



**北京航空航天大学**  
B E I H A N G U N I V E R S I T Y

**第二十九届“冯如杯”学生学术科技作品竞赛  
项目论文**

**多基站式空域管理系统**

——面向用户终端的快递外卖投送系统

## 摘要

面对外卖、快递市场不断膨胀、无人机运送技术走入现实的今天，如何设计更加高效可靠安全的空域管理系统，使得更多无人机能够同时进行远距离投送，成为亟待解决的问题。本系统设计了一种基于多基站式的独立运行、相互配合的工作模式，利用微信小程序和 Qt 软件，提供直观方便的人机接口，能够针对长距离多机寻路的现实问题进行实时动态调度、监控显示，通过几个独立子系统配合，实现空域管理的功能。

在单基站的调度方面，提出一种算力消耗显著低于传统的 A\* 等算法的 Inself 寻路算法，能够快速、低算力地得到一条非离散坐标的路径集合；同时基于此快速寻路算法，对碰撞点冲突进行重新寻路，根据路况信息和其他无人机的路径时间信息，实时动态地调整无人机路径。

而在多基站配合方面，采用部署 django 和 MySQL 数据库的方式进行多个独立运行系统之间的异步通信，实现了无人机从一个基站过渡到另一个基站。同时在创建无人机时，利用 Inself 算法的快速性，在总控台中得到一条静态地图下的路径及其经过各个基站的出入口集合，方便后续无人机的进入。

在人机接口交互方面，针对用户，去 app 化而采用微信小程序提供交互界面，通过部署的 django 实现数据与 MySQL 之间通信。针对监管人员，提供各个子基站独立运行的界面，能够监视本基站内的所有无人机以及实时监控、更改路况信息进行调度。

通过以上几个方面，本设计以多基站式的思想完成了长距离下的多机协同规划调度，避免了过大地图下重新寻路的大量计算，转而使用基站内出入口之间的调度调整路径，完成了空域管理调度系统。

**关键字：**空域管理，路径规划，实时调度

## Abstract

In the face of the expanding take-out and express market and the reality of unmanned aerial vehicle delivery technology, how to design a more efficient, reliable and safe airspace management system to enable more unmanned aerial vehicles to deliver at a long distance at the same time has become an urgent problem to be solved.

In terms of the scheduling of single base station, an Inself pathfinding algorithm with significantly lower computational force consumption than the traditional A\* algorithm is proposed, which can quickly and cheaply obtain A path set of non-discrete coordinates. At the same time, based on the fast pathfinding algorithm, the collision point conflict is re-pathfinding, and the aircraft path is dynamically adjusted in real time according to the road condition information and other aircraft path time information.

In the aspect of cooperation of multiple base stations, django and MySQL database are deployed to conduct asynchronous communication between multiple independent operating systems to realize the transition from one base station to another. At the same time, when creating the aircraft, the rapidity of Inself algorithm is utilized to obtain a path under the static map and the set of entrances and exits passing through each base station in the general control console.

In terms of human-machine interface interaction, for users, the WeChat small program is adopted to provide the interactive interface for de-app, and the deployed django realizes the communication between data and MySQL. For regulators, it provides an interface for each sub-base station to operate independently, which can monitor all aircraft in the base station and change condition information in real time.

Through the above aspects, this design completes the multi-machine coordinated planning and scheduling under the long distance with the idea of multi-base station to complete the airspace management scheduling system.

### **Keywords**

**Airspace management, Path planning, Real-time scheduling**

# 目录

摘要.....	i
Abstract.....	ii
第一章 项目简介.....	1
1.1 项目背景.....	1
1.2 项目制作的目的是与意义.....	2
1.3 项目创新点.....	2
第二章 系统方案选择与论证.....	3
2.1 总体设计.....	3
2.2 技术方案.....	4
第三章 单基站调度设计.....	5
3.1 总体设计.....	5
3.2 路径规划.....	5
3.2.1 栅格化地图寻路算法.....	6
3.2.2 回归模拟地图路径.....	8
3.3 多机协同调度.....	10
第四章 多基站协同管理设计.....	11
4.1 模型设计.....	11
4.2 模块实现.....	12
4.2.1 外部无人机静态路径规划.....	12
4.2.2 内部无人机子基站间调度.....	13
第五章 人机交互系统设计.....	14
5.1 基站管理交互.....	14
5.2 用户服务交互.....	15
第六章 应用前景.....	16
结论.....	18
参考文献.....	19

# 第一章 项目简介

## 1.1 项目背景

当今市场外卖、快递业务火热，作为在校大学生对其感触尤其深。快递方面，每当各大电商平台在双十一进行促销引发抢购狂潮时，学校附近的快递点往往会出现前所未有的壮观景象。过于庞大的订单量导致货物数量导致物流行业超负荷运转。尽管很多时候货物可以及时送达快递点，但“超长”的队伍还是令很多消费者难以及时拿到自己的购买的物品。另一方面，电商平台为了追求更快的配送速度，过大的订单量导致的超载也会引发一系列安全问题。而对于很多交通不发达的偏远地区，及时安全的配送服务更是难以实现。

与此相比，外卖行业的配送服务则更偏重于“短距离”和“实时性”。正常来说为了保证食品的质量，大部分的外卖订单需要在半个小时左右送到用户手中。很多时候外卖配送员为了抢时间、抢订单量会同时接多份订单，这势必会影响配送的速度和配送员的人身安全，大部分配送员为了方便配送都会选择电动车、摩托车等方式出行，道路上各种配送员的危险驾驶行为更是屡见不鲜。

目前除了京东在偏远地区的配送点之间采用了无人机送快递的方式之外，现在大多数地区的外卖和快递配送业务都是以人力配以一些简要的交通工具为主。相比于传统的物流，无人机能够极大地缩短配送时间，且不受地形限制。更重要的是无人机不需要耗费过多的人力，也不需要人来操作，可以节省很大一部分人力，对于外卖行业来说更是可以降低配送员遭遇交通事故的风险。

