

基于鸽群机制的无人机无源定位编队

摘要

本文对无人机的无源纯向定位这一问题展开研究，建立了不同的几何模型以及鸽群模型对问题进行求解。

问题一的第一问：首先对无人机编队的圆周均匀模型进行分析，建立了基本的模型，选定相邻的两架无人机，然后对处于圆心位置的无人机连线得到各个角度的位置关系，对三架无人机形成的三角形使用正弦定理建立方程式，对发射信号的无人机考虑其几何模型，判断被动接收信号无人机位于理想位置，建立被动接收信号无人机的几何模型，联立各式得到公式 5-12 解出被接收信号无人机的几何模型。

问题一的第二问：第二问在问题一的基础上进行分析，对每架无人机进行逐一分析，基于问题一中的公式对每一架无人机的航向角以及极坐标进行分析，建立三个几何模型得到无人机的发射信号选取方案，模型四：选定与编号 FY00, FY01, FY02, FY03 作为信号发射无人机。模型五：选定与编号 FY00, FY01, FY02, FY04 作为信号发射无人机。模型六：选定与编号 FY00, FY01, FY02, FY05 作为信号发射无人机。

问题一的第三问：建立鸽群模型以及人工势场模型对圆周模型进行分析，采用无人机自主编队控制模型，对无人机进行运动模型建模，鸽群模型建模，人工势场模型建模，最终得到公式 5-22，成功拟合仿真出实验结果，使得无人机能够朝向理想状态的圆周模型进行移动。

关键词：无人机；鸽群机制；编队控制；人工势场。

一、问题重述

无人机集群在遂行编队飞行时，每架无人机都有固定的编号，各个无人机之间的相对位置关系保持不变，但存在某些无人机的真实位置相较理想位置有所偏差，为科学合理的调整偏差无人机的位置达到理想状态，并避免发射电磁波信号的时候存在外界干扰，保证无人机调整过程中的安全性，拟采用纯方位无源定位的方法获取各无人机之间的位置信息。即从编队中选定某几架无人机向其他无人机发射信号，其余无人机被动接收信号，通过发射信号无人机的理想位置与接收信号无人机的接收角度信息，间接定位出接收信号无人机的真实位置，通过真实位置与理想位置的误差值来调整无人机达到无偏差的状态。

问题一：圆形编队由 10 架无人机组成，编号 FY00 位于圆心，其余 9 架无人机均匀分布在同一圆周上，无人机的高度保持一致。

(1) 无人机编队中，已知编号 FY00 的无人机位于圆形编队的圆心处，为无偏差的信号发射无人机，并且在圆周上还存在 2 架编号已知编号且无偏差的信号发射无人机，针对其余被动接收信号的无人机来说可能存在位置略有偏差的情况，建立其余被动接收信号无人机的位置模型。

(2) 无人机编队中，已知编号 FY00、FY01 的无人机为无偏差的信号发射无人机，而某位置略有偏差的无人机接收到 FY00、TY01 以及其他未知编号但无偏差的无人机发射的信号，判断至少还需要几架无偏差的信号发射无人机，才能建立其余被动接收信号无人机的位置模型。

(3) 无人机编队中，已知编号 FY00 的无人机位于半径为 100m 的圆形编队的圆心处，为无偏差的信号发射无人机，并且在圆周上最多还存在 3 架无偏差的信号发射无人机，通过表 1 所给的所有无人机的初始位置数据，要求提出一种调整方案，使得在圆周上的 9 架无人机最终均匀分布在某个圆周上。

问题 2: 结合实际情况考虑, 将纯方位无源定位思想应用于无人机锥形编队队形, 在该队形中每相邻两架无人机的间距为 50m, 若某架信号接收无人机出现位置偏差的情况, 要求设计一个适用于该队形的无人机位置调度方案。

二、问题分析

2.1 问题一的第一问分析

针对问题一中第一小问, 题中所给发射信号无人机条件, 得出除位于圆心编号为 FY00 的无人机, 其他两架发射信号无人机一定位于圆周上, 圆心角已知, 且被动接收信号无人机的位置与其与发射信号无人机之间角度有关, 建立出数学表达式。将角度之间的关系一一列举, 利用正弦定理建立被动接收信号无人机所在位置极角模型以及极径模型。

