福州大学

《无人机控制技术》实验报告

题 目： 集群无人机队形重构仿真

姓 名： 张少乾，查晴

学 号： 190127132，190120064

学 院： 电气工程与自动化学院

专 业： 控制工程

年 级： 2019级

指导教师： 黄捷

2020年6月25日

# 集群无人机队形重构仿真

## 研究背景

四旋翼无人机具有成本较低、设备简单、飞行时间灵活等特点，近些年被广泛应用于军事和民用领域，如目标侦察、应急救援、农业植保、无人机灯光表演。随着任务复杂度的增加，单架无人机往往难以满足任务需求，因此无人机集群控制及其应用由此成为目前的研究热点，多无人机集群能够提高执行任务效率和灵活度。无人机队形变换控制方法是实现多无人机编队飞行的前提，集群无人机队形重构问题是我们要考虑的一个重要问题，让每架无人机都能从初始位置无碰撞的到达最终位置，进而保证队形重构过程中代价最小或能耗最优。其中目标分配问题最多利用匈牙利算法进行解决，但是在多无人机轨迹规划时普遍存在计算难度大、规划时间增长、规划效率难以满足实际需求的问题。因此，探索计算简便、效率高的多无人机路径规划算法是目前迫切需要的。

## 任务要求

### （1）实验描述

十九架无人机组成的编队如图2.1所示，由F队形切换至Z队形，如果忽略本身以及外在约束条件的情况下，将会有多种不同的移动方案。但实际情形下无人机飞行距离越长，耗能也就越多，部分无人机到达目标的距离过长，消耗电力过多，从而提前降落。

。

图2.1： 编队实景图

使用传统匈牙利算法来解决队形变换的目标分配问题，其编队切换仿真图如下图，其中绿色方块表示无人机在F队形中的位置，红色圆点表示Z队形中无人机集群所要到达的各目标航点，黑色直线表示经分配后无人机由初始位置到达目标位置的直线路径。



图2.2： 匈牙利算法路径图

由上图可以看出，部分无人机到达目的地的直线路径过长，而大部分无人机初始位置与目标位置重合耗能少。移动路径长短不一，造成无人机耗能相差较大。所以我们需要寻求一种移动方式，使各个无人机移动路径长度趋于一致。

### （2）实验要求

1、改进分配方案，使无人机在编队切换过程中飞行路径相近，飞行至目标航点时间一致，避免无人机耗能相差较大的问题。

2、集群无人机空中个别无人机能量不足或故障情况需退出，编队无人机数目发生变化，为维持编队队形需重新规划各无人机所在位置。

## 解决方案

集群无人机编队重构中考虑时间的一致性问题，可扩展研究切换过程中各无人机的距离约束，规划编队重构中各无人机路径长度，在同样移动速度下，使集群无人机在避免碰撞的情形下尽量同时到达目标点。通过考虑无人机编队切换的距离大小，若在编队切换中各个无人机移动的距离相近，那么在避免碰撞的情况下到达指定位置，使整个机群飞至目标航点时间趋于一致。

和匈牙利算法类似的是，我们同样使位于变换图形边缘的九个重合目标不发生移动，但其余十个目标均发生移动（匈牙利算法中除边缘九个点还存在两个重合点，详情可见图2）。通过一个10\*10的代价矩阵（将移动距离设为代价），利用算法寻求出一个基于移动距离一致性为目标的优化分配方案。

## 设计流程

### 模型场景搭建

设置模型场景范围，通过建立一个x，y最大值均为100个单位的直角坐标系，用于将矩阵中的坐标信息转换为直观上的直角坐标信息。设置模型场景中相邻两架无人机距离为10个刻度。

### 位置映射

利用矩阵，将无人机的矩阵位置信息与模型坐标系中的位置进行一一映射。在模型坐标系中，存在两个无人机的位置距离信息，所以在映射过程存在对应关系。例如：两无人机的相聚间隔为10个距离单位，在矩阵第一行第二列的无人机在模型坐标中的具体坐标轴位置为(20, 90)；同理，在矩阵第一行第三列的无人机在模型坐标中的具体坐标轴位置为(30, 90)；故可得矩阵与模型位置的映射关系：