不过目前看来，在国内由于对无人机还没有一套比较成型的规范行业管制严格，想要实现无人机快递时代还需要一定的时间。相比而言，其它一些发达国家对无人机的管理已经形成比较完备的体系，在亚洲，新加坡邮政、日本邮政、韩国邮政都已经开展通过无人机实现物流投递包裹的研发和实验；在欧美国家，美国亚马逊无人机物流计划已经研发到第九代，由此看来无人机进行物流配送已经日益成为可能。

为了在无人机物流时代的对庞大数量的无人机进行高效合理的管控，使得无人机能够智能地选择高效安全的路线，本文提出一种多基站空域管理算法，主要针对无人机物流的“障碍多”、“数量大”的特点，并依据此算法实现了可视化平台，可以通过

微信小程序在不同位置添加无人机模型，由系统自行规划最优路线，为未来无人机平台管理提供一种高效合理的解决方案。

## 1.2 项目制作的目的与意义

为了方便长距离的运输和空域管理，我们设计了一种空域管理系统，面向用户和管理人员分别设计了交互界面，通过多基站式的分立调度与总控台的交互，实现了长距离多基站式的“接力棒”式管理，同时每个基站能够对当前基站内的路况实时更新、实时调整每架无人机的路径，实现动态调度规划。

本项目的意义在于为已经初现端倪的无人机运送、无人机快递外卖的明天提供更加可靠、安全的统一管理系统，所有运输的无人机在进入基站后将控制权交给基站进行实时调度、并采用低时空复杂度的寻路算法，从而降低了单机的算法复杂度、无人机的功耗，同时也使多机同时运行的空域更加安全、可控性大大提高。同时本系统面向用户和管理者提供不同的交互，方便用户进行下单运送快递、外卖等城际速递服务，而管理者可以通过基站更新路况信息以及实时调度，即便有拥堵或突变的天气情况时，也能够智能地减少新无人机的进入，相比于每架无人机独立运行，智能化和安全性、效率均大大提高。

## 1.3 项目创新点

首先是路径规划方面，采用启发式算法与视线引导法、最大三角形法想结合，提出了 Inself 算法进行规划寻路，在栅格化地图中，相较于传统的 BFS、Dijkstra、A\* 算法，时空开支大大减少，减轻了无人机的工作负担，使得系统运行效率和落实性大大提高。

其次是多基站式的空域管理，对于多机的长距离寻路，采用多基站式的分割与总控台相互配合，通过 django 以及 MySQL 作为异步通信的媒质，提高基站的独立性和可控性，动态规划调度无人机寻路，防止因拥挤、局部天气变化等因素造成交通瘫痪等现象。

最后是方便使用，用户可以方便快捷通过微信小程序进行“下单”，召唤创建无人机进行运送，服从途中各个基站的调度，完成运送。同时也方便管理人员通过基站运行界面查看、控制各个无人机以及区域路况信息。

总的来说，目前的空域管理一般没有采用多基站式的调度策略，本系统采用多基站式独立运行，同时对所有进入区域的无人机要求控制权，否则不予进入。同时本作采用的快速寻路算法，在大规模地图下性能非常明显优于传统的 A\* 等算法，能够提供移动前的第一次静态路径快速规划，从而得到途径各个基站的出入口信息，这是很多采用传统寻路算法难以做到的。本作采用的移动是非栅格化移动，不同于棋盘类的移动策略，无人机可以不沿网格自由移动。

## 第二章 系统方案选择与论证

### 2.1 总体设计

在快递或外卖投送系统中，一般是由客户下单，商家揽单后进行投送，在这个过程中，商家和用户需要输入起终点坐标、查看运行状态即可，而对一个管理系统而言，往往需要对不同区域进行监管、动态追踪、更新等，每个区域需要能够独立运行在不同的机器上，因此每个区域还应能够对本区域内所有的飞行物进行准入、准出以及动态调度、独立运行等功能。

在多基站式空域管理系统中，我们设想空域管理都是由若干个基站式的管理平台拼接而成，对于长距离的寻路，需要途中几个区域之间的配合，将无人机一站接一站地调度指挥到终点，这个过程就像一个针对每架无人机实时的交通信号指挥过程。而指挥必然需要两个方面，一方面是路径规划、一方面是统筹调度，多基站式的优点之一就是可以将统筹调度的范围缩小，每个基站只管理本区域内的飞行物和路况信息即可。而针对用户而言，人机接口就显得非常必要，需要提供更加便捷的交互方式、对于管理者来说，本区域的动态也需要能够实时的监控。

为此，我们设计以下的系统流程：

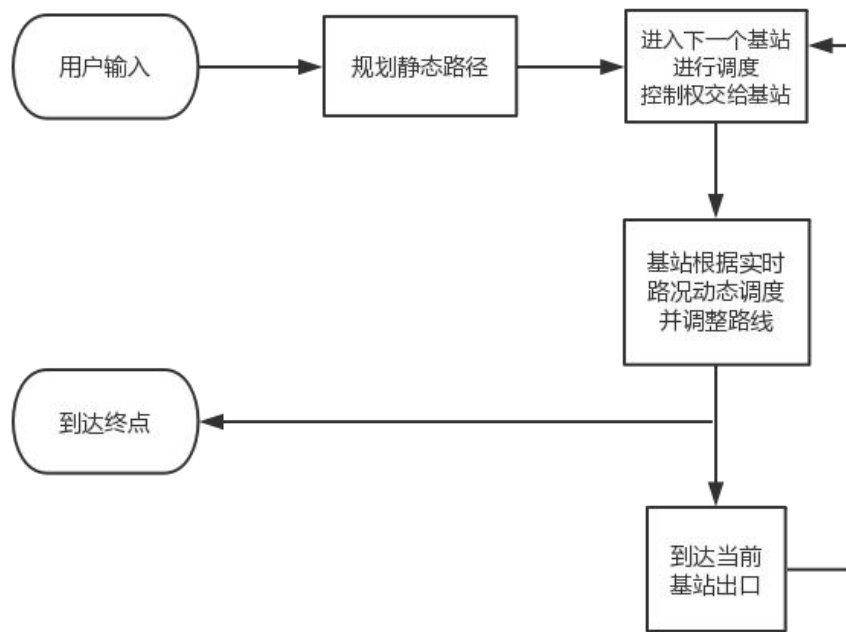


图 1 系统设计流程框图

为此，我们需要实现三方面可以独立运行的子系统：

- 用户接口子系统：实现用户的输入，创建新无人机并进行后续的分配、执行任务。
- 总控台管理子系统：对用户输入进行静态路径规划，并分配途径的子基站进出口点，告知无人机逐个通过，以及完成当无人机从一个基站退出时对下一个基站的通知。
- 子基站系统：对每一个携带当前基站出入口信息的无人机进行统一调度，针对当前基站内的无人机、实时天气路况，调整无人机的路线。

