

主干管布置算法

2020年4月6日 浙江大学郑友怡团队 庄稼捷

1. 问题描述

本算法的目标是为卫生间自动铺设管线连接一个进水口和若干出水口，使得管线总长度尽可能小，弯折尽可能少。

卫生间内存在大量用水设施，需要铺设管线来联通进水口和出水口。算法需要解决的问题是，如何保证在管线总长度尽可能小的情况下，同时满足：

- 管线的弯折尽可能少
- 避免四通的产生
- 管线尽可能靠墙
- 管线距离墙面（障碍物）为100mm的倍数，默认200mm
- 对于15m*15m的范围，30s内应该出结果
- 管线横平竖直

2. 算法接口

输入：

- X轴方向向量（用于校正区域，使其横平竖直）
- 区域的闭合空间
- 单个起点（可能在轮廓线内）
- 多个终点
- 不可走区域

输出：

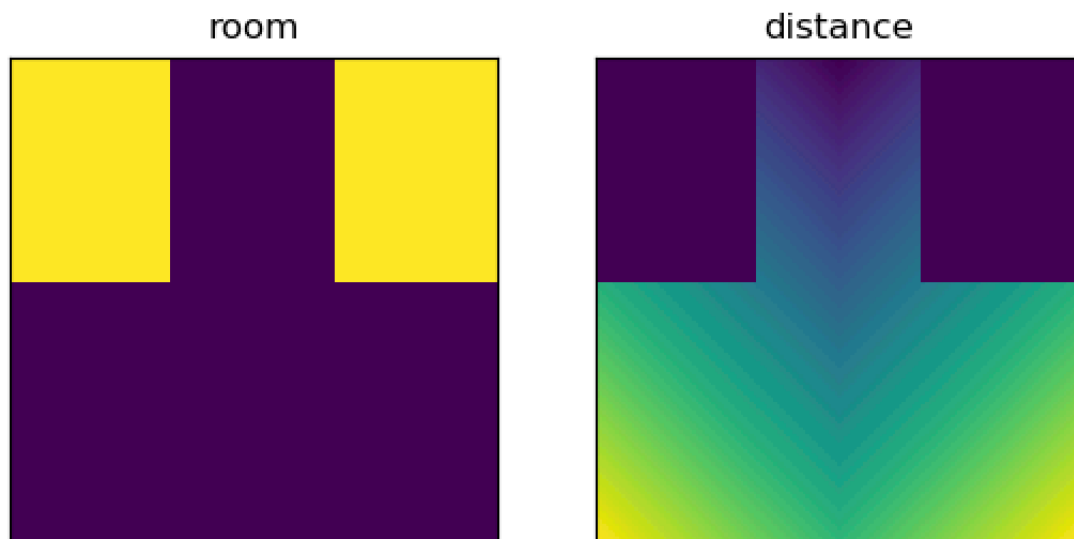
- 联通起点及各个终点路径。

3. 算法描述

3.1 封闭区域确定和分析

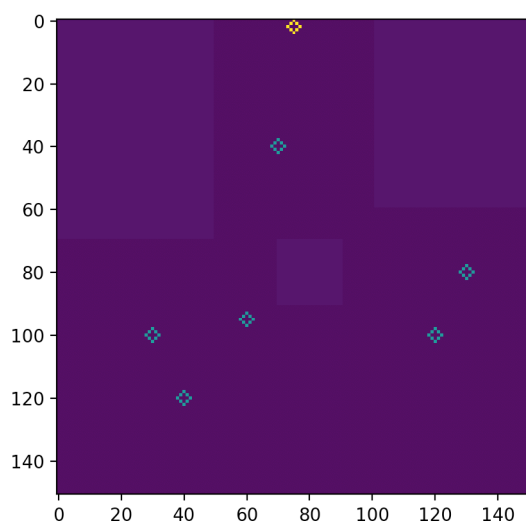
根据输入的封闭区域和障碍物，得出bounding box覆盖整片区域，并记录可通行和不可通行区域。

在获得了空间信息之后，计算空间内到给水起点的最短距离。将连续变为离散，这里假设管线以100mm为单位进行延伸，因此实际上计算的网格点到给水起点的距离。举个简单例子，下图一（room）中，紫色部分为空间可通行面积，图二（distance）为空间点到起点的距离，距离越小颜色越深。