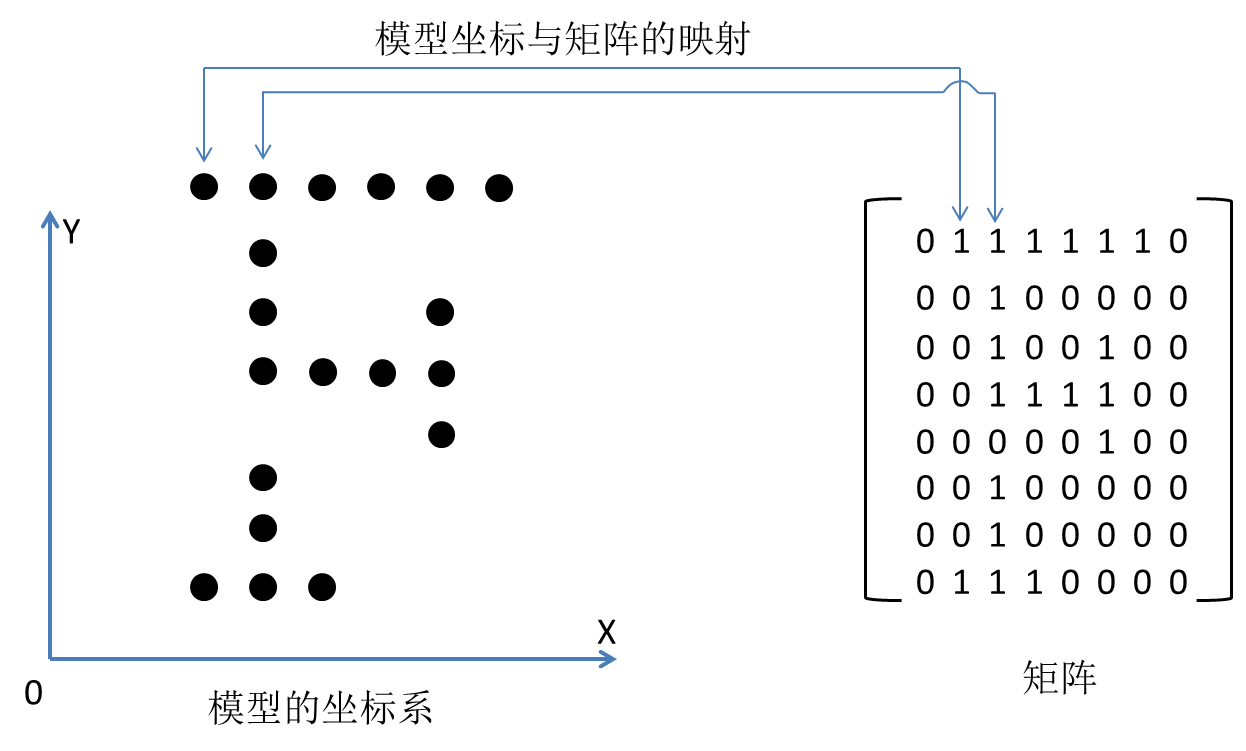


图4.1： 位置映射关系图

其中，(x1,y1) 表示矩阵的位置，width 表示模型两个无人机的位置距离。(X,Y) 表示无人机在模型中的仿真位置信息。映射关系在仿真中，用位置映射函数(calc.m)来完成这个对应关系。

### 移动模型

无人机想要按规律的移动，就必须给其进行逐个编号，算法的功能是告诉模型，哪一个ID的无人机要飞到哪一个位置，只要给模型报告位置信息即可。模型要做的就是怎么进行移动。

理论上无人机的移动是同时进行的，但是表现在程序上，是每一时刻重复扫描所有无人机的位置状态，进而感觉在移动。这个扫描时间（扫描步距）dt是可自定义设置，最小单位为1个时间单位，当所有无人机都到达指定位置，退出循环，返回队形变化的时间。