## 2.2 技术方案

由流程图分析可知，三个子系统相互独立运行却又需要相互配合，完成无人机的动态规划调度，同时管理基站使其能够响应多用户的输入、实时管理调度。

因此，三个子系统的需要进行异步通信，而数据库是一个很好的通信媒介，经过考虑和调研，我们决定采用数据库的方式，让三个子系统相互配合。而用户的输入，我们采用微信小程序的方式进行输入，更加方便快捷。



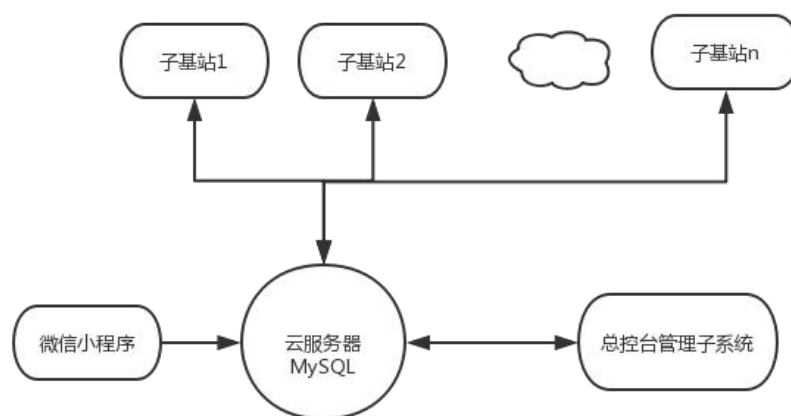


图 2 异步通信介质

于是我们租用阿里云服务器，搭建 django 网站以及 MySQL 数据库，作为微信小程序的响应以及多基站、后台子系统之间的通信载体。无人机的起终点、每个基站需要进入的无人机信息以及全局的路径规划、途径各基站的出入口坐标，都通过 MySQL 进行传递。基站只需对进入的无人机进行调度寻路、实时规划，而用户端只需输入信息，后台管理系统只需要接受到用户的输入并利用静态地图规划好途径基站出入口后，逐个通过 MySQL 告知基站即可。

## 第三章 单基站调度设计

### 3.1 总体设计

单基站的调度和规划是整个系统的核心功能，也是算法应用、技术的难点所在。单基站的功能无非是对每一个进入的无人机进行规划路线、调度前进，而多机时保证各个无人机之间不能相撞、对地图信息进行实时更新，如某个地区发生飓风、拥堵时，后续的无人机需要更换路径等，而在这个过程中，需要经常地进行路径规划，因此如果地图规模较大时，路径规划的快慢决定了系统是否稳定可靠、性能优越。

### 3.2 路径规划

在路径规划方面，首先要明确我们的问题是规划一条可行的、适用于非离散点的模拟路径，而计算机处理栅格化地图速度优势更明显，故采用先栅格化规划、再恢复

优化为非离散点的模拟路径的方案。而对于无人机而言，算力和电量资源非常紧张，对于路径规划，时空开支的要求比较高，需要优化成更加低功耗、适用于移动端的算法。为此，我们提出了一种速度优势非常明显，在大规模地图下时空开支碾压传统算法的规划算法，以下称为 Inself 算法，共分为三级实现，前两级实现栅格化地图规划，第三级实现回归到模拟路径。经过大量测试，在  $600 \times 600$  的二维地图下，在  $600 \times 600$  地图下，Inself 的平均时间为 A\* 的 8.26%、BFS 算法的 0.025%，开拓点数为 A\* 算法的 69.18%、BFS 算法的 1.11%，而路径长度平均增长为最短路径的 0.26%。在低牺牲路径长度情景下，有效减少了运算时空开支。

### 3.2.1 栅格化地图寻路算法

为了加快寻路规划，我们采用栅格化地图的方案，对模拟地图采用栅格化后进行规划，而栅格化地图常见算法有很多，如广度优先搜索衍生的 Dijkstra、A\* 算法等非常经典的算法，但这些算法在大地图下时空消耗较大，故采用启发式深搜+视线导引法作为优化进行栅格化的规划，作为 Inself 算法的前两级算法：

#### a) 第一级：启发式深搜

借鉴 A\* 的思路，将启发式算法思想与 BFS 相结合，能够更快地趋向于最终的最短路径开拓元素，从而减少开拓点数。若在深度优先搜索中加入启发函数项，可以利用 DFS 的快速省空间的优势同时，利用启发式一定程度上弥补其长度性能的劣势，让启发的道路能够趋向目标、提高路径质量。

算法思路为：仍采用一个栈结构和 closed 表进行深搜，但同时每个结点具备启发函数值 F。对于每个探索点而言，探索过程为计算/更新不在 closed 表中的临结点 G 值以及父结点信息，并以 F 值最小点入栈；若为不存在，则退栈一次；再将探索过的结点填入 closed 表。每轮结束后再以栈顶元素作为探索点重复上述过程，直至栈为空或探索至终点。若存在路径，则为自终点逆向遍历父结点至起点的所得路径。

在启发式深搜中，需要维护的最小堆大小是恒定的（二维情况下  $\leq 8$ ，三维情况下  $\leq 26$ ），为周边相邻可到达且未在 closed 表中的点，数量较小；移动到新位置后堆元素全部更新，故不存在 A\* 等算法中堆不断扩展从而耗时较长的问题，且由于能够直奔终点，减少了很多额外探索，进一步节省运算空间。

