

# 承诺书

我们完全清楚，在竞赛开始后参赛队员不能以任何方式，包括电话、电子邮件、“贴吧”、QQ 群、微信群等，与队外的任何人（包括指导教师）交流、讨论与赛题有关的问题；无论主动参与讨论还是被动接收讨论信息都是严重违反竞赛纪律的行为。

我们以中国大学生名誉和诚信郑重承诺，严格遵守竞赛章程和参赛规则，以保证竞赛的公正、公平性。如有违反竞赛章程和参赛规则的行为，我们将受到严肃处理。

我们授权全国大学生数学建模竞赛组委会，可将我们的论文以任何形式进行公开展示（包括进行网上公示，在书籍、期刊和其他媒体进行正式或非正式发表等）。

我们的报名参赛队号（12 位数字全国统一编号）： 202227003120

参赛队员 (打印并签名): 1. \_\_\_\_\_

3. \_\_\_\_\_

(指导教师签名意味着对参赛队的行为和论文的真实性负责)

日期: 2020 年 08 月 22 日

(请勿改动此页内容和格式。此承诺书打印签名后作为纸质论文的封面，注意电子版论文中不得出现此页。以上内容请仔细核对，如填写错误，论文可能被取消评奖资格。)

赛区评阅编号：  
(由赛区填写)

全国评阅编号：  
(全国组委会填写)

2023 高教社杯全国大学生数学建模竞赛

编 号 专 用 页

赛区评阅记录（可供赛区评阅时使用）：

评 阅 人						
备 注						

送全国评阅统一编号：

(赛区组委会填写)

(请勿改动此页内容和格式。此编号专用页仅供赛区和全国评阅使用，参赛队打印后装订到纸质论文的第二页上。注意电子版论文中不得出现此页。)

# 无人机编队飞行中的纯方位无源定位模型研究

## 摘要

针对问题一，此时编队为圆形编队，对第一小问，在知道圆心的一架无人机和圆周上两架发射机的编号的前提下，即已知发射机位置和接收角的前提下计算无人机的位置。通过平面几何基本定理，角的关系和解三角形的相关知识列出方程组，由于方程组中存在超越方程，因此采用图形的方式求得数值解，进而计算出所需定位的无人机位置从而实现定位。

对第二小问，此时仅知 FY00 和 FY01 两架飞机，剩余飞机编号未知，即剩余发射机的位置未知。根据几何知识，不共线的三点能确定一个圆，因此表示出所有发射机与接收机所在点所形成的圆轨迹方程，求出多圆交点即可确定接收机的飞机位置。采用依次增加无人机数量的方式。当增加的无人机数量为 1 时，此时共三架发射机，结果总存在两架对称的可能接收机，因此增加 1 架无人机无法实现定位。当增加的无人机数量为 2 时，设接收机位置为 M，共 4 架发射机，共 6 种组合，舍弃掉不确定的情况有四个组合中每三点建立外接圆。由平面几何知识可知过同一点的确定的四个圆不存在第二个交点，建立出定位模型，利用最小二乘法求解。因此当增加的未知编号的飞机数为 2 时可以实现接收机的定位。但若注意两架新增的发射机不能对称，否则仍无法对目标无人机进行定位。

对第三小问，圆周上三架未知编号的发射机和 FY00 可以确定接收机的位置。本问中，对接收机定位时认为发射机是在标准圆周上的，因此定位存在偏差。在此基础上将编队均匀分布到某个圆周上，我们将除 FY00 外的无人机的极径的平均作为理想圆周的半径。对问题进行分析可得无人机的偏差定位和实际定位与 FY00 是共线的，通过几何关系求出偏差的定位，将即接收机得知的定位，同时无人机通过期望移动到标准圆周上进行移动，之后换算出移动后的实际位置，与理想圆周的对应位置计算距离，将本轮所有除 FY00 对理想圆周的半径进行求和作为目标函数。遍历所有的发射机组合共 84 种计算出本轮最优的发射机组合，同时将该趟位置作为下一次迭代的初始位置。直到收敛。求得经过仅四次调整后误差大小为  $2.37e-02$ ，前三次调度的飞机号为 1 4 7, 3 4 5, 3 6 9，理想半径为 104.63m。效果优良，收敛迅速。

对问题 2，此时编队形状变成了锥形，对无人机的调整方案与问题 1 的第三小问是相同的，但由于形状的不同，对于偏差的定位存在差异。本问中提出基于最小误差角和的偏差点定位方式。使用 matlab 进行仿真，初始间距为 50m，添加 10% 的误差扰动。尝试调度到 45m 的间距上。经过约 10 次调整收敛，此时误差为 0.9047m。得到了十分满意的效果。

**关键字：** 无人机 纯方位无源定位 最小二乘法 贪心算法

## 一、问题重述

### 1.1 问题背景

无人机是利用无线电遥控设备和自备的程序控制装置操纵的不载人飞机。近年来，无人机在各个领域得到了广泛的应用，如军事、环境监测、影视航拍 [1] 等，尤其是无人机飞行时保持无线电静默，不被敌方雷达或敌方侦察机探测到在现代国防与战争中有着十分重要的意义。而单个无人机的应用是有限的，为了完成更多的任务，可以采用无人机组，编队飞行提高工作效率。

在编队飞行的过程中，无人机的队形保持 [2] 是最为基础的部分。无人机集群在遂行编队飞行时，为避免外界干扰，应尽可能保持电磁静默，拟采用纯方位无源定位的方法调整无人机的位置，即由编队中某几架无人机发射信号、其余无人机被动接收信号，从中提取出方向信息进行定位，来调整无人机的位置。在编队飞行中，每一架无人机有固定编号，且各个无人机相对位置保持不变。

### 1.2 问题提出

**问题一：**10 架无人机形成圆形编队，其中九架无人机均匀分布在某一圆周上，另一架无人机位于圆心，且均保持在同一高度上飞行。

(1) 位于圆心的无人机和编队中另 2 架无人机发射信号，其余位置略有偏差的无人机被动接收信号。当发射信号的无人机位置无偏差且编号已知时，建立被动接收信号无人机的定位模型。

(2) 某位置略有偏差的无人机接收到编号为 FY00 和 FY01 的无人机发射的信号，另接收到编队中若干编号未知的无人机发射的信号。若发射信号的无人机位置无偏差，除 FY00 和 FY01 外，还需要几架无人机发射信号，才能实现无人机的有效定位？

(3) 按编队要求，1 架无人机位于圆心，另 9 架无人机均匀分布在半径为 100 m 的圆周上。当初始时刻无人机的位置略有偏差时，请给出合理的无人机位置调整方案。每次只能选择编号为 FY00 的无人机和圆周上最多 3 架无人机遂行发射信号，仅根据接收到的方向信息来调整无人机的位置，使得 9 架无人机最终均匀分布在某个圆周上。

**问题二：**实际飞行中，无人机集群也可以是其他编队队形，例如锥形编队队形。仍考虑纯方位无源定位的情形，设计无人机位置调整方案。

## 二、问题分析

对于问题一：

问题一让我们针对无人机编队的形状为圆形的情况进行分析和讨论。此时我们假设所有飞机的飞行高度均一致，无人机编队飞行时可将其抽象成质点进行分析与研究。同时，为了方便定义每一架飞机的位置，我们建立了极坐标系与直角坐标系两个坐标系进行下面的分析。对于每架无人机，我们均不考虑其在队形移动时因为速度，偏转角度等对实际模型带来的影响，仅考虑通过无源定位模型进行无人机的调度。

第(1)问：

要求我们在 FY00 及其他两架位置无偏差的无人机进行信号发射且接收机能够知道发射机编号的前提下对其他无人机进行位置定位。由于接收机知道发射机的信号，故可精确的根据角度及半径  $R$  进行解三角形的处理，利用接收到的三个方向角进行当前位置的定位。所需要注意的是，由于解三角形时三角函数确定正负但并不能确定其所在象限，所以我们要对其进行分类讨论得到最后的定位结果。

第(2)问：

问题一（2）要求我们求解除 FY00, FY01 以外还需要几架无人机能在发射机位置无偏差、接收机不知道发射机编号的情况下对接收机进行准确的定位。首先我们根据圆的对称性规则得出以下结论：在只增添一架飞机的时候我们会有两架相对于  $x$  轴对称的无人机不能进行准确的定位，所以开始对增加两架飞机的情况进行分析。

首先，我们考虑 FY00, FY01, A1, A2 与接收机  $M'$  可能形成的夹角组合一共有种可能，通过对方程次数以及方程组同构现象的分析最终确定了三组夹角。根据几何知识可知三点可以唯一确定一个圆，且由于三点中包含点  $M'$ ，所以可以表示出四个  $M'$  的圆轨迹方程，这四个圆的共同交点即为  $M'$ 。所以我们可以通过方程组求解出  $M'$  的坐标。但是我们考虑到如果选择关于  $x$  轴对称的两架飞机时，无法对其他的关于  $x$  轴对称的两架无人机进行准确的定位区别。所以得出的结论是再加两架飞机即可进行定位，但是所选的这两架飞机不能关于  $x$  轴对称。

