



# 物理设计 Project

## 项目文档

迷宫算法布线

21307140085 秦振航 21307140085@m.fudan.edu.cn

21307140079 杨远达 yangyuanda922@gmail.com

2024 年 6 月 21 日

---

# 目录

<b>1</b>	<b>任务需求分析</b>	<b>3</b>
1.1	问题 . . . . .	3
1.2	需求分析 . . . . .	3
<b>2</b>	<b>团队分工情况</b>	<b>3</b>
2.1	分工情况 . . . . .	3
<b>3</b>	<b>程序实现思路与数据结构</b>	<b>3</b>
3.1	代码结构 . . . . .	3
3.2	命令行参数解析 . . . . .	4
3.3	实现思路 . . . . .	4
3.4	A* 算法 . . . . .	4
3.4.1	算法步骤 . . . . .	4
3.4.2	代价 F 的计算 . . . . .	5
<b>4</b>	<b>测试方法和验证</b>	<b>5</b>
4.1	测试方法 . . . . .	5
4.2	验证结果 . . . . .	5

---

# 1 任务需求分析

## 1.1 问题

本题需要实现的是迷宫布线算法。可以采用原始迷宫算法，也可以实现改进的 A\* 迷宫算法。迷宫的全重假设为 1。布线后的占据的迷宫位置要成为新的障碍。本题给定 prob4.txt 文件，描述的是一个矩阵，每一位表示该坐标上的路径状态：0 表示无障碍，1 表示障碍，2 表示有要连接的引脚。本题需要用迷宫布线算法将引脚都连接上，并且绕过障碍物。

## 1.2 需求分析

本题需要实现将输入文件 prob4.txt 中的引脚连接起来，并且绕过障碍物。引脚之间不重复连接

# 2 团队分工情况

## 2.1 分工情况

姓名	分工
秦振航	算法的主要实现
杨远达	报告的撰写，实现思路的讨论

# 3 程序实现思路与数据结构

## 3.1 代码结构

项目的源代码在 Astar/src 文件夹中，包含以下模块：

1. parser 输入文件解析模块。beginitemize
2. parser\_file 解析函数，输入迷宫文件，返回一个存储迷宫信息的二维数组。enditemize
3. main
  - main 入口函数。
  - find\_nearest\_point 寻找距离引脚 p 最近的已连接点。
4. Astar A\* 算法实现函数

- 
- Position 二维空间点的类
  - MapNode 存储遍历图的过程中点的信息，包括点的位置以及其距离函数 F,H,G

## 3.2 命令行参数解析

-i 输入文件名

## 3.3 实现思路

整体上采用 A\* 算法，解析文件之后得到引脚所在的位置，将第一个引脚位置放入到 *connected\_pin* 的向量里，然后遍历剩下的引脚。对于每个引脚，找到在 *connected\_pin* 里曼哈顿距离它最近的已连接点，将该点作为 A\* 寻径算法的目标点。在 A\* 算法中，会将最终结果路径上的点放入到 *connected\_pin* 里，作为已连接点。这样遍历所有应连接的引脚，即可以最短的线长连通所有点。

## 3.4 A\* 算法

A\* 算法作为一个启发式的寻径算法，与一般寻径算法的区别在于其在计算路径距离时不但考虑了距离起点的路径，即实际代价 G，还考虑了距离终点的距离，即启发式估计 H，两者相加的 F 作为总代价。

### 3.4.1 算法步骤

首先将起点放入开列表中。

在寻径过程中，从开列表中寻找到总代价最小的点，将其锁定，放入闭列表中，依次遍历四周的点。

1. 如果是目标点，结束，返回。
2. 如果该点已经被锁定，已放入到了闭列表中，或者该点是障碍或者非法点，跳过该点。
3. 如果该点未被访问，计算他的代价，将父节点设置为取出的点，将他标记已访问之后放入开列表里。
4. 如果已在开列表中，且不是非法点，则重新计算其代价 F，进行松弛，如果新计算的代价比原来的代价小，则更新开列表中对对应点的代价和父节点，如果代价更大则不管。

循环直到找到目标点。

### 3.4.2 代价 F 的计算

在寻径过程中，代价 F 由实际代价 G 和启发式估计 H 两部分组成。

实际代价 G 表示从起点到当前点的实际距离，在遍历节点的过程中，子节点的 G 为其父节点 +1。

启发式估计 H 表示从当前点到终点的预估距离，在本题中，启发式估计 H 为当前点到中点的曼哈顿距离。

## 4 测试方法和验证

### 4.1 测试方法

在 Astar 文件路径下，执行命令行：

```
.\build\bin\Astar.exe .\prob4.txt > .\output.txt
```

也可以通过 cmake 自行编译。

### 4.2 验证结果

输入后的结果如下 (见 output.txt)

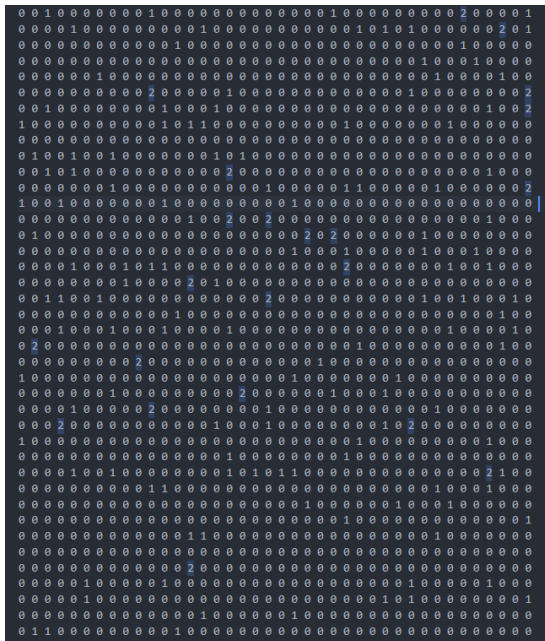


图 1: 输入网格

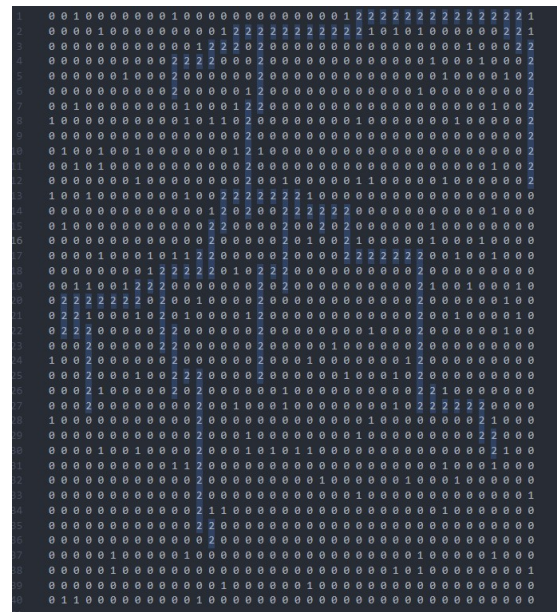


图 2: 输出网格