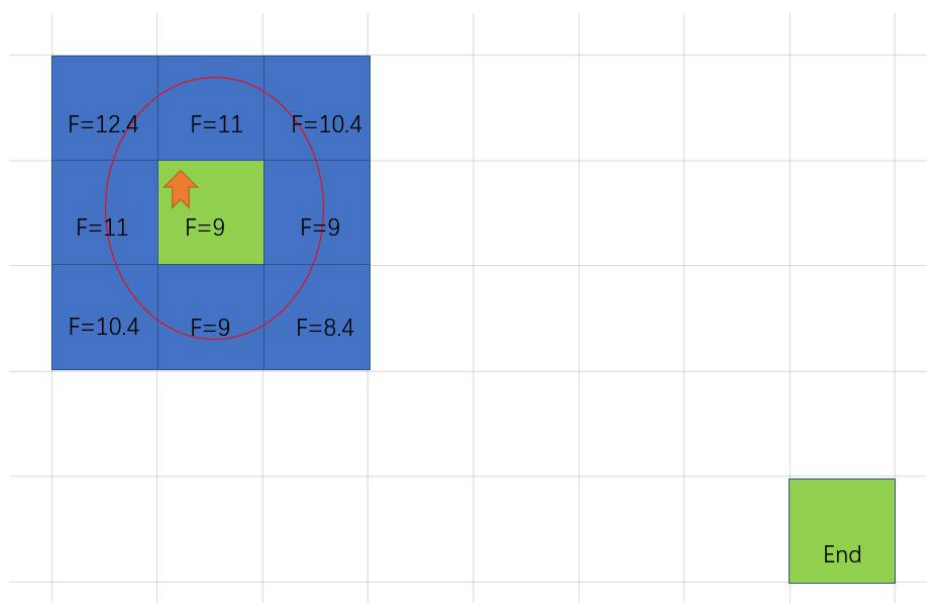


图 3 Inself 第一级算法示意图 (1)

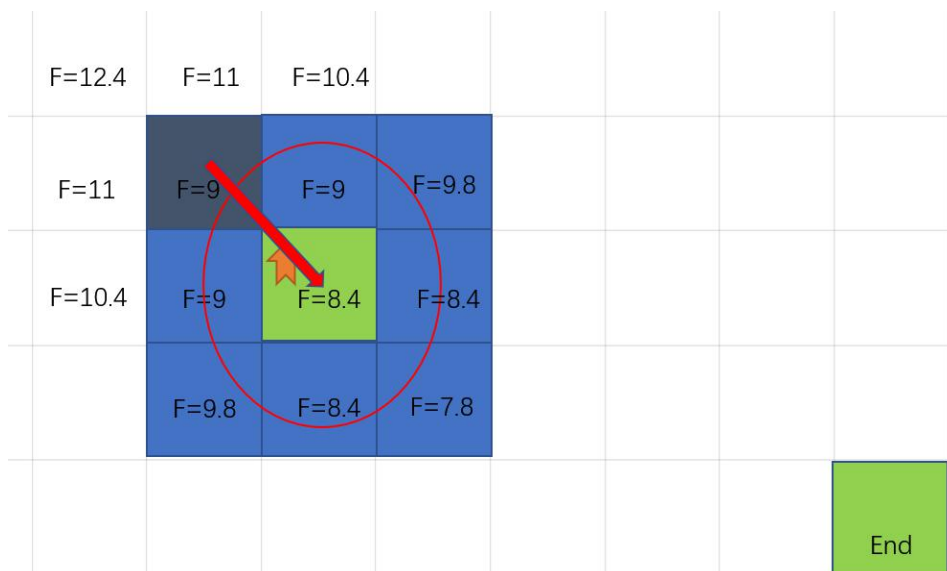


图 4 Inself 第一级算法示意图 (2)

#### b) 第二级：视线导引法

在启发式深搜后，虽然能够很快地得到一条较好的路径，但由于死路式障碍的遮挡，往往存在一些绕远、甚至是死路遍历的情况（如下图），经过测试发现，这是由于在 F 值计算结果相同时，邻点选择顺序不同会导致面对此类障碍时的反应不同。不管哪一种，都会比直接绕过障碍牺牲路径性能，因此，对路径进行适当的二次优化显得十分必要。

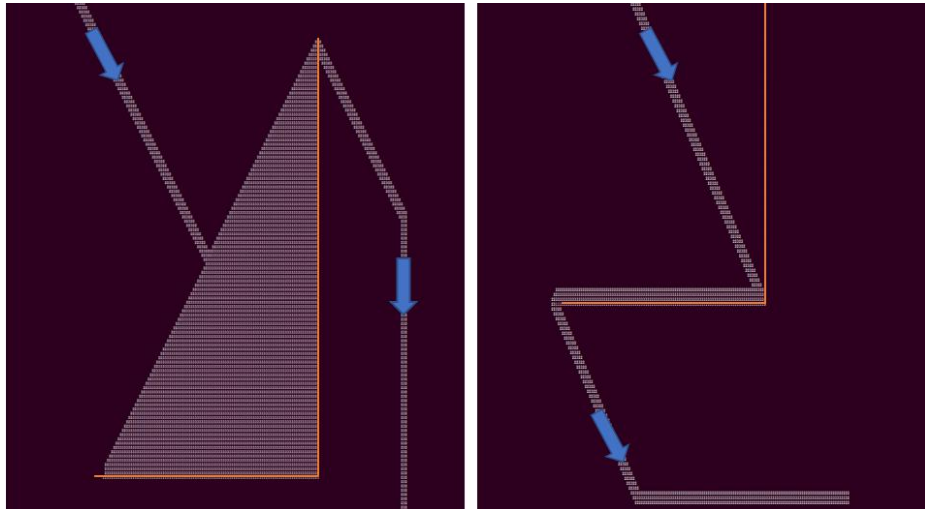


图 5 Inself 第二级算法要解决的问题

而路径再优化的方案也有很多，譬如多边形填充法、图形切割等等，但出于速度和空间的考量，我们采取自遍历的方式和视线导引法对路径绕远的问题进行优化，思路如下：

利用嵌套的两级 for 循环遍历路径链表结点，记外循环结点为  $i$ ，内循环结点为  $j$ ，若  $i$  和  $j$  在栅格地图中能够以直线连接且  $i$  和  $j$  直接没有障碍物，则将  $i$  和  $j$  直接的结点都删掉，并将其子结点、父结点填充入  $i$  和  $j$  之间线段的结点，从而删去了绕圈的路，优化路径。这种连直线优化的方式仅当  $i$  和  $j$  能够互相“看到”时才能够相连使用，故称为“视线导引法”，其时间复杂度为  $O(n^2)$ ， $n$  为第一级路径结点数，通过遍历的方式寻找其中可能存在的捷径，把非全局最优的解优化成为更好的解。