第(3)问：需要考虑在除了编号为 FY00 的无人机以外其他 9 架无人机位置均有偏差的情况下选择从外面 9 架飞机中选择 3 架无人机进行定位与调整。根据题目所给数据，可以观察到其余 9 架无人机和 FY00 的距离各有不同，因此可以把 FY01~FY09 无人机的位置看作是有偏差的，且当有偏差的无人机发射信号时，待接收信号无人机会按照有偏差的方向信息进行位置调整，最终会调整到一个有偏差的位置，也称为为错误理想位置。在计算的过程中，每一次调整的依据都是通过贪心算法选择总体误差最小的发射机组合进行迭代。所选择的理想半径是在动态调度过程中根据总体误差最小这个前提进行实时调整的，最终满足无人机均匀分布且在同一个圆周上。

对于问题二：

问题二要求我们对第一问所分析出的模型进行推广，我们考虑的推广模型为锥形。为了研究方便，我们以作为直角坐标系  $x$  轴的正半轴进行分析。根据模型一，我们首先对锥形分布的前提下为了进行准确定位与调整所需要的无人机架数进行讨论与分析。我

们对标准情况下的无人机坐标给予随机值进行偏移形成我们的初始模型。并对增加三架无人机的一般定位与模型方案进行计算，要求根据无源定位使得有偏差的无人机群通过调整回到锥形队列中。目标函数与问题一 (3) 一致，所需调整的仅有约束条件，最后得出调整方案

### 三、模型假设

1. 假设无人机的位置偏差不包含高度偏差
2. 无人机体积小，相对距离远，可以将其看作质点
3. 假设在编队飞行中无人机与无人机的相对位置关系保持不变
4. 假设发射信号无人机不存在发射故障情况
5. 假设发射信号的无人机在结束发射前不会移动
6. 假设所有待接收信号的无人机的移动是并行的
7. 假设无人机在调整位置时，忽略移动时间

### 四、符号说明

符号	意义
$\rho_i$	编号为 $i$ 的无人机距离中心点的距离
$p_i^T$	编号为 $i$ 的无人机的理想位置
$p_i^{FI}$	编号为 $i$ 的无人机的计算出的错误理想位置
$p_i^R$	编号为 $i$ 的无人机实际位置
$p_i^{R'}$	编号为 $i$ 的无人机移动后的实际位置
$p_i^C$	编号为 $i$ 的无人机的计算位置
$\vec{p}_D$	无人机移动的位移差

### 五、模型的建立与求解

#### 5.1 问题一第 (1) 问的模型建立与求解

##### 5.1.1 被动接收信号无人机的定位模型

根据题目，圆形编队队形如图 1 示，因无人机高度一定，我们将其转换为二维平面，以无人机 FY00 为极点，FY01 为极轴，建立极坐标系，如图 2 所示。设  $M'(\rho', \theta')$  为位

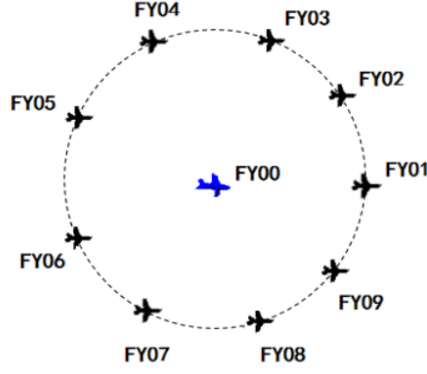


图 1 圆形无人机编队示意图

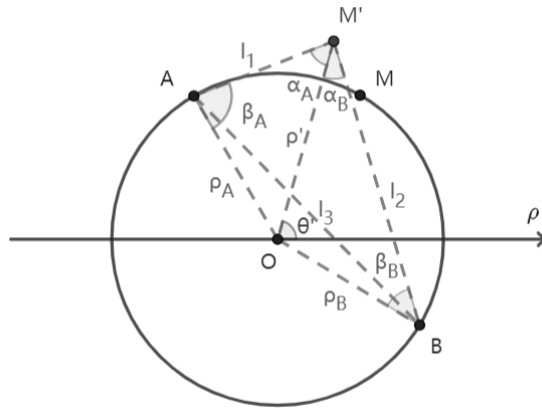


图 2 圆形无人机编队极坐标几何关系图

置有偏差的无人机，位于第一象限，其标准位置为  $M(\rho, \theta)$ ， $A(\rho_A, \theta_A)$ 、 $B(\rho_B, \theta_B)$  表示另外两架位于圆周上的无人机，位于以  $OM'$  两侧。其余分布情况与此情况类似，不做赘述。

对于  $\triangle BOM'$ ，由正弦定理得：

$$\rho_A \sin \beta_B = \rho' \sin \alpha_B \quad (1)$$

由内角和关系得：

$$\alpha_B + \min \{ \theta_B - \theta', [2\pi - (\theta_B - \theta')] \} + \beta_B = \pi \quad (2)$$

当  $\theta_B - \theta' \leq \pi$  时，公式 (2) 为  $\alpha_B + \theta_B - \theta' + \beta_B = \pi$ ；当  $\theta_B - \theta' > \pi$  时，公式 (2) 为  $\alpha_B + 2\pi - (\theta_B - \theta') + \beta_B = \pi$

对于  $\triangle AOM'$ ，由正弦定理得：

$$\rho_B \sin \beta_A = \rho' \sin \alpha_A \quad (3)$$

由内角和关系得：

$$\alpha_A + \min \{ \theta_A - \theta', [2\pi - (\theta_A - \theta')] \} + \beta_A = \pi \quad (4)$$

当  $\theta_A - \theta' \leq \pi$  时, 公式 (4) 为  $\alpha_A + \theta_A - \theta' + \beta_A = \pi$ ; 当  $\theta_A - \theta' > \pi$  时, 公式 (4) 为  $\alpha_A + 2\pi - (\theta_A - \theta') + \beta_A = \pi$

已知点 M、A、B 所代表的无人机均位于同一圆周上, 即

$$\rho_A = \rho_B = \rho \quad (5)$$

因此, 根据公式 (1) (5), 被动接收信号无人机的定位模型为

$$\begin{cases} \rho_A \sin \beta_B = \rho' \sin \alpha_B \\ \alpha_B + \min \{ \theta_B - \theta', [2\pi - (\theta_B - \theta')] \} + \beta_B = \pi \\ \rho_B \sin \beta_A = \rho' \sin \alpha_A \\ \alpha_A + \min \{ \theta_A - \theta', [2\pi - (\theta_A - \theta')] \} + \beta_A = \pi \\ \rho_A = \rho_B = \rho \end{cases} \quad (6)$$

### 5.1.2 模型求解

因为无人机 AB 的位置不固定, 其角的关系也有相应的变化, 所以需要分情况讨论  $\theta_A$  与  $\theta_B$ 。根据公式 (2) (4),  $\beta_A$  的值共有四种情况:

$$\beta_A = \begin{cases} \alpha_B + \theta_B + \beta_B - \alpha_A - \theta_A & , \theta_B - \theta' \leq \pi \text{ 且 } \theta_A - \theta' \leq \pi \\ \theta_A - (\alpha_B + \theta_B + \beta_B + \alpha_A) & , \theta_B - \theta' \leq \pi \text{ 且 } \theta_A - \theta' > \pi \\ \theta_B - (\alpha_B + \theta_A + \beta_B + \alpha_A) & , \theta_B - \theta' > \pi \text{ 且 } \theta_A - \theta' \leq \pi \\ \theta_A - \theta_B + \alpha_B + \beta_B - \alpha_A & , \theta_B - \theta' > \pi \text{ 且 } \theta_A - \theta' > \pi \end{cases} \quad (7)$$

由公式 (1) (2) (5) 得:

$$\sin \beta_A = \frac{\sin \alpha_A}{\sin \alpha_B} \sin \beta_B \quad (8)$$

由反三角函数解得:

$$\beta_A = \arcsin \frac{\sin \alpha_A}{\sin \alpha_B} \sin \beta_B \quad (9)$$

易知  $\beta_A \in (0, \pi)$ ,  $\beta_B \in (0, \pi)$  由公式 (7) (9) 可求得  $\beta_B$ , 但显然没有解析解, 可以通过曲线相交求得数值解

由公式 (1) (2) (5) 得:

$$\begin{aligned} \rho' &= \rho \frac{\sin \beta_B}{\sin \alpha_B} \\ \theta' &= \begin{cases} \pi - \alpha_B + \theta_B - \beta_B & , 0 < \theta' - \theta_B \leq \pi \\ \theta_B - \alpha_B - \beta_B - \pi & , \pi < \theta' - \theta_B \leq 2\pi \end{cases} \end{aligned} \quad (10)$$

此为待接收信号无人机的位置极坐标。



## 5.2 问题一第 (2) 问的模型建立与求解

### 5.2.1 模型建立

首先考虑还需要一架发射信号的无人机情况:

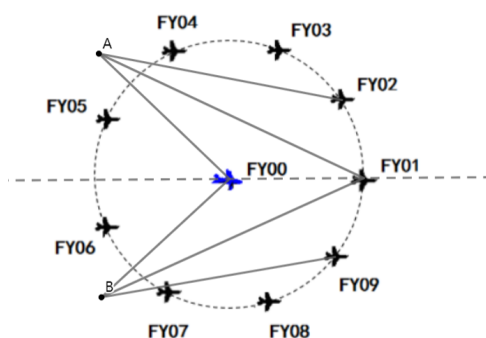


图 3 基于三架无人机发射信号定位图

如图图 3 所示, 假设待接收信号无人机位于 A 点, 发射信号飞机已知为 FY00, FY01, 因不知道第三架发射信号飞机的编号, 所以对于待接收信号飞机来说, 存在 B 点使无人机接收到的方位信息与在 A 点时相同, 因此待接收信号无人机并不能确定它自身是在 A 点还是 B 点。因此此情况不成立。