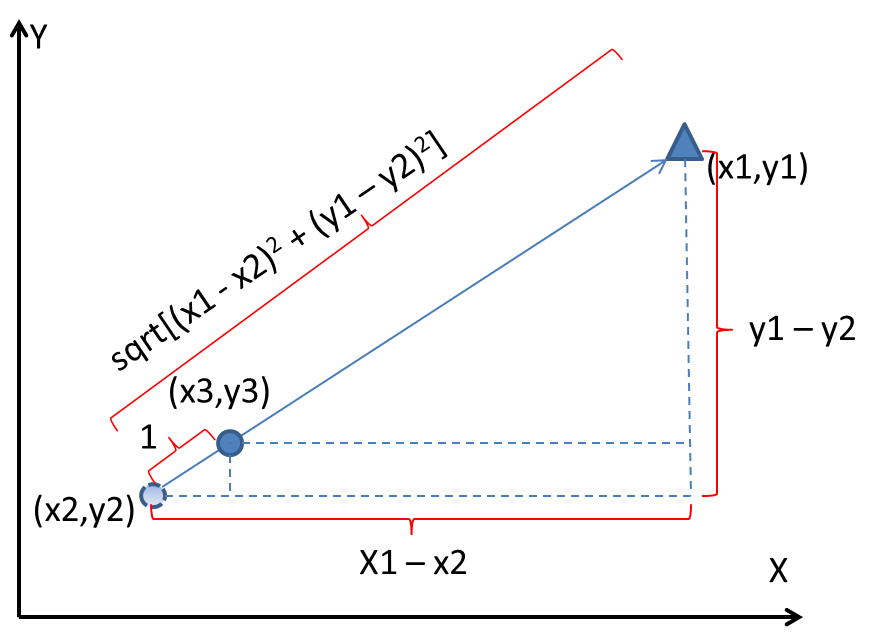
移动模型的核心函数是move函数，核心思想就是所有无人机无论在什么位置都能以最短距离的方式到达指定位置，两点之间直线最短，模型原理如下图所示：

图4.2： 移动模型图

假设无人机目标坐标为(x1,y1)与起始坐标为(x2, y2)均是已知，移动之后的坐标为(x3, y3)，我们已知的是从(x2,y2)的位置移动一个时间单位，移动的位置为1个距离单位，所以我们可以利用比例关系得容易获取下一位置信息(x3,y3) 的坐标。

### 避障模型

当算法根据起始坐标与终点坐标信息将得出的分配方案告诉模型之后，所有需要移动的无人机同一时刻开始由起始点沿直线向终点开始移动，其中不可避免的会产生碰撞，造成无人机坠落，所以集群无人机在队形变换中避障模块的设计不可忽略。