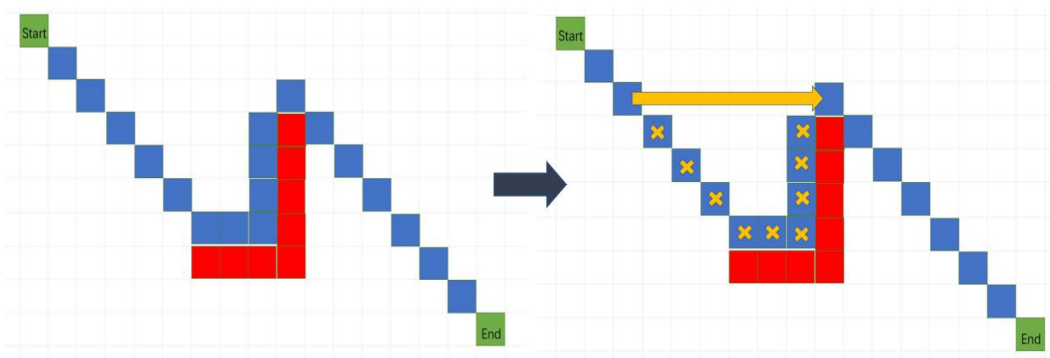


图 6 Inself 第二级算法效果

### 3.2.2 回归模拟地图路径

模拟地图栅格化后虽然便于计算机处理，且量化大小不同，得到路径的模糊程度和计算耗费此消彼长，但存在一个很大的问题：无论量化多么精密，最终的路径只能

是直线式平移或 45 度角斜向移动构成,难以使用现实寻路的任意角度方向可移动的场景下。故为了适应模拟地图,补充 Inself 的第三级算法,以最大三角形法的思想,将栅格化得到的路径进行优化。

在优化之前,我们首先分析一下前两级算法所保障的结果:由于第二级算法是对栅格地图下直/斜通连接的结果,保证了是栅格地图下不会绕远的路径,即路径中任意两点不可能不经路径而直连或斜连。保证这一点后,我们发现在三维地图下,或许只需对拐点进行优化即可,即让最终路径与水平的夹角为任意值,而非只有 0、45、90 度三种情况。但为了得到真正最优的路径,障碍可能存在于栅格路径附近的任何位置,故提出最大三角形法进行优化:

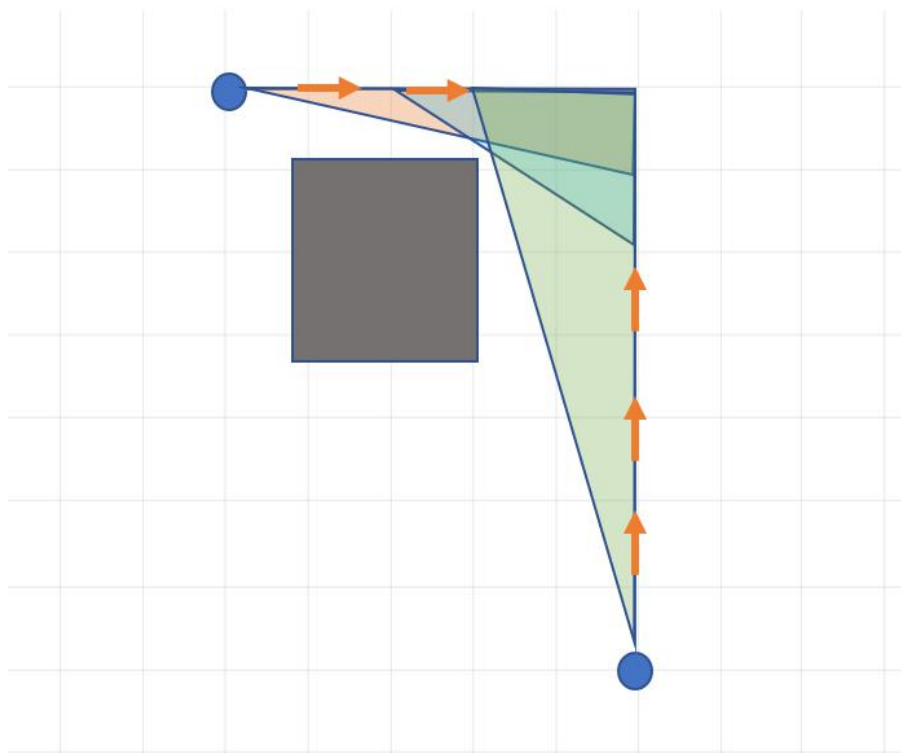


图 7 最大三角形法回归模拟地图示意图

由前两级算法可知,若要在栅格路径中得到模拟路径,则每段独立完整的优化路径必然在相邻的三个结点之内确定。故可采用最大三角形法,将优化路径的两点分别从相邻两条边进行遍历,测试能否直连,找到使得直连后三角形面积最大的边。对于整条路径而言,每相邻的三个点都经此方法进行优化,即可得到方向角度任意的基于栅格化 Inself 算法的最优模拟路径,从而回归到现实的寻路问题中,对目标的移动方向策略提供可执行的参考。