然后考虑还需要两架发射信号的无人机情况:

1. 当两架无人机关于 FY00 和 FY01 所在直线对称时, 如图 4 所示, 假设待接收信号无人机位于 A 点, 发射信号飞机已知为 FY00, FY01, 若第三、四架无人机编号为 FY02, FY09 或 FY03, FY08 或 FY04, FY07 或 FY05, FY06, 对于待接收信号无人机来说, 与上述情况相同, A, B 点无人机接收到的方位信息相同, 无法进行有效定位, 此情况不成立。

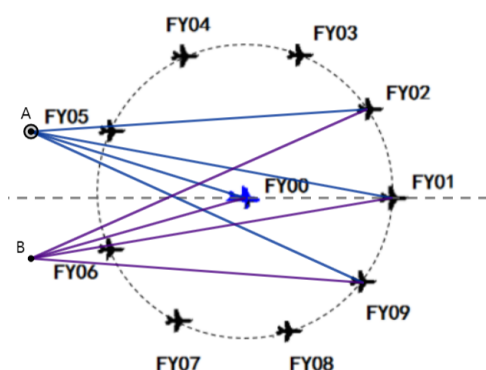


图 4 基于四架无人机发射信号定位图

2. 当两架无人机关于 FY00 和 FY01 所在直线不对称时:

以无人机 FY00 为原点，原点到 FY01 为 X 轴正方向，建立直角坐标系。设  $M'(x, y)$  为待接收信号的无人机， $A_1A_2$  为新增加发射信号无人机，无人机 FY00 与 FY01 的距离为  $R$ ，则无人机的坐标为  $FY00(0, 0)$ ， $FY01(R, 0)$ ， $A_1(R\cos(k_1\alpha), R\sin(k_1\alpha))$ ， $A_2(R\cos(k_2\alpha), R\sin(k_2\alpha))$ ，其中  $k_1, k_2$  为常数，且  $k_1 \neq -k_2, \alpha = \frac{2}{9}\pi$ 。根据几何知识，不在同一直线的三点能确定一个圆，因此表示出所有信号发射飞机与  $M'$  点所形成的圆轨迹方程，求出多圆交点即可确定待接收信号的飞机位置。

圆中三点选取有六种组合，分别为：(1) $FY00, FY01, M'$ ；(2) $FY00, A_1, M'$ ；(3) $FY00, A_2, M'$ ；(4) $FY01, A_1, M'$ ；(5) $FY01, A_2, M'$ ；(6) $A_1, A_2, M'$ 。其中，组合 (6) 的圆方程式中会同时包含  $k_1, k_2$  两个未知数，无法求解，故不考虑此组合；组合 (2)(3) 的圆方程式所差参数由  $A_1A_2$  决定，会出现同构情况。因此只考虑 (1)(3)(4)(5) 组合，如图 5 所示：

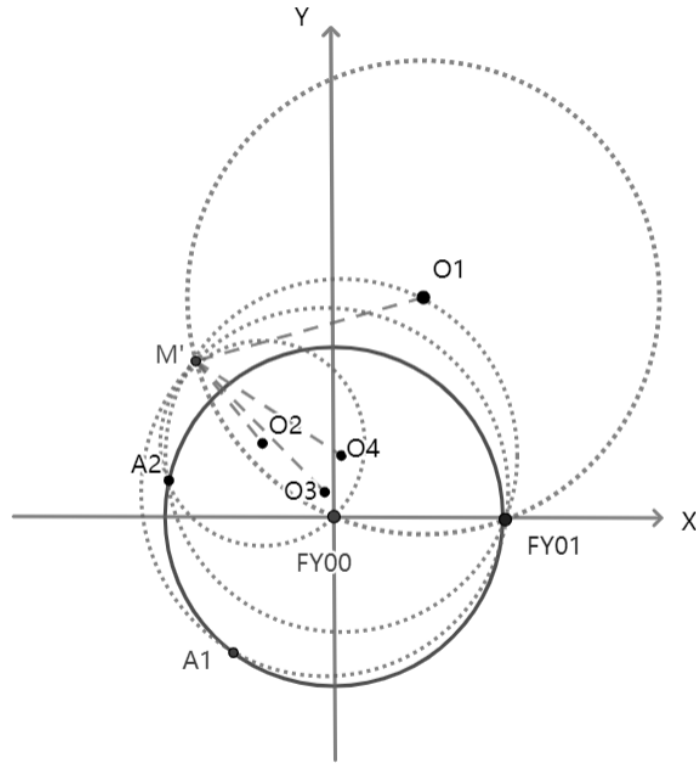


图 5 无人机定位模型示意图

设圆方程的圆心为  $O_i(x_i, y_i)$ ，半径为  $r_i$ ， $\alpha_i$  为  $M'$  接收到其余两架无人机发送信号的方位信息角， $i = 1, 2, 3, 4$  分别对应组合 (1)(3)(4)(5) 所构建的圆。其方程为：

$$(x - x_i)^2 + (y - y_i)^2 = r_i^2, i = 1, 2, 3, 4 \quad (11)$$

则圆  $O_1$  必过  $FY00(0, 0)$ ， $FY01(R, 0)$  两点。且根据圆上弦所对的圆周角度数相等及垂径

定理可得方程组：

$$\begin{cases} x_1^2 + y_1^2 = r_1^2 \\ (x_1 - R)^2 + y_1^2 = r_1^2 \\ R^2 = 4r_1 \sin \alpha_1 \end{cases}$$

同理可得到其余圆方程组：

$$\begin{cases} x_2^2 + y_2^2 = r_2^2 \\ (x_2 - R \cos k_2 \alpha)^2 + (y_2 - R \sin k_2 \alpha)^2 = r_2^2 \\ R_1^2 = 4r_2 \sin \alpha_2 \end{cases}$$

$$\begin{cases} (x_3 - R)^2 + y_3^2 = r_3^2 \\ (x_3 - R \cos k_1 \alpha)^2 + (y_3 - R \sin k_1 \alpha)^2 = r_3^2 \\ (R - R \cos k_1 \alpha)^2 + (1 - R \sin k_1 \alpha)^2 = 2r_3^2 \sin^2 \alpha_3 \end{cases}$$

$$\begin{cases} (x_4 - R)^2 + y_4^2 = r_4^2 \\ (x_4 - R \cos k_2 \alpha)^2 + (y_4 - R \sin k_2 \alpha)^2 = r_4^2 \\ (R - R \cos k_2 \alpha)^2 + (1 - R \sin k_2 \alpha)^2 = 2r_4^2 \sin^2 \alpha_4 \end{cases}$$

解得：

$$\begin{cases} r_1 = \frac{R}{2 \sin \alpha_1} \\ x_1 = \frac{R}{2} \\ y_1 = \frac{R}{2 \tan \alpha_1} \end{cases} \quad (12)$$

$$\begin{cases} r_2 = \frac{R}{2 \sin \alpha_2} \\ x_2 = \frac{R}{2 \cos k_2 \alpha} - y_2 \tan k_2 \alpha \\ y_2 = \frac{R \sin k_2 \alpha}{2} + \sqrt{\frac{R^2 \sin^2 k_2 \alpha}{4} - r_2^2 \cos^2 k_2 \alpha} \end{cases} \quad (13)$$

$$\begin{cases} r_3 = \frac{\sqrt{R^2(2 - \cos k_1 \alpha)^2 + (1 - R \sin k_1 \alpha)^2}}{2 \sin \alpha_3} \\ x_3 = \frac{R}{1 + \frac{(2 \sin^2 k_1 \alpha)^2}{\sin^2 k_1 \alpha}} + \sqrt{\frac{R^2}{1 + \frac{(1 - \cos k_1 \alpha)^2}{\sin^2 k_1 \alpha}} - \frac{R_1^2}{1 + \frac{1 - \cos^2 k_1 \alpha}{\sin^2 k_1 \alpha}}} \\ y_3 = \frac{1 - \cos k_1 \alpha}{\sin k_1 \alpha} x_3 \end{cases} \quad (14)$$

$$\begin{cases} r_4 = \frac{\sqrt{R^2(2 - \cos k_2 \alpha)^2 + (1 - R \sin k_2 \alpha)^2}}{2 \sin \alpha_3} \\ x_4 = \frac{R}{1 + \frac{(2 \sin^2 k_2 \alpha)^2}{\sin^2 k_2 \alpha}} + \sqrt{\frac{R^2}{1 + \frac{(1 - \cos k_2 \alpha)^2}{\sin^2 k_2 \alpha}} - \frac{R_2^2}{1 + \frac{1 - \cos^2 k_2 \alpha}{\sin^2 k_2 \alpha}}} \\ y_4 = \frac{1 - \cos k_2 \alpha}{\sin k_2 \alpha} x_3 \end{cases} \quad (15)$$