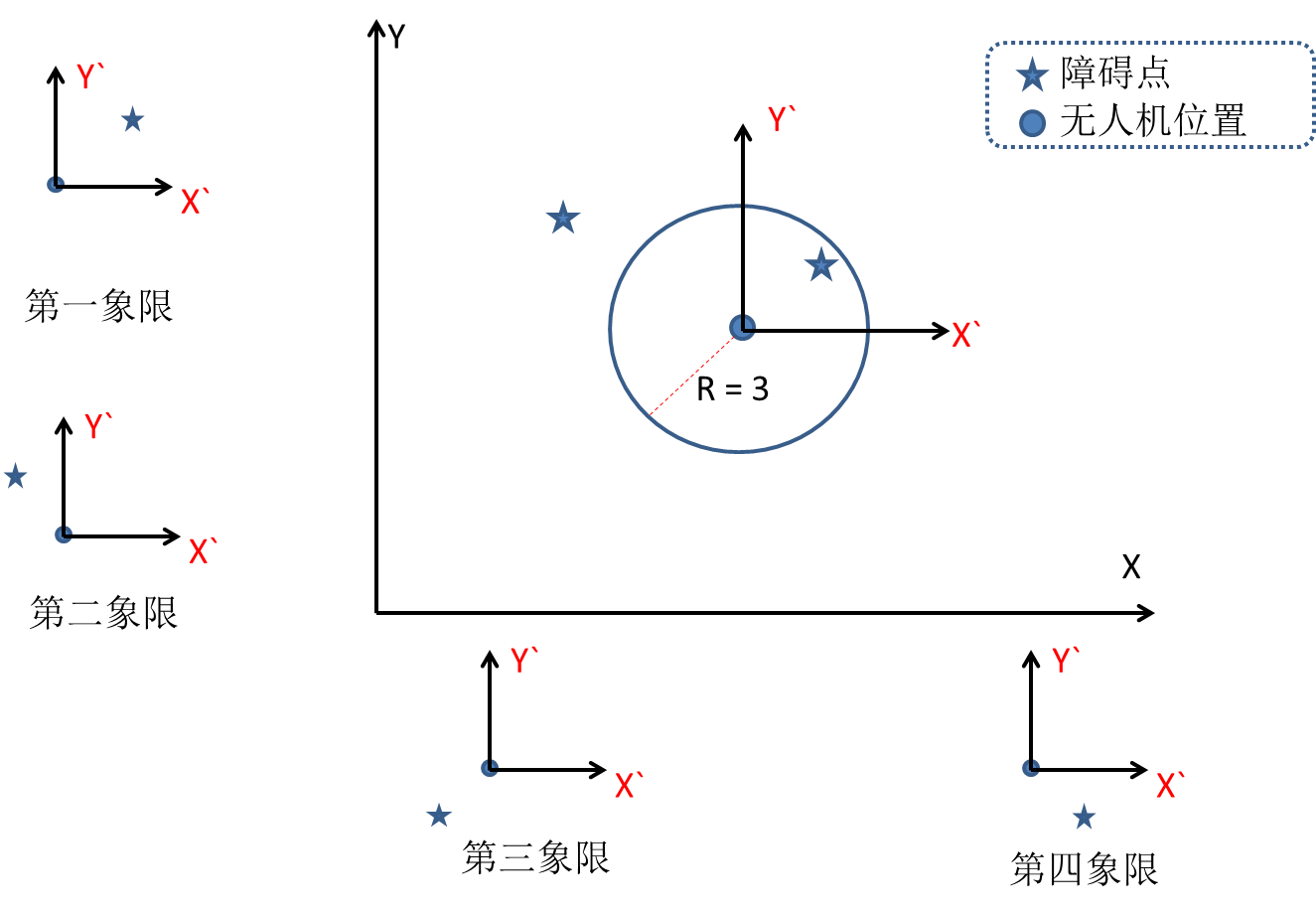


图4.3： 避障模型原理图

采用简单的避障方法，就是在无人机的3个距离单位的范围之内，就当做是障碍点。这时候就要进行相应的避让操作。

当障碍点位于无人机的第一象限，该点向左移动一个单位。

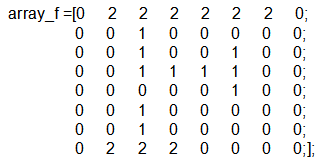
当障碍点位于无人机的第二象限，该点向右移动一个单位。

当障碍点位于无人机的第三象限，该点向右移动一个单位。

当障碍点位于无人机的第四象限，该点向左移动一个单位。

## 坐标安置

新建一个8\*8的零矩阵，通过将零矩阵中部分点变为1，在矩阵中产生一个F图案。由于在进行队形重构中位于边缘的重合点不再进行移动，因此我们将不移动的“1”变为“2”，从而方便我们调取移动点的数据，Z图案进行与F图案相同的操作。最后得到的矩阵如下：



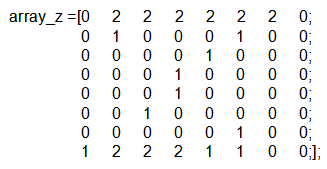


图5.1： 图案FZ在矩阵中的表示

其中array\_f中的“1”元素所在的所有位置为起点坐标，将起点坐标进行编号，生成T0矩阵。array\_z中的“1”元素所在的所有位置为终点坐标，将终点坐标进行编号，生成T1矩阵。

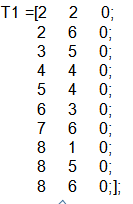
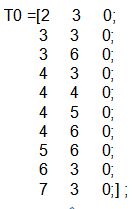