2.2 问题一的第二问分析

针对问题一中的第二小问, 已知编号 FY00、FY01 的无人机为无偏差的信号发射无人机, 其余无人机中分为若干无偏差且未知编号的信号发射无人机和未知编号且被动接收信号的无人机, 回顾第一问可知, 采用三个已知编号的无人机可定位其他被动接收信号的无人机。若当编号未知时, 存在接收到同一角度的信号, 无法判断信号源的位置, 通过判断 a_1 和 a_2 的偏差大小可确定信号来源的方向, 故可将其转化为第一问的模型求解。但是存在一种特殊情况, 当待定位的无人机具有偏差, 与编号 FY00、FY01 的无人机处于统一水平线上时, 无法通过判断 a_1 和 a_2 的偏差大小实现定位, 故此时应再加一个无偏差的信号发射无人机。

2.3 问题一的第三问分析

针对问题一中的第三小问, 需要对题干所给的数据, 确定好均匀分布在圆周模型的各个无人机的理想状态位置, 尽可能的去减少发射信号无人机的数量, 至多三台无人机作为信号发射源。且要给出一个合理的调整方案, 根据表中所给出的真实数据, 对无人机的位置矢量信息进行修正。已知在空间内不共线的三点可以确定另外一点的位置信息, 依照“就近原则”, 基于鸽群机制的领导层次模型, 对无人机逐一层级编号, 由一个核心领导无人机以及另外两台无人机作为信号发射源, 对下属层次无人机逐个确定位置矢量, 依次轮换交替, 实现圆周模型的均分分布调整方案。

2.4 问题二的分析

针对问题二，需要考虑到不仅仅是问题一的圆周模型，在实际场景中，如三角模型，锥形模型，矩形模型等，都是在无人机的编队控制系统中所能用到的。对于小型的无人机点阵模型展示技术，无人机的数量也上升到一个量级，不仅仅是十几架的无人机编队，也可能是几百架的一个数量级。多余庞大的无人机编队数量，如果没有一个合理的调整方案，无视无人机发射信号所带来的干扰源，在执行相关任务的过程中，可能会出现撞机，追尾，飞行不稳定等现象。对于问题二中有可能出现的编队二维平面模型，采用基于人工势场的鸽群生物模型进行求解，在鸽群的生物模型中，总是有一个处于绝对领导位置的头鸽，依次向下形成层次模型，有效的规划好编队模型的队列，速度，航向角，根据预先设置好的编队夹角，无人机之间的欧式间距，在二维的笛卡尔坐标系中有序的按照势场模型运动，朝向理想状态下引力级目标位置运动，避开有障碍物的斥力级运行，以免出现撞机等情况，有效的解决无人机的编队问题。

三、模型假设

- 1. 假设无人机位置略有偏差时，位置不存在偏离超过邻近范围无人机。
- 2. 假设所有无人机的时钟同步，接收信号的延迟忽略不计；
- 3. 假设无人机在平面内的运动视为质点运动。

四、符号说明

本论文所使用到的相关符号如表 4-1 所示。

表 4-1 符号说明表

符号	含义	单位
γ	无人机极坐标极角	°
ρ	无人机极坐标极径	m ³
V_i^c	控制系统输出	m/s
α_i^c	航线角	°
X_i	无人机的欧式距离	m
v_i	无人机初始速度	m/s
Kp	势场常数因子	

K_v	速度的反馈常数因子	
t_v	速度时间因子	s
t_α	无人机航向角时间因子	s

五、模型建立与求解

在被动接收信号无人机位置的确定中，利用现已知条件，考虑寻找各角度之间的关系通式，确立被动接收信号无人机的极坐标通式为位置确立模型。

5.1 问题一的模型建立与求解

5.1.1 第一问的模型建立

由题可知，问题一中发射信号无人机均匀分布在圆周上，三架无人机所构成圆周角 $\theta = (K-1)2\pi/9$ ($K=0, 1, 2, 3, 4$)， K 表示分布在圆周上两架发射信号无人机之间跨越几架无人机，00 代表编号为 FY00 的无人机，01 代表编号为 FY01 的无人机，依次类推，根据一小问中约束条件，建立基本模型建立如下。