综上，式 (11) (12) (13) (14) (15) 为对于无人机 M' 的定位模型

### 5.2.2 模型求解

在这里，采用最小二乘法对模型进行求解。将各个圆方程相减，可得其矩阵形式：

$$\mathbf{AX} = \mathbf{B} \quad (16)$$

其中，

$$\mathbf{A} = \begin{pmatrix} 2(x_2 - x_1)x & 2(y_2 - y_1)y \\ 2(x_3 - x_2)x & 2(y_3 - y_2)y \\ 2(x_4 - x_3)x & 2(y_4 - y_3)y \end{pmatrix} \mathbf{X} = \begin{pmatrix} x \\ y \end{pmatrix} \mathbf{B} = \begin{pmatrix} r_1^2 - r_2^2 - (x_1^2 + y_1^2) + (x_2^2 + y_2^2) \\ r_2^2 - r_3^2 - (x_2^2 + y_2^2) + (x_3^2 + y_3^2) \\ r_3^2 - r_4^2 - (x_3^2 + y_3^2) + (x_4^2 + y_4^2) \end{pmatrix}$$

设误差向量为：

$$\epsilon = \mathbf{AX} - \mathbf{B}$$

其误差平方和为

$$E = |\epsilon|^2 = \epsilon^T \epsilon = (\mathbf{AX} - \mathbf{B})^T (\mathbf{AX} - \mathbf{B})$$

要使得误差最小，即使得 E 最小，则对 E 进行求导：

$$\frac{dE}{d\mathbf{X}} = 2\mathbf{A}^T \mathbf{AX} - 2\mathbf{A}^T \mathbf{B} = 0$$

解得：

$$\mathbf{X} = (\mathbf{A}^T \mathbf{A})^{-1} (\mathbf{A}^T \mathbf{B}) \quad (17)$$

由于 M' 坐标只有一个，故可以消去  $k_1, k_2$  得到唯一解。所以仅需新增两架无人机即可完成有效定位

## 5.3 问题一第 (3) 问的模型建立与求解

### 5.3.1 无人机一次移动前后极角变化情况

在无人机的调度过程中，改变的参数为距离中心点的距离  $\rho$  和极角  $\theta$ 。首先我们需要讨论一下极角的变化情况，假设无人机移动前后极角不变。

以无人机 FY00 为极点，FY01 为极轴，建立极坐标系，如图 6 所示。设  $M''(\rho'', \theta'')$  为待接收信号无人机位置， $M'(\rho', \theta')$  为待接收信号无人机移动后位置， $A(\rho_A, \theta_A), B(\rho_B, \theta_B), C(\rho_C, \theta_C)$  为三架发射信号无人机。 $A_0(\rho, \theta_A^0), B_0(\rho, \theta_B^0), C_0(\rho, \theta_C^0)$  为  $A, B, C$  无人机的原本位置。

**证明 1** 证明  $OM'M''$  共线。

$\alpha_A$  为  $\angle OM''A$ ， $\alpha_B$  为  $\angle OM''B$ ， $\alpha_C$  为  $\angle OM''C$ 。连接  $OM''$ ，过  $B$  作  $B'M''$  的平行线交  $OM''$  直线于点  $M'$ ，连接  $M'A^0, M'C^0$ 。我们要由同位角关系，显然有：

$$\angle OM'A^0 = \angle OM''A = \alpha_A$$

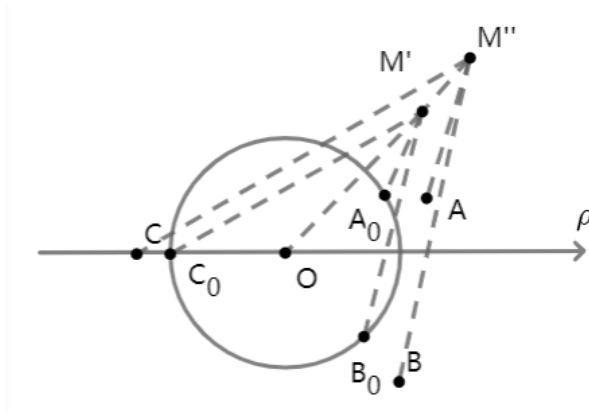


图 6 有偏差无人机移动前后几何关系图

$$\angle OM'B^0 = \angle OM''B = \alpha_B$$

$$\angle OM'C^0 = \angle OM''C = \alpha_C$$

该点  $M'$  满足  $M''$  所处的无人机计算点的要求，即获得的三个角度是由各个发射信号的无人机处在原本位置（标准圆）上时发射的。

**引理 1** 三个圆的交点至多存在两个。

**证明 2** 在平面几何中有且只有一个点  $M'$  满足上述条件

当点  $OA^0M'$  不共线时，即发射信号的无人机与接收信号的无人机不相同时，显然成立。同理，点  $OB^0M'$  不共线且  $OC^0M'$  不共线。

作  $OA^0M'$ ,  $OB^0M'$ ,  $OC^0M'$  的外接圆，分别为  $\odot A$ ,  $\odot B$ ,  $\odot C$

显然该三个圆存在两个交点  $O$  和  $M'$ 。由引理 1 得不存在第三个点使得三圆相交，即不存在别的  $M$  点使得  $\angle OMA^0 = \alpha_A$ ,  $\angle OMB^0 = \alpha_B$ ,  $\angle OMC^0 = \alpha_C$ ，因此  $M'$  唯一。

由证明 12 知，无人机移动前后极角不变

### 5.3.2 基于偏差的无人机定位调度模型

根据题目所给数据，FY00-FY10 无人机的初始位置如图所示。可以观察到其余 9 架无人机和 FY00 的距离各有不同，因此可以把 FY01-FY10 无人机的位置看作是有偏差的，且当有偏差的无人机发射信号时，待接收信号无人机会按照有偏差的方向信息进行位置调整，最终会调整到一个有偏差的位置，也称为为错误理想位置。

因此，无人机调度方案的目标函数是使总偏差最小，即无人机根据方向信息调整后的实际位置与我们希望的理想位置的实际距离最小。

$$\min \sum_{i=1}^n \|p_i^{R'} - p_i^t\|^2 \quad (18)$$

理想位置为:

$$p_i^T = (\frac{1}{n} \sum_{i=1}^n \rho_i, (i-1)\frac{2}{9}\pi) \quad (19)$$

具体调度流程如图 7所示:

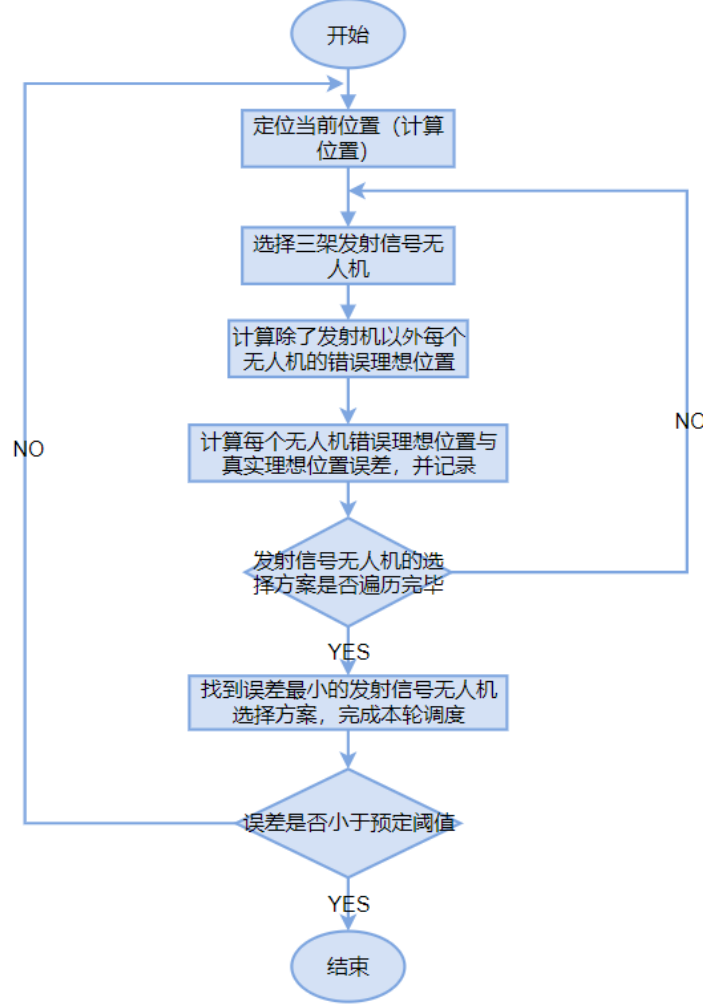


图 7 无人机调度流程图

**step1** 定位当前位置, 把当前位置作为计算位置, 即

$$p_i^C = p_i^R \quad (20)$$

其中, 当前位置为  $p_i^R = (\rho', \theta') = (\rho', \theta'')$

如图 6所示,

$$\angle k = \pi - ((\theta_B^0 - \theta) + \alpha_1)$$

在  $\triangle B^0 M' O$  中,

$$\frac{\rho}{\sin \alpha_1} = \frac{\rho'}{\sin \angle k}$$

解得：

$$\rho' = \frac{\sin(\min 2\pi - (\theta_B^0 - \theta), \theta_B^0 - \theta + \alpha_1)}{\sin \alpha_1} \rho = \frac{\sin(\min 2\pi - (\theta_B - \theta), \theta_B - \theta + \alpha_1)}{\sin \alpha_1} \rho \quad (21)$$