图5.2： 起点坐标与终点坐标集合

其中，T0的（2，3）为编号1的起点坐标，（3，3）为编号2的起点坐标，以此类推，T1中的坐标亦是如此。

## **代价函数**

将无人机从起点位置运行到指定终点位置的直线距离设置为运行代价，T0中的每个起点分别对T1中的十个终点求取代价，这样我们可以得到一个10\*10的代价矩阵，算法就是基于代价函数信息来寻求最优匹配。代价矩阵如下图所示

代价矩阵

1.0000 1.4142 4.1231 2.2361 2.8284 3.6056 4.4721 5.0000 4.1231 5.0990

3.0000 3.1623 1.0000 3.6056 2.8284 2.2361 2.0000 3.0000 5.0000 5.8310

2.2361 2.0000 1.0000 2.2361 1.4142 1.0000 1.4142 2.2361 3.6056 4.4721

2.2361 1.4142 2.2361 1.0000 0 1.0000 2.0000 2.2361 2.2361 3.1623

3.1623 2.2361 2.8284 1.4142 1.0000 1.4142 2.2361 2.0000 1.4142 2.2361

4.0000 3.0000 4.2426 2.0000 2.2361 2.8284 3.6056 3.1623 0 1.0000

5.8310 5.0000 4.0000 4.2426 3.6056 3.1623 3.0000 2.0000 3.1623 3.0000

6.3246 5.3852 7.0711 4.4721 5.0000 5.6569 6.4031 5.8310 2.8284 2.2361

6.3246 5.3852 5.0990 4.4721 4.1231 4.0000 4.1231 3.1623 2.8284 2.2361

6.7082 5.8310 5.0000 5.0000 4.4721 4.1231 4.0000 3.0000 3.6056 3.1623

图6： 代价矩阵

代价矩阵中第一列为T0的一号起点坐标分别对T1的十个终点坐标求取移动代价，第二列为T0的二号起点坐标分别对T1的十个终点坐标求取移动代价，第三列至第十列以此类推。

## **算法简介**

第一步：计算代价矩阵每一列的平均值，并找出每一列的最小值，带入下式来求得一个新均值。

min\_mean(i) = a\*min\_r+b\*ave\_vl;

其中min为第i列的最小值，ave\_vl为第i列的平均值，min\_mean为考虑最小值之后的新均值，用以将均值与最小值进行一个配比，从而达到更好的控制效果。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| a | 0 | 0.1 | 0.2 | 0.3 |
| b | 1 | 0.9 | 0.8 | 0.7 |
| 均方差 | 3.5009 | 0.4509 | 0.3085 | 0.3085 |

图7.1：a，b取值与均方差关系

将每列最小值与均值按不同的比例进行运算，结果发现0.2倍最小值搭配0.8倍均值运算，均方差与代价均会较小，结果更加优化。在试验对比阶段我们将会对比0.2倍最小值搭配0.8倍均值与0.1倍最小值搭配0.9倍均值的实验数据与仿真图，从而更加直观的看到哪一种配比使我们的仿真更加的优化。

图7.2为0.2倍最小值搭配0.8倍均值所求得的新均值。

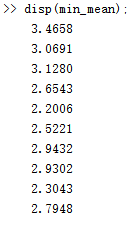


图7.2：代价矩阵每一列新均值

第二步：类似于匈牙利算法中用0标记最小值，利用上式求出的“平均值”，算出原代价矩阵中每列的各个代价值与其对应列的新均值之间的差值，最终得到一个均差代价矩阵，如图7.2所示。最后将均差代价矩阵中每列最小值标记为0，此时矩阵中每列都有且仅有1个0。