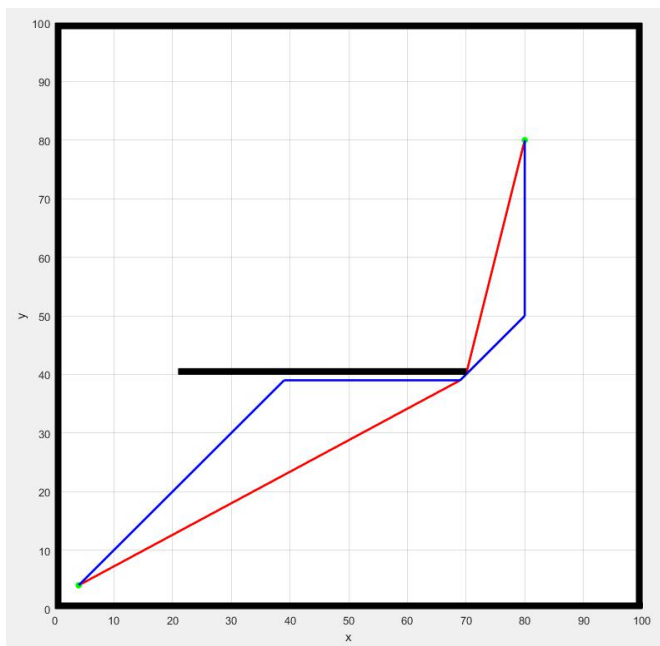


图 8 Inself 第三级算法示意图

至此，Inself 算法介绍完毕，通过 Inself，完成子基站内的路径规划，消耗的时空开支明显小于 BFS、A\* 等最短路径算法。

### 3.3 多机协同调度

在一个基站中，无人机很难自始至终都沿着既定的路线前进，在这个过程中，可能会遇到新无人机加入、可能会遇到飓风等天气情况出现需要临时绕路等等。为此，需要基站拥有所有此区域内飞行的无人机群的调度控制权，对于不能接受调度的无人机拒绝进入空域。

而管理的核心是依靠维护一个无人机栈结构：有无人机进入时首先进行一次路径规划，之后进入无人机栈，而如果某处发生飓风后，按照栈的顺序从栈底到栈顶遍历，判断每架无人机是否可能与飓风相撞，如果会，则对该架无人机从当前点到当前基站的出口重新规划路径，循此以往，直至每架无人机移动至出口后，将无人机对应的编号告知总控台，从而使无人机进入下一个基站。

而每一架无人机进入时、遇飓风需要重新寻路时，往往需要避开固定障碍物、临时障碍物和其他无人机，故采用遍历对比的方式，除了障碍物以外，对于多机协调，由于路径规划的算法速度优势非常明显，故采用规划测试、判断相撞点后循环规划的方案，直至找到一条修正后的路径。

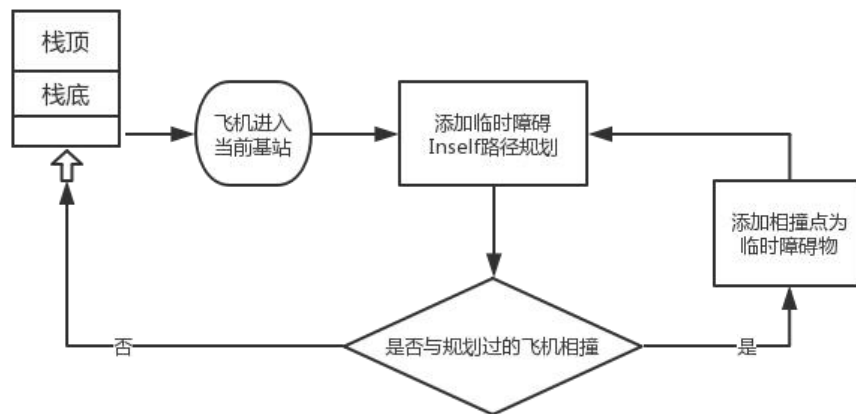


图 9 基站调度策略

## 第四章 多基站协同管理设计

### 4.1 模型设计

总控台与子基站将数据上传至数据库并从数据库读取数据以实现“通信”。示意图如下：

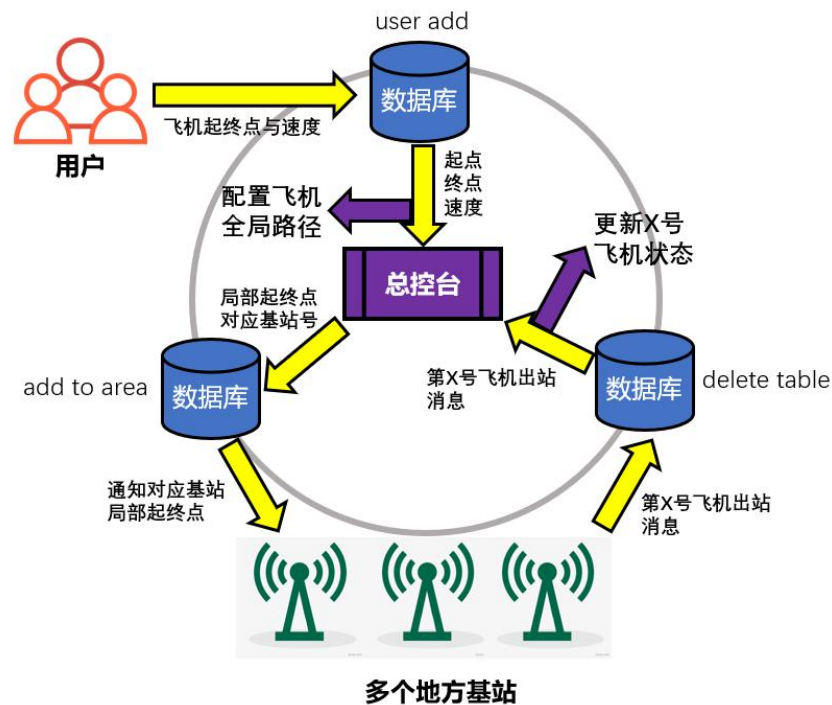


图 10 总控台控制示意图

在一个控制周期中总控台分别从 `user add`, `add to area`, `delete table` 三个数据表中读取数据, 并对读取的数据完成相关的控制操作。

