可；附加题不作考察但鼓励大家积极思考；可视化细节、报告长度等细节不作要求，请勿在不重要的细节上内卷

Submit

- 2022xxxxxx_xiaoming_lab1.zip (./code ./report.pdf)
- <https://k.ruc.edu.cn>, DDL 2024.09.26 23:59