其中， $\theta_B = i_B \frac{2}{9} \pi (i_A = 1, 2, \dots, 9)$

**step2** 计算每个无人机错误理想位置  $p_i^{FI}$ ，得到位移差

$$\vec{p}_D = p_i^{FI} - p_i^C \quad (22)$$

$$p_i^{\vec{R}'} = p_i^{\vec{R}} + \vec{p}_D \quad (23)$$

**step3** 根据公式 (18) (22) (23) 计算与真实理想的误差

**step4** 选择误差最小的发射信号无人机选择方案进行本轮调度

**step5** 重复 step1-step4，直到误差小于预定阈值，本题的模型精度为  $1e-6$

### 5.3.3 模型求解

我们选择贪心算法进行模型的求解。贪心算法，在求解过程的每一步中都采取在当前状态下看来是最好或最优的选择，从而希望最终的结果是最好或最优的。

求解步骤：

1. 将原问题分解成若干轮的无人机移动子问题
2. 对每一子问题分别求解，计算子问题中所有发射信号无人机选择方案及移动误差
3. 得到所有子问题的局部最优解。即在第  $i$  轮移动时，得到在当前位置下，最佳的发射信号无人机选择方案及移动结果
4. 从初始解出发，不断迭代轮数，得到原问题的全局最优解

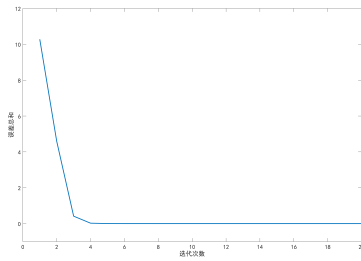
初始条件下的模拟结果如表 1 所示，选出第一轮总误差最小的发射信号无人机方案：FY01, FY04, FY07

通过 matlab 模拟多轮进行求解，得到发射信号无人机的选择方案如表 2 所示。

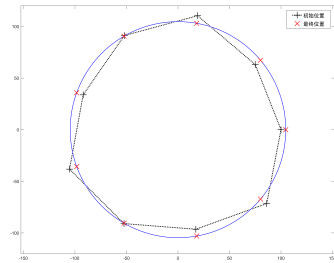
那么无人机的调整方案为：根据表 2 进行发射信号无人机的选择，其余无人机接收方位信息进行调整，进行九轮迭代，即可停止，此时误差精度  $<1e-6$ ，无人机调整完毕的位置见附录表 5，与初始位置进行对比如图 8b 所示：

表 1 第一轮模拟结果

模拟次数	无人机方案	FY01 误差	...	FY09 误差	总误差
1	FY01,FY02,FY03	4.44444	...	4.44444	45.1239
2	FY01,FY02,FY04	4.44444	...	4.44444	38.2773
3	FY01,FY02,FY05	4.44444	...	6.87147	56.4034
...	...	...	...	...	...
84	FY08,FY09,FY07	0.525494	...	7.57402	17.5103



(a) 总误差随迭代次数的变化图



(b) 无人机调整前后位置图

图 8 模型求解结果

表 2 发射信号无人机的选择方案

调整次数	无人机方案			总误差
1	FY01	FY04	FY07	10.2791
2	FY03	FY04	FY05	4.59416
3	FY03	FY06	FY09	0.415374
...	...	...	...	...
20	FY02	FY08	FY09	1.24368E-13

如图 8a所示，此模型对于无人机位置的调整效果好，且进行九轮位置调整便满足误差  $<1e-6$ ，进行五轮位置调整后误差  $<0.01$ 。



## 5.4 问题二的模型建立与求解

实际生活中，无人机编队可能是圆形、锥形、方形等很多形状。这里我们通过研究锥形来推广第一题所得到的结论，锥形编队如图 9 所示。为了研究时更能直观的表现出无人机编队形状的特点，我们均假设同一编队的无人机飞行时均位于同一高度上。

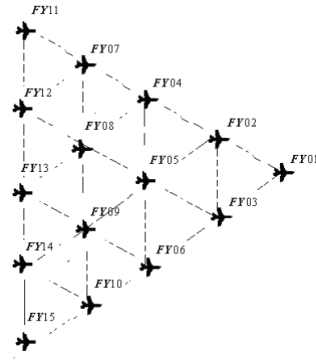


图 9 锥形无人机编队示意图

### 5.4.1 确定发射信号无人机数量

为了便于后续对无人机定位，我们取编号为 FY13 的飞机作为参照飞机，即 FY13 的坐标为  $(0,0)$ 。同时，定义标准情况 FY13-FY01 的方向为  $x$  轴正方向，建立直角坐标系。下面对除了无人机 FY13 以外还需要几架飞机进行讨论：

1. 只增加一架无人机。此时由于信号发射机与任意一架接收机一共只有三个无人机。由于三个点仅能确定一个圆，也即信号接收机的可能轨迹是一个圆。此时无法实现对信号接收机的准确定位。故这种情况不存在
2. 增加两架无人机。此时信号发射机与信号接收机一共四架无人机。由于过三点可以确定一个圆，所以过未知无人机  $M'$  的圆一共有三个，也即这三个圆的交点即为待接收信号无人机的位置。又根据三个互相相交圆的交点个数可能为一个或两个，所以这里进行进一步讨论：

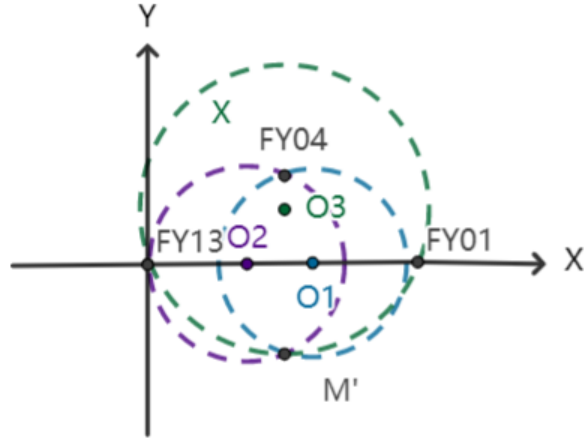


图 10 锥形无人机定位图

(1) 发射信号的两架无人机坐标相较于 FY13 均标准，也即发射信号的无人机所处位置均无偏差。如图 10 所示，取无人机  $FY01(2\sqrt{3}\rho, 0)$  与  $FY04(\sqrt{3}\rho, \sqrt{3}\rho)$  发射信号，其中  $\rho$  为相邻两架飞机之间距离的长度。取点  $FY04, FY01, M'$  构成  $\odot O_1$ ，点  $FY04, FY13, M'$  构成  $\odot O_2$ ，点  $FY01, FY13, M'$  构成  $\odot O_3$ ，根据公式 (11)，其方程组分别为：

$$\begin{cases} (x_1 - \sqrt{3}\rho)^2 + (y_1 - \rho)^2 = r_3^2 \\ (x_1 - 2\sqrt{3}\rho)^2 + y_1^2 = r_1^2 \\ \rho = r_1 \sin \alpha_1 \end{cases}$$

$$\begin{cases} x_2^2 + y_2^2 = r_1^2 \\ (x_2 - \sqrt{3}\rho)^2 + (y_2 - \rho)^2 = r_2^2 \\ \rho = r_2 \sin \alpha_2 \end{cases}$$

$$\begin{cases} x_3^2 + y_3^2 = r_3^2 \\ (x_3 - 2\sqrt{3}\rho)^2 + y_3^2 = r_3^2 \\ \sqrt{3}\rho = r_3 \sin \alpha_3 \end{cases}$$

解得：

$$\begin{cases} r_1 = \frac{\rho}{\sin \alpha_1} \\ x_1 = \frac{3\sqrt{3}\rho + \cot \alpha_1}{2} \\ y_1 = \sqrt{3}x_1 - 4\rho \end{cases} \quad (24)$$

$$\begin{cases} r_2 = \frac{\rho}{\sin \alpha_2} \\ x_2 = \frac{\sqrt{3}\rho + \rho \cot \alpha_2}{2} \\ y_2 = 2\rho - \sqrt{3}x_2 \end{cases} \quad (25)$$

$$\begin{cases} r_3 = \frac{\sqrt{3}\rho}{\sin^2 \alpha_3} \\ x_3 = \sqrt{3}\rho \\ y_3 = \sqrt{3}\rho \cot^2 \alpha \end{cases} \quad (26)$$

综上，式 (11) (24) (25) (26) ( $i \neq 4$ ) 为对于无人机  $M'$  的定位模型

与问题一第 (2) 问类似，根据式 (16) (17) 采用最小二乘法进行求解验证，此情况成立，即三架无人机发射信号即可对剩余无人机完成有效定位

(2) 如果除 FY13 以外的所有无人机坐标可能存在误差的情况，上述方程将会新增两个未知数故无法求解，所以我们考虑再增加一架信号发射无人机进行求解。

#### 5.4.2 锥形编队无人机定位调度模型

在圆形编队中，无人机的实际点与计算点具有较为特殊的性质，其主要原因是圆的对称性强且圆上点到圆心的距离是相同的。在本编队中，我们的原点取得是 FY13 这一架飞机所在位置，显然在本编队中，各飞机到达原点位置是不同的，因此无人机无法再认为所接受到的角是从发射机的原编队位置发射来的。但是无人机可以寻找一个点使得认为从发射机原编队位置发射信号所形成角度尽可能地接近原始接收到的角度，基于以上思想，我们提出一种基于最小化接受角误差的无人机定位模型。具体描述如下：