问题中根据角度关系求解极坐标，主要分为考虑极角 $\gamma = \alpha + \theta$ 、 $\gamma = \alpha$ 、 $\gamma = 2\pi - \alpha$ 三种情况建立模型。

模型一：选定相邻两架无人机作为发射信号无人机，如：FY01、FY02 或 FY02、FY03。被动接收信号无人机与发射信号无人机分布如图 5-1 所示：

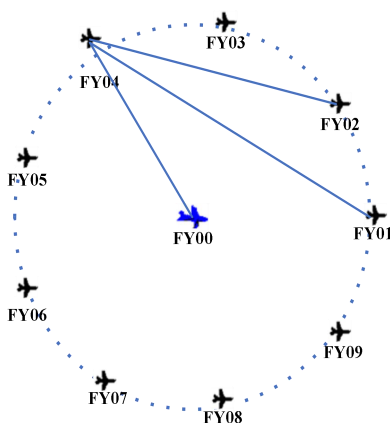


图 5-1 无人机分布图

如图 5-1 所示，被动接收信号无人机未位于理想位置，其极角 $\gamma = \alpha + \theta$ 。

利用被动接收信号无人机与任意两架发射信号无人机之间形成的夹角 α_1 、 α_2 ，并连接 FY00、FY01，FY00、FY02，对形成的三角形使用正弦定理可建立方程式，其示意图如图 5-2 所示：

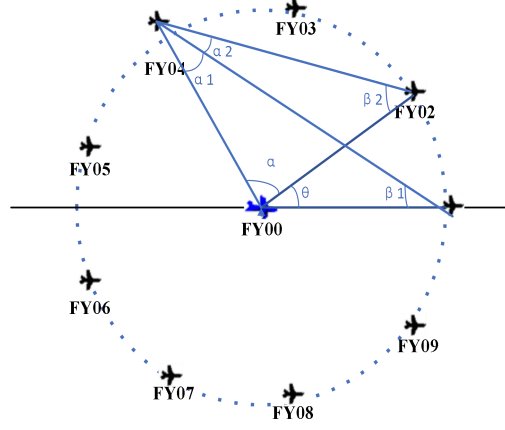


图 5-2 位置关系图

图 5-2 中， $0k$ 代表编号为 FY0k($k \in \{3, 4, 5\}$)的无人机，以 00-01 之间连线为极轴， $\alpha_1 = \angle 00-0k-01$ ； $\alpha_2 = \angle 00-0k-02$ ； $\beta_1 = \angle 00-01-0k$ ； $\beta_2 = \angle 00-02-0k$ ； $\alpha = \angle 02-00-0k$ ； $\theta = \angle 01-00-02$ 。

利用问题中所给条件，结合图 5-2 可得式(5-1)：

$$\begin{cases} \frac{\sin(\alpha_1)}{r} = \frac{\sin(\beta_1)}{\rho} \\ \frac{\sin(\alpha_1 + \alpha_2)}{r} = \frac{\sin(\beta_2)}{\rho} \end{cases} \quad (5-1)$$

式(5-1)中， r 表示位于圆周上发射信号无人机极径； ρ 表示被动接收信号无人机极径。

根据三角形内角和为 π ，建立式(5-2)：

$$\begin{cases} \beta_1 = \pi - (\alpha + \alpha_1 + \theta) \\ \beta_2 = \pi - (\alpha_1 + \alpha_2 + \alpha) \end{cases} \quad (5-2)$$

将式(5-2)代入(5-1)中，化简得到式(5-3)：

$$\begin{cases} \alpha = \arctan \frac{[\sin(\alpha_1 + \theta) - \sin(\alpha_1)] \sin(\alpha_1 + \alpha_2)}{\sin(\alpha_1) \cos(\alpha_1 + \alpha_2) - \sin(\alpha_1 + \alpha_2) \cos(\alpha_1 + \theta)} \\ \rho = \frac{\sin(\alpha) \cos(\alpha_1 + \theta) + \cos(\alpha) \sin(\alpha_1 + \theta)}{\sin(\alpha_1)} * r \end{cases} \quad (5-3)$$

