



智能无人机技术设计实践

--实验4指导书

李潇翔

联系方式: lsx17@mails.tsinghua.edu.cn

时间: 2019.10.26





目 录

- 1. MATLAB 安装与使用
- 2. A*算法实现



1 MATLAB安装与使用

- ◆ 连接校园网，从信息门口登录信息化用户平台，在计算软件模块下载安装
- ◆ 下载链接：<http://its.tsinghua.edu.cn/column/jsrj>，安装过程省略
- ◆ MATLAB 大部分内容向下兼容，课程指导以2017b版本为例

首页 / 校园正版化软件下载 / 计算软件

操作系统

防病毒软件

开发软件

办公软件

计算软件

计算软件

• 清华大学MATLAB全校授权许可证更新指南	2019-04-27
• MATLAB 在线课程与学习资源（2019版）	2019-03-12
• 清华大学MATLAB网络版转为单机版指南	2018-11-23
• MATLAB单机版安装指南	2018-11-23
• Matlab安装手册及相关文件	2018-08-20
• Matlab R2019a for Windows	2019-07-03
• Matlab R2019a for Mac	2019-07-03
• Matlab R2019a for Linux	2019-07-03
• Matlab R2018b for Windows 64位（DVD1）	2018-09-29
• Matlab R2018b for Windows 64位（DVD2）	2018-09-29



1 MATLAB安装与使用

◆ 基础的MATLAB使用教程

- ◆ 电子系同学可参见先修课程《MATLAB高级编程与工程应用》。
- ◆ 其他院系同学请自学，在此不再赘述。



2 A*算法实现

提示：所有资料在网络学堂的实验4资料文件夹下。

◆ Demo演示：heuristic_demo.m

详情请参看程序的注释，在此对程序做几点说明。

```
%%
startPosition = [3 1];%Initial point
goalPosition = [14 12];%End point

% Map initialization, 15 for obstacles, 0 for free space
map = zeros(15,15);%It's the simplest demo
%load('map_test.mat');%use this map to check your A* algorithm

[mapRow, mapCol] = size(map);

if map(startPosition(1), startPosition(2))
    error('Parameters Error! in startPosition');
elseif map(goalPosition(1), goalPosition(2))
    error('Parameters Error! in goalPoaition');
end
```

这只是一个说明各接口的演示程序；
在A*算法的测试中请使用map_test作为地图



2 A*算法实现

◆ Demo演示: heuristic_demo.m

```
%%  
% Variables in the two structures can be changed, but 'closeList' need  
% to be saved as results and please reserve 'row', & 'col' & 'f'.  
% Here, 'row' and 'col' are index numbers of the nodes' row and the coloum,  
% 'f' caculates the distance  
% 'comefrom' is an index of its parent nodes  
  
closeList = struct('row', 0, 'col', 0, 'f', 0, 'comefrom', 0); %The set has been estimated  
closeListLength = 0;  
openList = struct('row', 0, 'col', 0, 'f', 0, 'comefrom', 0); %The set to be estimated  
openListLength = 0;
```

closeList是存储搜索结果的变量，需要保存并作为结果存储。
在设计算法的过程中结构体内部可以自由增减变量，但坐标 (row, col) 和距离 f 需要保留。
comefrom是作为父节点的指标方便最后打印输出，也建议保留。



2 A*算法实现

◆ Demo演示: heuristic_demo.m

```
%%  
%Search algorithm  
direction = [1, 1; -1, -1]; %Scanning direction  
openList(1).row = startPosition(1);  
openList(1).col = startPosition(2);  
openListLength = openListLength + 1;  
openList(1).f = 0;  
|  
while true  
    f = openList(1).f;  
    nodePosition = 1; %  
    for i = 1:openListLength  
        if f > openList(i).f  
            f = openList(i).f;  
            nodePosition = i;  
        end  
    end  
end
```