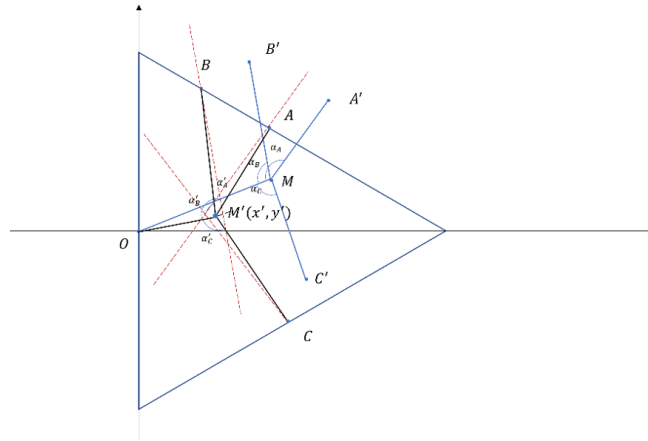


图 11 锥形无人机定位图

首先过  $A$  作  $A'M$  的平行线  $l_A$ ，过  $B$  做  $B'M$  的平行线  $l_B$ ，过  $C$  做  $C'M$  的平行线  $l_C$ ，由于锥形编队的特点，显然三条直线不存在唯一的交点，而是围成一个三角形

区域。此时期望从中找一点  $M'$  使得分别所接收到的角与  $M$  处接收到的角尽可能接近, 如图 11 所示。由于该角属于  $[0, \pi]$  中, 因此使用余弦进行衡量, 目标函数如下:

$$\min \sum_{i=1}^n |\cos \alpha'_i - \cos \alpha_i| \quad (27)$$

设点  $M' = (x', y')$ , 则有

$$\cos \alpha'_i = \frac{\vec{M'O} \cdot \vec{M'B}}{|\vec{M'O}| \cdot |\vec{M'B}|} = \frac{x'^2 + y'^2 - x'x_i - y'y_i}{2x'^2 + 2y'^2 - 2x'x_i - 2y'y_i + x_i^2 + y_i^2} \quad (28)$$

假设  $l_A$ 、 $l_B$ 、 $l_C$  直线方程分别为:

$$l_A: A_1x + B_1y + C_1 = 0 / l_B: A_2x + B_2y + C_2 = 0 / l_C: A_3x + B_3y + C_3 = 0 \quad (29)$$

由  $A'M$  直线与  $B'M$  和  $C'M$  能够求得上述式子中的  $A$  和  $B$ , 其中  $C$  通过  $M'$  点进行求解。具体过程略。

因此, 该处的目标为在三条之间所夹的区域内确定  $M' = (x', y')$  使得目标函数最小。约束条件如下:

$$(A_1x' + B_1y' + C_1) * (A_2x' + B_2y' + C_2) * (A_3x' + B_3y' + C_3) \leq 0 \quad (30)$$

使用 `matlab` 对上述式子进行求解, 解得结果即为  $M$  点的计算位置  $M'$ 。

其余调度思想与问题一第 (3) 问相同。

### 5.4.3 模型求解

首先确定锥形无人机的初始位置 (有误差)。根据无人机锥形编队的性质: 直线上相邻两架无人机的间距相等为 50 m, 可以在直角坐标系下计算出无人机的标准位置。基于标准位置设置 10% 的误差, 计算得到有误差的初始位置, 见附录表 6

具体求解思想与问题一第 (3) 问相同, 采用贪心算法。

初始条件下的模拟结果如表 3 所示, 选出第一轮总误差最小的发射信号无人机方案: FY01, FY09, FY10

通过 `matlab` 模拟多轮进行求解, 得到发射信号无人机的选择方案如表 4 所示。

表 3 第一轮模拟结果

模拟次数	无人机方案	FY01 误差	...	FY15 误差	总误差
1	FY01,FY02,FY04	1.11485	...	12.0180	2985.47
2	FY01,FY02,FY05	1.11485	...	12.0180	2984.57
3	FY01,FY02,FY06	1.11485	...	12.0180	2983.45
...	...	...	...	...	...
352	FY12,FY14,FY15	29.5770	...	4.50855	169.547

表 4 发射信号无人机的选择方案

调整次数	无人机方案			总误差
1	FY01	FY09	FY10	18.4047
2	FY03	FY04	FY05	15.3164
3	FY7	FY12	FY15	12.2350
...	...	...	...	...
50	FY03	FY11	FY15	0.0779

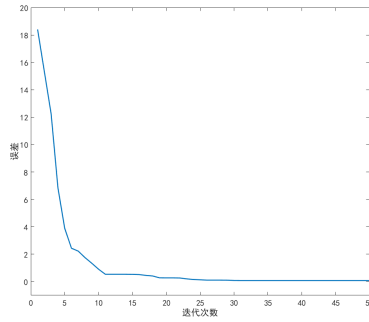
那么无人机的调整方案为：根据表 4 进行发射信号无人机的选择，其余无人机接收方位信息进行调整，进行迭代，大概十轮误差收敛，无人机调整完毕的位置见附录表 7，与初始位置进行对比如图 12a 所示

如图 12b 所示，此模型对于无人机位置的调整效果好，且进行十轮位置调整便满足实际需要。

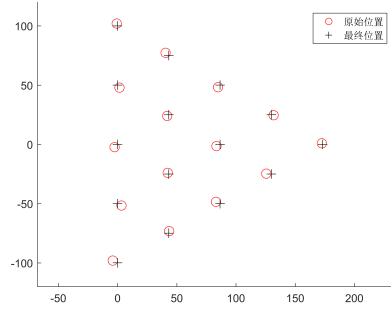
## 六、模型的评价与推广

### 6.1 模型优点

1. 问题一第 (1)(2) 问可以准确的表示出所需要定位的无人机的坐标。
2. 问题一第 (3) 问的模型基于偏差进行定位，能更好的找到最优半径进行调度，且该模型收敛速度快，效果好。



(a) 总误差随迭代次数的变化图



(b) 无人机调整前后位置图

图 12 模型求解结果

3. 问题一第 (3) 问及问题 2 可以对任意要求的飞机编队的半径进行调整与修改，即可以根据实际要求规划目标机群的半径。

4. 对于无人机队形的模型建立及优化对于其他编队的情况可以直接进行思想的迁移，即目标函数都为求解误差最小的标准位置并对无人机群进行调整。

## 6.2 模型缺点

1. 由于专业知识的欠缺以及题目假设条件导致的简化，本文建立的模型无法完全模拟无人机在实际飞行过程中由于速度、方向等所造成的位置调整情况。

2. 无人机在实际飞行时如果不能向地面发射信号需考虑其高度的误差，本模型在符合题目实际背景的前提下没有对高度差问题进行推广。

## 6.3 模型推广

1. 设计其他编队队形的无人机位置调整模型时，可以多分析几种队形，增加模型的普适性。

2. 在各个模型的设计时，可以加入时间、速度、高度等因素。

# 七、参考文献

[1] Y. Zeng, R. Zhang and T. J. Lim, "Wireless communications with unmanned aerial vehicles: opportunities and challenges," in IEEE Communications Magazine, vol. 54, no. 5, pp. 36-42, May 2016, doi: 10.1109/MCOM.2016.7470933.

[2] 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.

## 附录 A 支撑材料列表

### 问题一第 (3) 问

#### 代码

arrs.mat

kOperate.m

plot3Circle.m

plotPositions.m

polorToXYZ.m

q3Simulate.m

xyzToPolor.m

#### 数据

无人机最优调度方案 1.xlsx

第一轮模拟结果 1.xlsx

### 问题二

#### 代码

deleteNoNeedFromCms.m

getConicalPlanesPos.m

etRandConPlanePos.m

plotPositionsConinal.m

q4Solve.m

#### 数据

无人机最优调度方案 2.xlsx

第一轮模拟结果 2.xlsx

## 附录 B 相关数据

表 5 问题一第 (3) 问无人机最终位置坐标

无人机编号	极角 $\theta$	极径 $\rho$
FT00	0	0
FY01	0	104.63
FY02	40	104.63
FY03	80	104.63
FY04	120	104.63
FY05	160	104.63
FY06	200	104.63
FY07	240	104.63
FY08	280	104.63
FY09	320	104.63



表 6 问题二无人机初始位置坐标

无人机编号	x	y
FY01	172.49	0.85
FY02	131.81	24.59
FY03	125.57	-24.71
FY04	84.89	48.20
FY05	83.54	-1.51
FY06	83.07	-48.50
FY07	40.58	77.34
FY08	41.96	24.07
FY09	42.41	-24.04
FY10	43.64	-73.12
FY11	-0.7	101.97
FY12	1.6	47.93
FY13	-2.56	-2.30
FY14	3.27	-51.65
FY15	-4.09	-98.11

表 7 问题二无人机最终位置坐标

无人机编号	x	y
FY01	173.20	0.00
FY02	129.90	25.00
FY03	129.90	-25.00
FY04	86.60	50.01
FY05	86.60	0.00
FY06	86.60	-50.00
FY07	43.30	75.01
FY08	43.30	25.00
FY09	43.30	-25.00
FY10	43.30	-75.00
FY11	0.00	100.01
FY12	0.00	50.01
FY13	0.00	0.00
FY14	0.00	-50.00
FY15	0.00	-100.00