式(5-3)中 $(\alpha+\theta)$ 为被动接收信号无人机极角； ρ 表示被动接收信号无人机极径。得到极坐标为 $(\rho, \alpha+\theta)$ 。

模型二：选定发射信号无人机之间相隔 2 到 3 架无人机，如 FY01、FY04，FY01、FY05。无人机分布如图 5-3 所示：

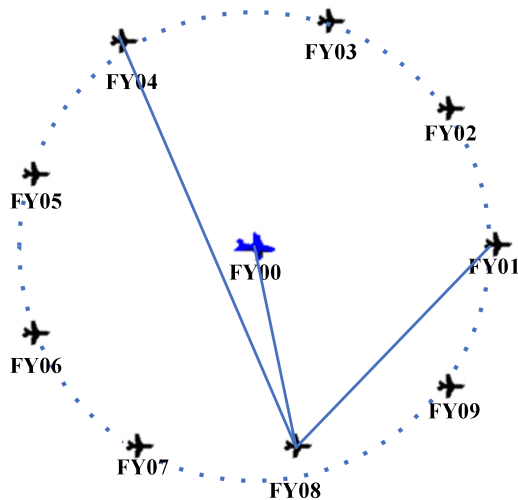


图 5-3 无人机分布图

如图 5-3 所示，被动接收信号无人机未位于理想位置，其极角 $\gamma=\alpha+\theta$ 。同模型一中，作连线，构建无人机之间的三角形，角度之间的关系如图 5-4 所示：

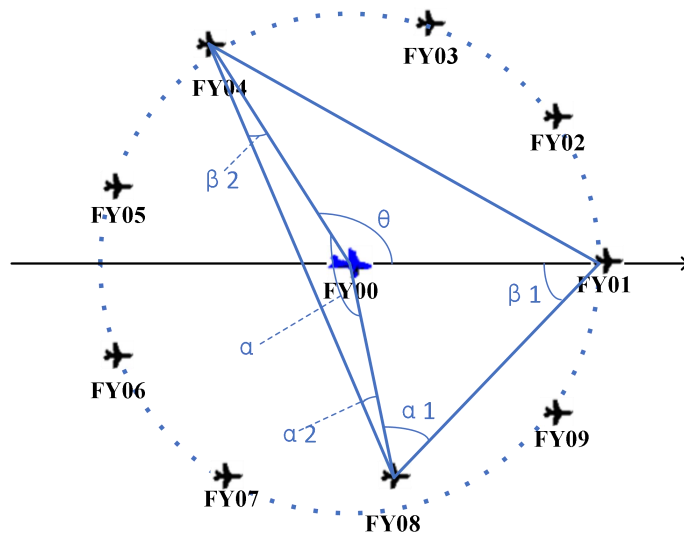


图 5-4 位置关系图

图 5-4 中， $0k$ 代表编号为 FY0k($k \in \{6, 7, 8\}$)的无人机以 00-01 之间连线为极轴， $\alpha_1 = \angle 00-0k-01$ ； $\alpha_2 = \angle 00-0k-04$ ； $\beta_1 = \angle 00-01-0k$ ； $\beta_2 = \angle 00-04-0k$ ； $\alpha = \angle 04-00-0k$ ； $\theta = \angle 01-00-04$ 。由三角形内角和为 π ，得出角度之间关系式(5-4)：

$$\begin{cases} \beta_1 = \pi - [(2\pi - \alpha - \theta) + \alpha_1] \\ \beta_2 = \pi - (\alpha_2 + \alpha) \end{cases} \quad (5-4)$$

根据式(5-4)中角度关系，利用正弦定理建立式(5-5)：

$$\begin{cases} \frac{\sin(\alpha_1)}{r} = \frac{\sin(\beta_1)}{\rho} \\ \frac{\sin(\alpha_2)}{r} = \frac{\sin(\beta_2)}{\rho} \end{cases} \quad (5-5)$$