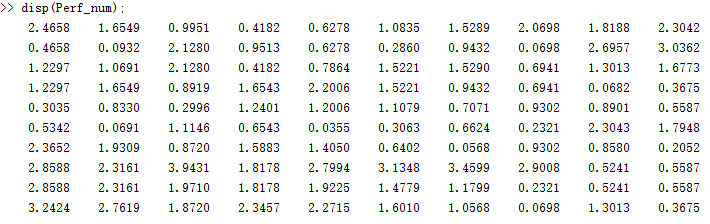


图7.2：均差代价矩阵

第三步：检查每行0的个数，若同一行出现多个0，说明此时有任务冲突，故需要对该行0所在的多列重新选0，此时进行第四步。标0后的均差代价矩阵如图7.3所示。

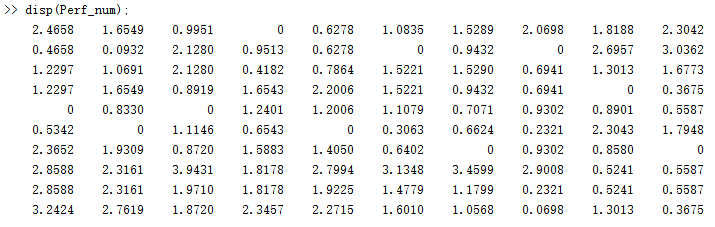


图7.3：标0均差代价矩阵

第四步：取出同行0所在的位置在原代价矩阵中的代价值，判断该代价值与总平均代价值之间差距的大小，将判断得到的差值大的那个位置对应均差代价矩阵中的那个0取为一个较大数100，使得后面重取0时该位置不被重取，而通过再找列最小值将均差代价矩阵中该列的原第二小的值标记为0；而其余列的0不变；这个步骤需要进行多次以使得得到10个独立0。

## **仿真实验**

### （1）实验数据

在matlab上运行仿真程序，实验数据如下：

最优匹配为:4 5 1 3 2 6 7 0 9 8；

T0的一号机移动到T1的四号号位，T0的二号机移动到T1的五号位，以此类推。

对应的代价分别为为:2.2361 2.2361 4.1231 2.2361 2.8284 2.8284 3.0000 3 .0000 2.8284 2.2361；

平均代价为2.7553；

最优均方差为0.3085；

运行时间41s（本设计中速度v=1)；

### （2）路径图

按照最优分配结果，将起始位置与对应的终点位置进行匹配，并画出两点直线路径，所得的移动路径如图8所示：



图8：移动路径图

## **均值配比对比试验**

### （1）实验数据

当按照0.1倍最小值搭配0.9倍均值运算，所得匹配结果为：5 4 1 3 2 6 7 10 9 8；

对应的代价分别为:3.1623 1.4142 4.1231 2.2361 2.8284 2.8284 3.0000 3.0000 2.8284 2.2361；

此时的均方差为0.4509，平均代价为2.7658，均大于按0.2倍最小值搭配0.8倍均值运算的均方差0.3085，平均代价2.7553，因此配比要按照0.2倍最小值搭配0.8倍均值的方式进行运算。

### （2）路径对比

仿真所得的移动路径对比图如下，通过图像对比，发现0.2+0.8的搭配效果要略好于0.1+0.9搭配。



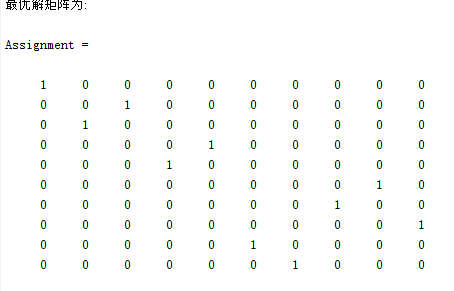
图9.1：0.2+0.8配比移动路径



图9.2：0.1+0.9配比移动路径

## **匈牙利算法对比试验**

运行匈牙利算法得到匹配矩阵如下



其中T0的一号机移动到T1的1号位，T0的二号机移动到T1的3号位，T0的三号机移动到T1的2号位，T0的四号机移动大T1的5号位，以此类推得到匹配关系。

### （1）实验数据

当按照匈牙利算法进行分配运算，所得最优匹配为：1 3 2 5 4 9 10 7 6 8 ；

对应的代价分别为为:1.0000 2.0000 1.0000 1.4142 0 4.0000 4.0000 2.0000 0 2.2361；

此时匈牙利算法的均方差为2.4742，平均代价为2.2063，均大于最优分配的均方差0.3085，平均代价2.7553。

### （2）路径对比

最优分配移动路径与匈牙利算法移动路径如下图所示：

图10.1：最优分配移动路径



图10.2：匈牙利算法移动路径

### （2）实验分析

通过matlab画出匈牙利算法与优化后的算法机群移动路径，进行对比发现：

匈牙利算法中机群移动距离长短相差较大，导致无人机飞至目标点的时间相差较大。经过优化后，各无人机飞至目标点的路径长度将趋于一致，飞行时间也将趋于一致。由此可以看出系统运行的一致性得到了优化。