## 附录 C 问题一第 (3) 问代码

```

kOperate.m
function [theta] = kOperate(theta)
% 保证theta在0-180度中
theta = abs(theta);
theta = min(360-theta, theta);
end

plot3Circle.m
% 绘制三圆
rou = 20;

```

```

a = [rou,40];
b = [rou,160];
c = [rou,280];

% 未知点
M = [rou+10,130];

ad = polorToXYZ(a); bd = polorToXYZ(b); cd = polorToXYZ(c); Md = polorToXYZ(M);
od = [0,0];

%plot circle
% viscircles([0 0],rou,'Color','b');%圆心坐标为(0,0), 半径为100,轮廓颜色为蓝色
axis equal
% set(gca,'XLim',[-30 30]);
% set(gca,'YLim',[-30 30]);
hold on
%% plot 直线
% %am
% plot([ad(1),Md(1)], [ad(2),Md(2)], 'k','linewidth',2);
% %bm
% plot([bd(1),Md(1)], [bd(2),Md(2)], 'k','linewidth',2);
% %cm
% plot([cd(1),Md(1)], [cd(2),Md(2)], 'k','linewidth',2);
%
% %ao
% plot([ad(1),0], [ad(2),0], 'k','linewidth',2);
% %bo
% plot([bd(1),0], [bd(2),0], 'k','linewidth',2);
% %co
% plot([cd(1),0], [cd(2),0], 'k','linewidth',2);
%
% %mo
% plot([Md(1),0], [Md(2),0], 'k','linewidth',2);

% 画外接圆
plotOutTri([ad;od;Md])
plotOutTri([bd;od;Md])
plotOutTri([cd;od;Md])

plotOutTri.m
function plotOutTri(p)
% p=rand(3,2);% (x,y)

cen1=(p(1,:)+p(2,:))/2; %三角形一条边中点
cen2=(p(2,:)+p(3,:))/2; %另一条边中点

k1=-1/((p(1,2)-p(2,2))/(p(1,1)-p(2,1))); %一条边垂直平分线

```

```

b1=cen1(2)-k1*cen1(1);

k2=-1/((p(2,2)-p(3,2))/(p(2,1)-p(3,1))); %另一条边垂直平分线
b2=cen2(2)-k2*cen2(1);

x0=-(b1-b2)/(k1-k2); %求两直线交点
y0=-(-b2*k1+b1*k2)/(k1-k2);

r=sqrt((y0-p(1,2))^2+(x0-p(1,1))^2);

hold on;
% plot(p(:,1),p(:,2));
p=circshift(p,1);
% plot(p(:,1),p(:,2));

theta=0:0.01:2*pi;
x=x0+r*cos(theta);
y=y0+r*sin(theta);
plot(x,y,'-',x0,y0,'.','linewidth',2);
axis equal
end

plotPositions.m
% 绘画positions
% pos = [[100,0;102.947370926424,40.9878215447839;104.701621875316,79.7685766852877;
105,119.750000000000;101.584229317453,159.399501370066;108.092849067949,
199.397185466715;105.000000000000,240.070000000000;102.181496909929,279.878588984783;
107.503207550062,320.932826590393]];

function plotPositions(pos)
clf
rou = mean(pos(:,1));

set(gca,'XLim',[-120 120]);
set(gca,'YLim',[-120 120]);

% 画圆
viscircles([0 0],rou,'Color','b','linewidth',1);%圆心坐标为(0,0), 半径为100,轮廓颜色为蓝色
hold on;

x = [0];
y = [0];

for i=1:size(pos,1)
tmp = polorToXYZ(pos(i,:));
x(i+1) = tmp(1);
y(i+1) = tmp(2);

```

```

end

scatter(x,y,'k');
plot([x(2:end),x(2)],[y(2:end),y(2)],'r');
axis equal
drawnow;

end

polorToXYZ.m
function [P] = polorToXYZ(p)
P = zeros(1,2);
rou = p(1); theta = p(2);
x = rou*cosd(theta);
y = rou*sind(theta);
P(1) = x; P(2) = y;
end

```

```

q3Simulate.m
%% Question 模拟
clear

% 原始位置1-9 共10台机
pos = [100, 0;
98, 40.10;
112, 80.21;
105, 119.75;
98, 159.86;
112, 199.96;
105, 240.07;
98, 280.17;
112, 320.28
];

% rng(1);
% % 误差20%
%
% stdpos = [100, 0;
% 100, 40;
% 100, 80;
% 100, 120;
% 100, 160;
% 100, 200;
% 100, 240;
% 100, 280;
% 100, 320
% ];

```

```

%
% pos = stdpos + rand(size(stdpos))*40-20;
% pos(pos(:,2)<0,2) = pos(pos(:,2)<0,2) + 360;

% 选择三台机
arr = [3,6,9];

% 移动系数 lmd1,lmd2 from Oto1
lmd1 = 1;
lmd2 = 1;
lmd = [lmd1, lmd2];

%% 排列
% arrs = combntns([1:9],3);
load('arrs.mat'); % arrs 84*3

%% 总运行次数
total_arr_num = 84;
plane_num = 9;

%% 迭代次数
maxIter = 20;

% 变量preset
res_planeses = zeros(maxIter, length(arr));
res_errorres = zeros(maxIter,1);
res_positiones = cell(maxIter,1);
res_detail_errorres = zeros(maxIter,plane_num);

for iter = 1:maxIter
    iter
    total_dises = zeros(total_arr_num,plane_num);
    positionses = cell(total_arr_num,1);
    %% 调用代码开始模拟
    for j=1:total_arr_num
        % j = 1;
        arr = arrs(j,:);

        [dises, positions] = simulateArr(arr,lmd, pos);

        total_dises(j,:) = dises';
        positionses{j} = positions;
    end

    sum_dises = sum(total_dises,2);
    %% 单次结束后, 第j个组合
    [~, index] = sort(sum_dises, 'ascend');

```

```

first_ind = index(1);
% 结果飞机调度, 总误差, 最后飞机位置, 误差细节
res_planes = arrs(first_ind,:);
res_error = sum_dises(first_ind);
res_position = positionses{first_ind};
res_detail_errors = total_dises(first_ind,:);

res_planeses(iter,:) = res_planes;
res_errores(iter) = res_error;
res_positiones{iter} = res_position;
res_detail_errorses(iter,:) = res_detail_errors;

pos = res_position;

plotPositions(pos);

end

figure();
plot(res_errores);

function [dises, positions] = simulateArr(arr, lmd, pos)

% 理想角 (360/9)
I_Angles = [0,40,80,120,160,200,240,280,320];

% 理想rou
rou = 100;

% 飞机个数
plane_num = 9;

% % 原始位置1-9 共10台机
% pos = [100, 0;
% 98, 40.10;
% 112, 80.21;
% 105, 119.75;
% 98, 159.86;
% 112, 199.96;
% 105, 240.07;
% 98, 280.17;
% 112, 320.28
% ];

dises = zeros(plane_num,1);

positions = zeros(plane_num, 2);

```

```

p_T_rou = mean(pos(:,1));

%% 测试固定半径
% p_T_rou = 110;

%
for i=1:plane_num

%% 开始运算
% i = 3;
p_R = pos(i,:); % 当前实际位置（极坐标）

%% 计算当前位置
p_C = zeros(1,2); % 当前计算位置（极坐标）
p_C(2) = p_R(2); % theta' = theta';

theta_R = p_R(2);

% 第几台机，存在精度误差，取中位数
rou_cs = zeros(3,1);

for k=1:length(arr)
indArrPlane = k;
% 求解alphaA和thetaA
thetaA = I_Angles(arr(indArrPlane));

% 测试利用全圆定位通过
% 用实际飞机定位，利用向量积求角（先转为平面直角坐标）
ARealPos = pos(arr(indArrPlane),:); % 极坐标
OPos = [0,0];
AM = polarToXYZ(p_R) - polarToXYZ(ARealPos);
OM = polarToXYZ(p_R) - polarToXYZ(OPos);
sigma = acos(dot(AM,OM)/(norm(AM)*norm(OM)));
alphaA = sigma*180/pi;

% 求解rou'
rou_c_coe = sind(kOperate(thetaA - theta_R) + alphaA)/sind(alphaA);
rou_c = rou*rou_c_coe;

rou_cs(k) = rou_c;
end
rou_c = median(rou_cs);

p_C(1) = rou_c;

%% step2 计算 当前位置与假性理想位置差（直角坐标）