对邻居节点的搜索方向，在A*中支持对角搜索，所以邻居最多需要搜索8个。



2 A*算法实现

◆ Demo演示: heuristic_demo.m

```
for i = 1:size(direction,1)
    newPosition = [closeList(closeListLength).row, closeList(closeListLength).col] + direction(i, :);
    if (all(newPosition > 0) && newPosition(1) <= mapRow && newPosition(2) <=...
        mapCol && map(newPosition(1), newPosition(2)) ~= 15)% Exclusion of obstacles
        flag = false;
        for j = 1:closeListLength
            if closeList(j).row == newPosition(1) && closeList(j).col == newPosition(2)
                flag = true;
                break;
            end
        end
    end

    if flag
        continue;
    end

    openList(openListLength).row = newPosition(1);
    openList(openListLength).col = newPosition(2);
    openList(openListLength).comefrom = closeListLength;
    openList(openListLength).f = heuristic_estimate_demo(newPosition, goalPosition);
end
```

在可行域内搜索，排除障碍物与边界

已经估计完成的节点不必重复估计了

计算距离的启发式函数

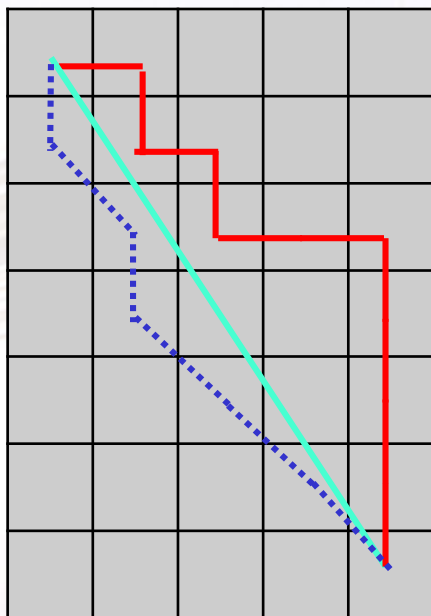


2 A*算法实现

◆ Demo演示: heuristic_estimate_demo.m

```
function cost = heuristic_estimate_demo(start, goal)
    %Calculation of diagonal distance
    dx = abs(start(1) - goal(1));
    dy = abs(start(2) - goal(2));
    cost = dx + dy + (sqrt(2)-2)*min(dx, dy);
end
```

计算的是对角距离



—— 曼哈顿距离

..... 对角距离

—— 欧几里得距离

```
dx = abs(start(1) - goal(1));
dy = abs(start(2) - goal(2));
cost = dx + dy;
```

```
dx = abs(start(1) - goal(1));
dy = abs(start(2) - goal(2));
cost = dx + dy + (sqrt(2)-2)*min(dx, dy);
```

```
cost = sqrt((start(1) - goal(1))^2 + (start(2) - goal(2))^2);
```



2 A*算法实现

◆ Demo演示: heuristic_demo.m

```
while true
    if(closeList(figflag).row == startPosition(1) && closeList(figflag).col == startPosition(2))
        break;
    end
    path = [[closeList(figflag).row, closeList(figflag).col]; path];
    figflag = closeList(figflag).comefrom ;
end

for k = 2:size(path,1)
    if(closeList(k).row == goalPosition(1) && closeList(k).col == goalPosition(2))
        break;
    end

    map(closeList(k).row, closeList(k).col) = 20;
    pcolor(map)
    pause(0.3);
end
```

利用父节点的索引指标将最短路径找出

这部分是可视化结果显示，方便确认结果是否正确。



作业提交

◆ 任务要求

修改 “Astar_v0.m” , 实现A*算法。满足:

- ① 使用 “map_test.mat” 作为测试地图;
- ② 保留closeList以及其中的row, col, f 作为结果存储, 其他变量可自行设计;
- ③ 搜索方向 direction 已经定义好, 可以改变搜索顺序但是需要满足对角搜索条件;
- ④ 启发式距离函数可以自定义; 也可以使用heuristic_estimate_demo.m;
- ⑤ 最后存储时请将文件名改为 “result_yourstudentID” , 记录path与closeList;
- ⑥ 最后打包提交自己的源代码与 “result_yourstudentID.mat” 文件。
- ⑦ 源代码请**详细地**写出注释, 也可以另写一份实验报告做说明。



谢谢!

答疑地点：罗姆楼11层101房间