## **减少一架无人机**

### 问题描述

机组中个别无人机由于移动距离较其他的无人机大，在电量不足情况下退出，编队无人机数目发生变化，无人机的运行结果如何值得我们考虑。

### 解决思路

当有一架无人机在飞行中出故障时，缺少一架无人机，故有一组匹配不能完成需要去除。由于在本实验对象中“F”和“Z”的无人机组合相对较完整且密集，故基本上去除任何一组匹配均可完成最后的图像“Z”的展示，因此，在本设计中采用去除原匹配结果中飞行代价最大的一组匹配以使能耗最小。

本设计中为了达到中途丢失一架无人机的效果，将筛选出的匹配组对应的无人机的地址分配至坐标图中最右下方的位置，即掉落的无人机最后飞行至右下角。

在本实验中，由于T0中三号机最终匹配结果移动距离最长，代价最大，故通过套用前述仿真模型以及算法进行实验，“剔除”代价最大的无人机，使其降落在固定位置。

## **实验收获**

通过这几周的无人机课程学习，我们受益匪浅。忙碌了半个学期，仿真水平及matlab编程都有所提高。

首先是增强了我们自身matlab编程与仿真水平，matlab在以后的学习中使用频率非常高，学习起来可谓是大有裨益。其次，在无人机控制技术课程中也学到了大量无人机相关知识，无人机是当今研究热点，这次课程也算是让自己深入的学习了一番。程序设计方面，我们合作设计了这么长的语言程序，经过不断的尝试、修改最后直至完成，感觉十分有成就感，在进行讲解的过程中，梳理自己的思路，对集群无人机的队形重构编程思想进行融合，对于仿真其他实验也很有帮助。

在这里我们感谢老师的指导，感谢同学们的帮助，能够学习这门课我觉得非常庆幸，真正的学到了知识，增强了技能，但愿它能够帮助到更多的人。

## **参考文献**

1. 张思宇. 多无人机协同航迹规划及其控制方法研究[D].北京理工大学,2016.
2. 卢燕梅,宗群,张秀云,鲁瀚辰,张睿隆.集群无人机队形重构及虚拟仿真验证[J].航空学报,2020,41(04):248-259.
3. 毛琼,李小民,王正军.基于规则的无人机编队队形构建与重构控制方法[J].系统工程与电子技术,2019,41(05):1118-1126.
4. 赵凌楚.多旋翼无人机编队队形变换算法研究[J].信息通信,2020(03):57-59.
5. 林倩玉. 多无人机协同编队控制算法研究[D].哈尔滨工业大学,2018.
6. 吴宇,梁天骄.基于改进一致性算法的无人机编队控制[J/OL].航空学报:1-18[2020-06-26].http://kns.cnki.net/kcms/detail/11.1929.V.20200521.1555.012.html.
7. 马培蓓,雷明,纪军.基于一致性的多无人机协同编队设计[J].战术导弹技术,2017(02):86-90.
8. 陈杰敏,吴发林,耿澄浩,徐珊.四旋翼无人机一致性编队飞行控制方法[J].航空兵器,2017(06):25-31.
9. 郑玉会. 多无人机任务分配算法研究[D].西安电子科技大学,2018.

[10]韩亮,任章,董希旺,李清东.多无人机协同控制方法及应用研究[J].导航定位与授时,2018,5(04):1-7.

[11]Weiran Yao,Naiming Qi,Neng Wan,Yongbei Liu. An iterative strategy for task assignment and path planning of distributed multiple unmanned aerial vehicles[J]. Elsevier Masson SAS,2019,86.

[12]Niaona Zhang,Xiaofang Zhang. Application of multi-agent consistency analysis based on finite state machine in driverless vehicles formation control[J]. Elsevier Ltd,2018,51(31).

[13]Jun ZHANG,Jiahao XING. Cooperative task assignment of multi-UAV system[J]. Elsevier Ltd,2020.

[14]https://blog.csdn.net/qq\_33829154/article/details/62425921