

利用状态空间法解决八数码问题

学号 21160928 姓名 赵泽宇 完成时间 2018 年 10 月 19 日

1. 八数码简介

在 3×3 的方格棋盘上，摆放着 1 到 8 这八个数码，有 1 个方格是空的，其初始状态如图 1 所示，要求对空格执行空格左移、空格右移、空格上移和空格下移这四个操作使得棋盘从初始状态到目标状态。

2. 八数码问题的状态空间法表示

状态空间法是一种基于解答空间的问题表示和求解方法，它是以状态和操作符为基础的。在八数码问题的宽度优先搜索技术中，状态码由 11 个字符组成，包括父节点编号，操作符和 9 位状态；在八数码问题的深度优先搜索技术中，状态码由 12 个字符组成，包括父节点编号，操作符、9 位状态和深度；在八数码问题的深度优先搜索技术中，状态码由 12 个字符组成，包括本节点编号、父节点编号，操作符、9 位状态、深度、估价和总代价。

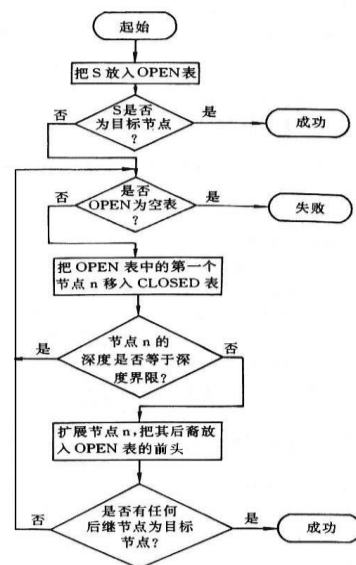
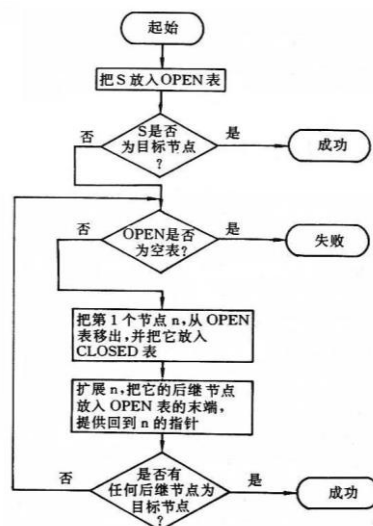
3. 八数码问题的盲目搜索技术概述

宽度优先搜索的基本思想是：从初始节点 S_0 开始进行节点扩展，考察 S_0 的第 1 个子节点是否为目标节点，若不是目标节点，则对该节点进行扩展；再考察 S_0 的第 2 个子节点是否为目标节点，若不是目标节点，则对其进行扩展；对 S_0 的所有子节点全部考察并扩展以后，再分别对 S_0 的所有子节点的子节点进行考察并扩展，如此向下搜索，直到发现目标状态 S_g 为止。因此，宽度优先搜索在对第 n 层的节点没有全部考察并扩展之前，不对第 $n+1$ 层的节点进行考察和扩展。

宽度优先搜索的盲目性较大，缺点是当目标节点离初始节点较远时，将会产生许多无用节点，搜索效率低。优点是宽度优先搜索总可以得到最优解。

深度优先搜索的基本思想是：从初始节点 S_0 开始进行节点扩展，考察 S_0 扩展的最后 1 个子节点是否为目标节点，若不是目标节点，则对该节点进行扩展；然后再对其扩展节点中的最后 1 个子节点进行考察，若又不是目标节点，则对其进行扩展，一直如此向下扩展。当发现节点本身不能扩展时，对其 1 个兄弟节点进行扩展；如果所有的兄弟节点都不能够扩展时，则寻找到它们的父节点，对父节点的兄弟节点进行扩展；依次类推，直到发现目标状态 S_g 为止。因此，深度优先搜索法存在搜索和回溯交替出现的现象。