其中 `user add` 中的数据由用户添加, 其中内容为飞入管控区域无人机的起终点坐标及其飞行速度。每一个控制周期中, 总控台查询该数据库, 如果存在数据, 则进行读取并对该无人机的全局路径进行配置, 得到当前无人机在每个子基站的起终点坐标。将对应的起终点坐标数据以及坐标对应的数据库序号存入 `add to area` 数据库, 等待子基站的读取。子基站读取 `add to area` 数据库中内容后针对当前子区域的特殊情况 (比如流动性障碍的出现) 对无人机的路径重新规划。

当无人机飞离子基站所管辖区域后, 子基站将该无人机编号存入 `delete table` 数据库。在每个控制周期里, 总控台将读取 `delete table` 中数据。对于其中的无人机, 在总控台保存的无人机队列信息中查询其下一站应到达的子区域, 将该区域对应的基站序号以及其起终点存入 `add to area` 数据库, 等待子基站对其读取与规划。若当前无人机已经到达终点, 则清空数据, 释放使其飞出总控台管辖区域。

## 4.2 模块实现

上一节中说明了总控台协同控制的具体框架, 这一节将说明其对应的模块实现。总控台的协同交互主要通过对三个数据库的读取和写入进行实现。而其主要实现两种的功能: 1. 对新进入的外部无人机再静态地图下进行规划路径; 2. 对内部无人机在子基站之间进行调度。

### 4.2.1 外部无人机静态路径规划

用户输入的外部无人机信息将存入 `add user` 表。总控台在每一个工作周期中会读取一次 `add user` 表。若存在新入无人机, 则首先根据其起终点信息规划整体路径。然后通过整体路径来判断会经过哪些子区域, 以及经过各子区域的起终点坐标。将这些信息存入 `add to area` 表等待子基站读取。

其程序框图如下:



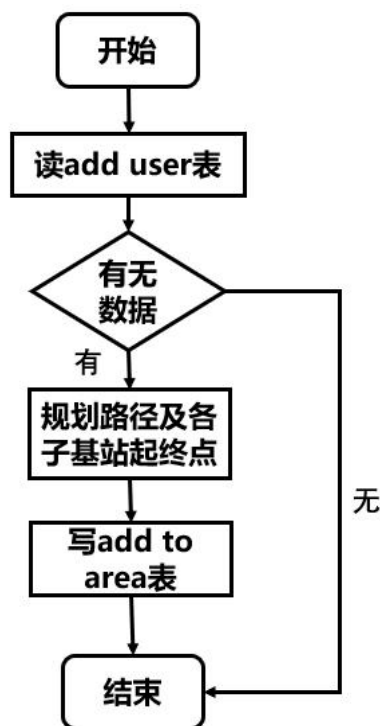


图 11 外部无人机进入区域规划流程图

#### 4.2.2 内部无人机子基站间调度

总控台需要解决无人机在总体管辖区域内，各子基站管辖区域之间的调度。对于子基站，其读取 add to area 表，当存在数据时，表示有无人机飞入该子区域。对于此无人机的起终点进行重新规划路径。当无人机飞出该子区域时，子基站向 delete table 表中存入该无人机编号。总控台在每一个工作周期中会读取 delete table 表，若存在无人机编号，则需要为这架刚刚出子基站的无人机进行调度，即“告知”它将去往的子基站它的到来。实现过程中我们利用总控台查询该无人机编号并找到最初规划时该无人机的途径基站队列。将它下一个将去基站的起终点坐标以及基站序号存入 add to area 表。

其程序框图如下：

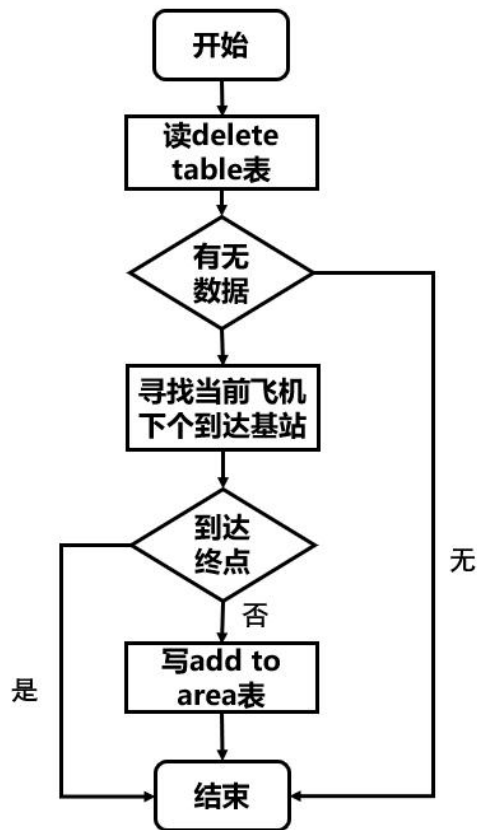


图 12 内部调度方案程序框图

## 第五章 人机交互系统设计

### 5.1 基站管理交互

为了方便管理、便于监管人员对基站进行监视，以及查看当前区域内的天气路况、实时更新，需要对每一个基站实现独立运行、查看，故每个基站都采用 QT 搭建界面运行程序，多个基站之间可以独立运行于不同电脑中，实现独立监管。

而为了进行模拟环境测试，加入了基站的人机交互功能：

- a) 右侧输入功能：手动输入基站内的新建无人机以及临时飓风区域。
- b) 鼠标左键点击：设置无人机起终点，创建新无人机并加入到基站中去。
- c) 鼠标右键点击：按照右侧输入的飓风时间和半径，创建飓风区域。