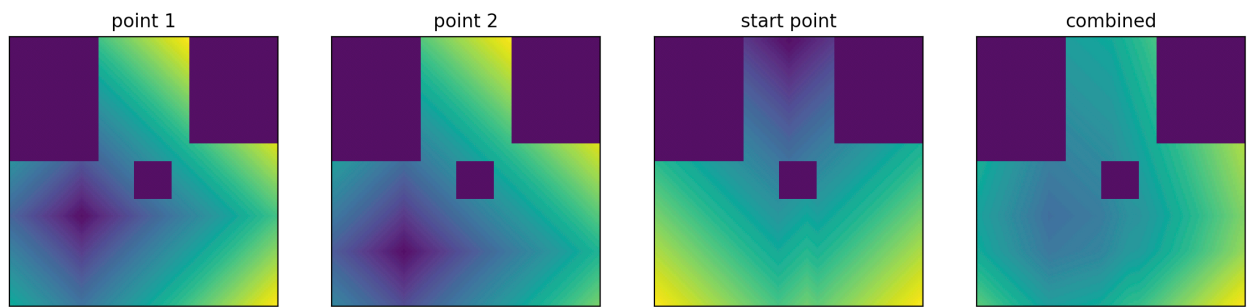


3.2 路径合并

如下图，为起始点和给水点的分布图，采用之前的算法，即找出每个点到起始点的最短路径，然后合并，算法非常复杂，调整起来也很麻烦。因此对算法进行了改进。



新的算法采用合并的方式来进行。管理一个list，每次，我们从这个list中取出两个不同的点 $p_1 p_2$ ，计算distance map（同3.1），得到 $distance_1$ ， $distance_2$ ，以及最开始获得的 $distance_{start}$ 。将这三个distance map合并，得到 $distance_{new}$ ，其上的任意一点的数值，等于 p_1 到该点的最短距离+ p_2 到该点的最短距离+ p_{start} 到该点的最短距离。如下图中的combined，可以看到，颜色最深的那一点，就是总距离最短的一点。找到该点，然后找出该点到起始点和 $p_1 p_2$ 到该点的最短距离，就获得了总长度最短的路径。这样的好处是，不需要像之前一样考虑加障碍物后调整，可以直接求出路径。



简要描述算法如下：

输入点 $p_{start}, p_1, \dots, p_n$

初始设置 list $L = \{ p_1, \dots, p_n \}$

while $\text{len}(L) > 1$:

$n = \text{len}(L)$

$p_{min} = -1$

 距起始点最长距离 $S = -1$

 for i in $\text{range}(n)$:

 for j in $\text{range}(i+1, n)$:

 取出 $p_1 = L[i], p_2 = L[j]$, 以及对应的 distance map D_1, D_2 (如果没有, 则计算出来)

 得到 combined distance map $D_{combined} = D_1 + D_2 + D_{start}$

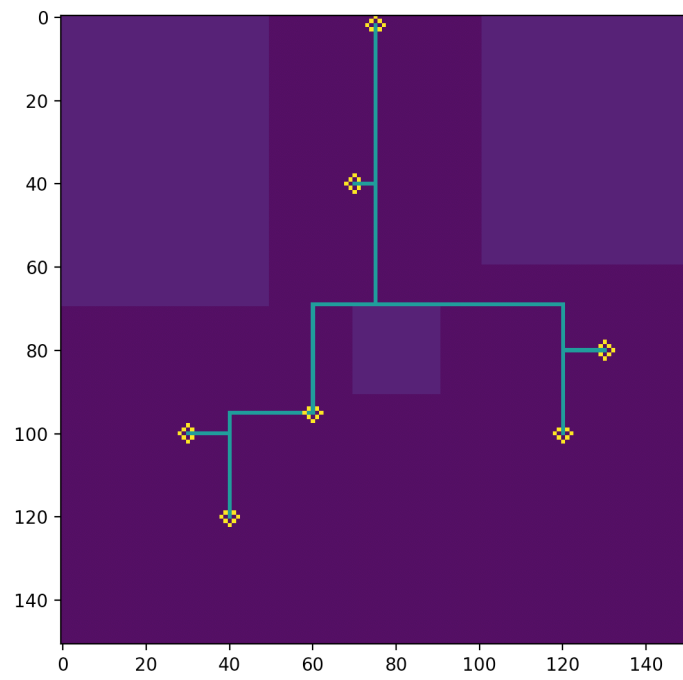
 找到最短总距离点, 如果该点到起始点的距离比 S 长, 则修改 p_{min} 为该点

 保存此次计算结果

 将 p_{min} 对应的 p_1, p_2 从 L 中去除, 加入 p_{min}

顺次连接, 计算各点到起始点的路径

最终, 找到最短路径组:



接下来的任务

- 进一步修改，弯折少&靠墙，优先级？
- 从geojson中提取出数据，进行测试