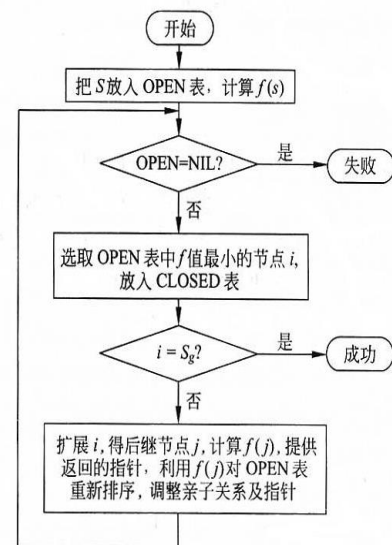
在深度优先搜索中，搜索一旦进入某个分支，就将沿着该分支一直向下搜索。如果目标节点恰好在此分支上，则可较快地得到问题解。但若目标节点不在该分支上，且该分支又是一个无穷分支，就不可能得到解。



4. 八数码问题的启发式搜索技术

我使用的估价函数为节点 n 的状态与目标状态之间数字不在位的个数（错放棋子的个数，不算空格）。算法的基本原理是利用问题拥有的启发信息来引导搜索，达到减少搜索范围、降低问题复杂度的目的。具体步骤如下：

1. 将初始节点 S_0 放入 Open 表中；
2. 如 Open 表为空，则搜索失败，退出；
3. 把 Open 表的代价最小的一个节点取出，放入到 Closed 表中，并把该节点记为节点 n ；
4. 如果节点 n 是目标节点，则搜索成功，求得一个解，退出；
5. 扩展节点 n ，生成一组子节点；
- 6.1 对既不在 Open 表中也不在 Closed 表中的子节点，计算出相应的估价函数值，然后将该节点放入 Open 表；
- 6.2 对于已经在 Open 表中的节点，比较两者代价，将代价小的节点放入 Open 表，代价大的节点删除；
- 6.3 对于已经在 Closed 表中的节点，比较两者代价，若新节点代价较小，将新节点放入 Open 表，代价大的节点删除；
7. 转到 2。



5. 例子及分析

(1) 简单例子

初始状态

2	8	3
1	6	4
7		5

目标状态

1	2	3
8		4
7	6	5

表 1：八数码问题 1 的运行记录表

算法	步数	操作符	OPEN 表大小	CLOSED 表大小	CPU 运行时间 (ns)
宽度优先搜索算法	5	上上左下右	21	26	未测到
深度优先搜索算法	11	右上左上左下 右右下左上	7	266	1 000 000
启发式搜索算法	5	上上左下右	7	7	未测到

(2) 稍复杂的例子

初始状态

	2	4
1	5	3
8	7	6

目标状态

1	2	3
8		4
7	6	5

表 2：八数码问题 2 的运行记录表

算法	步数	操作符	OPEN 表大小	CLOSED 表大小	CPU 运行时间(ns)
宽度优先搜索算法	12	右右下左上左下 下右右上左	1122	1385	18 000 000
深度优先搜索算法	12	右右下左上左下 下右右上左	3	6491	589 000 000
启发式搜索算法	12	右右下左上左下 下右右上左	75	108	4949808

6. 体会与致谢

通过本次实验，我掌握了如何使用状态空间法来表示知识，还掌握了盲目式搜索技术和启发式搜索技术。在编写宽度优先和深度优先搜索的程序时，使用了 C 语言面向过程的编程方法；在编写启发式搜索的过程中发现使用 C 语言编写过程较为复杂，请教了梁佳威同学，从网络上学习了 C++ 和 STL 库，但由于对 C++ 不熟悉，不能很快编写出相应程序，于是使用 Python 编写了启发式搜索算法，Python 提供了很多操作列表的方法，极大地简化了代码。非常感谢在本课程中帮助过我的同学。

7. 实验程序简单说明

本程序分两部分，第一部分为盲目式搜索，使用 C 语言编写，第二部分为启发式搜索，使用 Python 编写，使用方法大致相同。根据提示输入选择算法、初始节点的生成方法（包括随机生成、用户输入、默认初始节点），然后程序会自动打印 open 表、closed 表、变化过程、操作符以及运行时间。