除此以外，基站右下角还对鼠标当前所在点对应的坐标进行显示，方便监管人员及测试人员加入无人机、飓风以及查看对应无人机的当前位置。

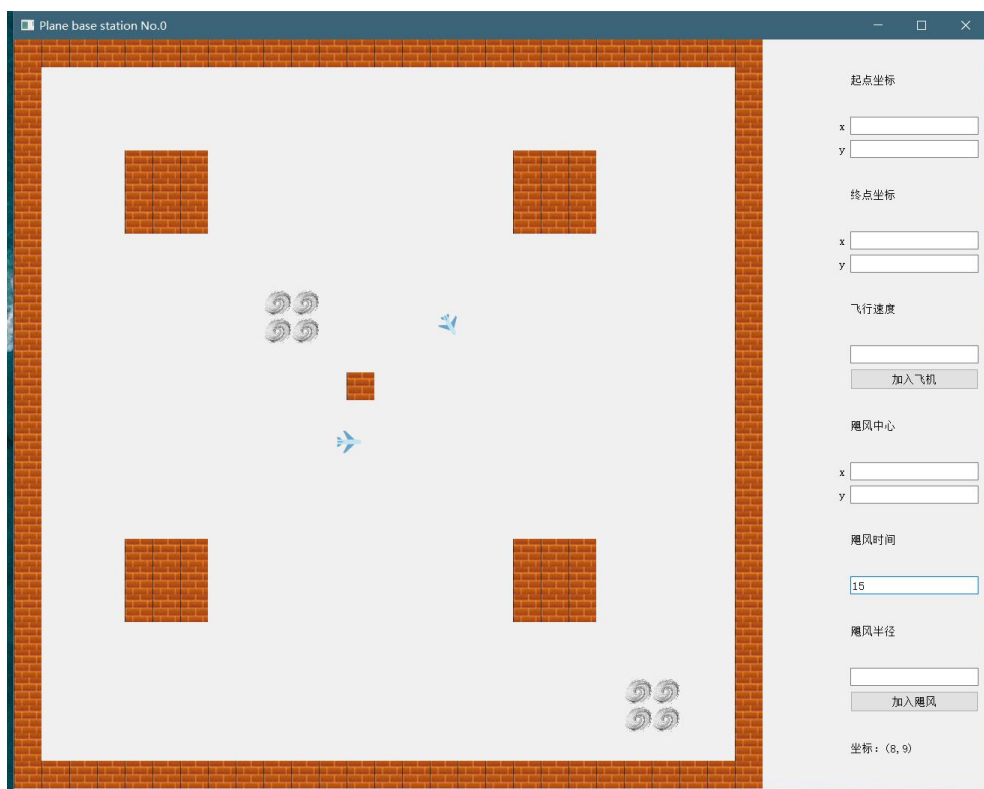


图 13 基站交互界面

## 5.2 用户服务交互

为了方便用户的输入，设计开发了微信小程序作为终端交互窗口，目前的测试版提供清空整个数据库的功能，正常的使用方式为添加全局的起终点坐标，点击创建后，即可将无人机信息添加入 MySQL 中去，通过总控台的规划路径以及逐个基站起终点交给子基站进行调度，实现动态的调度规划和空域管理。以下为操作界面截图：



图 14 用户交互界面

为了能够响应微信小程序，让输入的信息进入 MySQL 数据库中，我们利用阿里云服务器搭建了 django 网站，对小程序的 request 请求包含的信息头进行提取，将输入的信息利用 python 的 MySQL 库函数，写入到 MySQL 中，完成相应的 MySQL 操作，从而完成用户输入到数据库的系统入口部分，也提供更加便捷的人机交互模式，采用去 app 的传统移动终端开发模式，采用微信小程序平台搭建入口，便于使用。

## 第六章 应用前景

本文提出的多基站无人机调度算法不仅可以应用在无人机物流领域，其优秀的普适性甚至可以推广到民航管理等领域。无人机由于不受空间地形的限制，能够更加机动灵活、快速高效地完成配送服务，尤其是在偏远地区或者有急需物品需要配送时，合理的交接使得无人机能够 24 小时不间断地完成配送服务。

在控制端方面，该多基站空域管理系统通过使用当下流行的微信小程序完成无人机的配置，控制方法操作简单直观，仅需输入起点和终点位置，小程序将其上传到服

务器后即可由系统自动规划出最高效的飞行路线，并自动分配合适的无人机进行该路线的物流配送任务，这很大程度地降低了物流管理人员的学习成本。当体系完善之后，可进一步将这一过程自动化，进一步节省人力开支，这很好地适应了当下高度信息化、自动化的发展趋势。物流公司还可以通过提供的可视化界面，实现对无人机运行状况的实时监控和管理。消费者也可通过该可视化界面实时关注订单的配送情况，这也一定程度上增大了配送的实时性，能够更及时地向消费者反映目前的情况。

在通信方面，即将到来的 5G 技术也将为无人机提供更为高速稳定的数据传输和定位服务。精准的定位甚至能够实现物流服务中一直没能很好解决的“最后一公里”问题提供解决方案。

总之，本文提出的面向用户终端的快递外卖投送系统为未来的无人机物流提供了一种高效可行的解决方案，随着无人机和物流行业的迅猛发展，该管理系统将拥有广阔的应用前景。

## 结论

本文提出了一种面向无人机物流服务的多基站空域管理系统，能够针对长距离多机寻路问题进行实时动态调度和实时监控。针对无人机物流“障碍多”、“数量大”的特点，使用了自行设计的 Inself 寻路算法，与传统的 A\* 算法相比显著降低了算力的消耗。并基于此算法，考虑碰撞点冲突问题，实时根据路况信息和其它无人机的飞行情况对无人机的路线进行实时调整。由于单基站的覆盖范围有限，实际问题中往往涉及多基站的协同调度。针对这个问题本文提出用部署 django 和 MySQL 的方式实现异步通信，从而达到不同基站覆盖范围的平稳过渡。

为了方便控制端进行无人机起点终点信息的输入，本文提出利用当下流行的微信小程序提供简单明了的交互界面，并利用 Qt 设计并实现可视化界面为物流平台的管理人员提供了各个基站覆盖范围内的无人机运行情况，方便管理人员对无人机进行实时监控和调度。

综上，本文针对物流产业的特点，提出并实现了一种高效可行的无人机物流管理雏形，具有很高的普适性和广阔的应用前景。

## 参考文献

- [1] 杨波,徐肖豪,戴福清,项恒.空域管理和评估系统中空域辅助系统的设计[J].中国民航学院学报,2004(05):5-8.
- [2] 尹佳,张学军,王伟,刘松.空域管理与评估系统的设计[J].空中交通管理,2004(01):14-16.
- [3] 罗广恒.基于 Django 和 MySQL 的网络化测试数据查询系统研究[J].智能物联技术,2019,51(02):15-21+31.