```



```

p_FK = [rou, I_Angles(i)];

% 位置差（直角坐标）
p_D = polarToXYZ(p_FK) - polarToXYZ(p_C);

% 乘上移动系数
p_D = lmd.*p_D;

%% step3 换算到当前位置下p_R（即进行移动）
p_R_XYZ = polarToXYZ(p_R);

% 若是调度的飞机则不移动
if ismember(i,arr)
    p_R_XYZ = polarToXYZ(p_R);
else
    p_R_XYZ = p_R_XYZ + p_D;
end

p_R_after = xyzToPolar(p_R_XYZ);

clear TH R

%% step4 计算与真实理想位置
% 计算真实理想位置
p_T_polar = [p_T_rou, I_Angles(i)];
p_T_XYZ = polarToXYZ(p_T_polar);

% 计算距离
dis = sqrt(sum((p_R_XYZ - p_T_XYZ).^2));

% 赋值
dises(i) = dis;
positions(i,:) = p_R_after;
end
%

end

xyzToPolar.m
function p = xyzToPolar(p_xyz)
p = zeros(1,2);
[TH, R]= cart2pol(p_xyz(1),p_xyz(2));
p(1) = R;
p(2) = TH*180/pi;
if p(2) < 0
% 为负则变成正
p(2) = 360 + p(2);

```

```
end
```

```
end
```

## 附录 D 问题二代码

```
deleteNoNeedFromCms.m

clear
% 鉴别组合和不需要的
noneed = [1 2 3
1 4 6
1 7 10
1 11 15
1 12 14
1 8 9
5 2 3
5 4 6
5 7 10
5 11 15
5 12 14
5 8 9
];

need_plans_plane = [1,2,3,4,5,6,7,8,9,10,11,12,14,15];

arrs = combntns(need_plans_plane,3);
newArrs = [];
k = 1;
flag = true;

for j = 1:size(arrs,1)
    flag = true;
    for i = 1:size(noneed,1)
        if sum(sort(noneed(i,:)) == sort(arrs(j,:))) == 3
            % 存在了
            flag = false;
        end
    end
    if flag == true
        % 添加
        newArrs(k,:) = arrs(j,:);
        k = k+1;
    end
end
```

```

end
arrs = newArrs;

plotPositionsConinal.m
function [position] = getConicalPlanesPos(number)
% 返回锥形所处位置i的飞机
X = [
4,0;
3,1;
3,-1;
2,2;
2,0;
2,-2;
1,3;
1,1;
1,-1;
1,-3;
0,4;
0,2;
0,0;
0,-2;
0,-4];
position = X(number,:);
end

getRandConPlanePos.m
function [position] = getRandConPlanePos(number)
% 获取飞机真实位置
% 误差1%内
ratio = 0.01;
X = [
4,0;
3,1;
3,-1;
2,2;
2,0;
2,-2;
1,3;
1,1;
1,-1;
1,-3;
0,4;
0,2;
0,0;
0,-2;
0,-4];
X = X + rand(size(X))*ratio*2-ratio; %[-0.05,0.05];

```

```

position = X(number,:);
end

plotPositionsConinal.m
% 绘画positions
% pos =
    [[100,0;102.947370926424,40.9878215447839;104.701621875316,79.7685766852877;105,119.750000000000;101.5

function plotPositionsConinal(pos)
% clf
% rou = mean(pos(:,1));

% x = [0];
% y = [0];

% for i=1:size(pos,1)
%     x(i) = tmp(1);
%     y(i) = tmp(2);
% end

x = pos(:,1)';
y = pos(:,2)';

scatter(x,y,72,'k+');
set(gca,'XLim',[-120 120]);
set(gca,'YLim',[-120 120]);
% plot([x(1:end),x(1)],[y(1:end),y(1)],'r');
axis equal
% drawnow;

end

q4Slove.m
%% q4 求解
clear
ratio = 0.1;
sdPos = [
4,0;
3,1;
3,-1;
2,2;
2,0;
2,-2;
1,3;
1,1;
1,-1;

```

```

1,-3;
0,4;
0,2;
0,0;
0,-2;
0,-4];

rdPos = sdPos + rand(size(sdPos))*ratio*2-ratio; %[-0.05,0.05];

% 间距
sigma = 50;

new_sigma = 45; % 固定间距

plane_num = 14;
% new_sigma = 45; % 新间距

lmd = [1,1];
arr = [2,6,7];

%% 实际计算
rng(1);

onex = sqrt(3)/2;
oney = 1/2;
std_cords = [onex, oney];

% std and new
cords = sigma * std_cords;
new_cords = new_sigma * std_cords;

sdIdealpos = sdPos.*cords;
pos = rdPos.*cords;

dises = zeros(plane_num,1);
positions = zeros(plane_num, 2);
%plane number

need_plans_plane = [1,2,3,4,5,6,7,8,9,10,11,12,14,15];

% arrs = combntns(need_plans_plane,3);
load('arrsQ4.mat');
total_arr_num = size(arrs,1);

%% 迭代次数
maxIter = 50;

```

```

% 变量preset
res_planeses = zeros(maxIter, length(arr));
res_erroses = zeros(maxIter,1);
res_positiones = cell(maxIter,1);
res_detail_erroses = zeros(maxIter,plane_num);

for iter = 1:maxIter
    iter
    total_dises = zeros(total_arr_num,plane_num);
    positionses = cell(total_arr_num,1);

    for arrt=1:total_arr_num
        arr = arrs(arrt,:);

        for i = 1:plane_num
            % 共14架
            pn = need_plans_plane(i);

            % pn = 8;

            O = [0,0];
            %标准机位置
            % 4
            sdA = sdIdealpos(arr(1),:);
            % 7
            sdB = sdIdealpos(arr(2),:);
            % 6
            sdC = sdIdealpos(arr(3),:);

            % 偏移后
            A = pos(arr(1),:);
            B = pos(arr(2),:);
            C = pos(arr(3),:);

            M = pos(pn,:);

            %% 求解直线参数
            % A,B
            [la_params(1),la_params(2)] = solveAB(M,A);
            [lb_params(1),lb_params(2)] = solveAB(M,B);
            [lc_params(1),lc_params(2)] = solveAB(M,C);

```

```

% C
la_C = -la_params(1)*sdA(1) - la_params(2)*sdA(2);
lb_C = -lb_params(1)*sdB(1) - lb_params(2)*sdB(2);
lc_C = -lc_params(1)*sdC(1) - lc_params(2)*sdC(2);

[xab,yab] = solveLineInsect([la_params,la_C], [lb_params,lb_C]);
[xbc,ycb] = solveLineInsect([lb_params,lb_C], [lc_params,lc_C]);
[xac,yac] = solveLineInsect([la_params,la_C], [lc_params,lc_C]);

% 用边界做初始值
x0 = [xac,yac];

params = [la_params;lb_params;lc_params]; %3*2
Ces = [la_C,lb_C,lc_C];

noteqcons = x0*params' + Ces ; %1*3 <=0

% 标准点集合
sdPoints = [sdA;sdB;sdC];
% 标准alpha
alphaA = calAlpha(A,M,0);
alphaB = calAlpha(B,M,0);
alphaC = calAlpha(C,M,0);
alphas = [alphaA,alphaB,alphaC];

X_tmp = [xab,yab;xbc,ycb;xac,yac];
X_tmp = [X_tmp;mean(X_tmp(:,1)),mean(X_tmp(:,2))];
tmp_err = zeros(4,1);
for j=1:4
tmp_err(i) = cosError(X_tmp(j,:),sdPoints,alphas);
end
[~,ind] = sort(tmp_err,'ascend');
first_ind = ind(1);
res = X_tmp(first_ind,:);
fval = tmp_err(first_ind);
% initial_error = cosError(x0,sdPoints,alphas);
%
% fun = @(X)cosError(X,sdPoints,alphas);
% conFun = @(X)unlinearC(X,params,Ces);
%
% options = optimset('Algorithm','sqp');
% [res,fval] = fmincon(fun,x0,[],[],[],[],[],[],conFun,options);

% % 画图查看点和区域分布
% plot([xab,xbc,xac,xab],[yab,ycb,yac,yab]);
% hold on;

```

```

% scatter(res(1),res(2));

%% res为计算距离，要移动到目标位置
p_FI = sdIdealpos(pn,:);
p_R = M;
p_C = res;
p_T = sdIdealpos(pn,:);

%% 移动向量
p_D = p_FI - p_C;

p_D = lmd .* p_D; % 控制移动步长

%% 飞机移动
% 若是调度的飞机则不移动
if ismember(pn,arr)
p_R_after = p_R;
else
p_R_after = p_R + p_D;
end

% 计算误差
dis = sqrt(sum((p_R_after - p_T).^2));

% 赋值
dises(i) = dis;
positions(i,:) = p_R_after;

end

total_dises(arrt,:) = dises';
positionses{arrt} = positions;

end

sum_dises = sum(total_dises,2);
%% 单次结束后，第j个组合
 [~, index] = sort(sum_dises,'ascend');
first_ind = index(1);
% 结果飞机调度，总误差，最后飞机位置，误差细节
res_planes = arrs(first_ind,:);
res_error = sum_dises(first_ind);
res_position = positionses{first_ind};
res_detail_errors = total_dises(first_ind,:);

res_planeses(iter,:) = res_planes;

```



```

resErrores(iter) = res_error;
res_positiones{iter} = res_position;
res_detail_errors(iter,:) = res_detail_errors;

% pos = res_position;
% 添加原点
pos(1:12,:) = res_position(1:12,:);
pos(14:15,:) = res_position(13:14,:);
pos(13,:) = [0,0];

plotPositionsConical(pos);
end

figure();
plot(resErrores);

function [c,ceq] = unlinearC(X,params,Ces)
% X(x0,y0)
ctmp = X*params' + Ces ; % 1*3 <=0
c = prod(ctmp); %1*1 <=0
ceq = [];
end

```