式(5-5)中， r 表示位于圆周上发射信号无人机极径；将式(5-4)代入式(5-5)中，并化简得式(5-6)：

$$\begin{cases} \alpha = \arctan \frac{\sin(\alpha_1 - \theta) \sin(\alpha_2) - \sin(\alpha_1) \sin(\alpha_2)}{\sin(\alpha_1) \cos(\alpha_2) + \cos(\alpha_1 - \theta) \sin(\alpha_2)} \\ \rho = \frac{\sin(\alpha_2 + \alpha)}{\sin(\alpha_2)} * r \end{cases} \quad (5-6)$$

式(5-6)中 $(\alpha + \theta)$ 为被动接收信号无人机极角； ρ 表示被动接收信号无人机极径。得到极坐标为 $(\rho, \alpha + \theta)$ 。

模型三：选定的两个发射信号无人之间相隔 1 到 $N(N \in \{2, 3, 4\})$ 个无人机，如：FY01、FY03 或 FY02、FY05。对位于其间的无人机位置进行求解，被动接收信号无人机与发射信号无人机分布如图 5-5 所示：

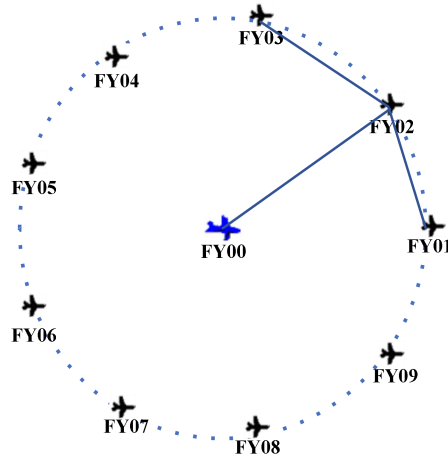


图 5-5 无人机分布图

图 5-5 中被动接收信号无人机未位于理想位置，其极角 $\gamma = \alpha$ 。同模型一中，作连线，构建无人机之间的三角形，角度之间的关系如图 5-6 所示：

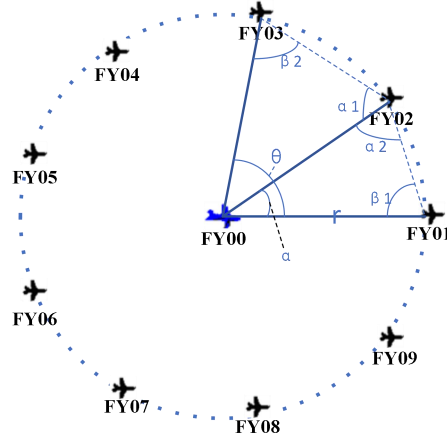


图 5-6 无人机位置关系图

图 5-6 中， $0k$ 代表编号为 $FY0k(k \in \{2, 3, 4\})$ 的无人机以 $00-01$ 之间连线为极轴， $\alpha_1 = \angle 00-0k-01$ ； $\alpha_2 = \angle 00-0k-03$ ； $\beta_1 = \angle 00-01-0k$ ； $\beta_2 = \angle 00-03-0k$ ； $\alpha = \angle 03-00-0k$ ； $\theta = \angle 01-00-03$ 。由三角形内角和为 π ，得出角度之间关系式(5-7)：

$$\begin{cases} \beta_1 = \pi - (\alpha_2 + \alpha) \\ \beta_2 = \pi - [(\theta - \alpha) + \alpha_1] \end{cases} \quad (5-7)$$

利用正弦定理建立出极径与角度之间的关系，如式(5-8)：

$$\begin{cases} \frac{\sin(\alpha_2)}{r} = \frac{\sin(\beta_1)}{\rho} \\ \frac{\sin(\alpha_1)}{r} = \frac{\sin(\beta_2)}{\rho} \end{cases} \quad (5-8)$$

式(5-8)中， r 表示位于圆周上发射信号无人机极径；将式(5-7)代入式(5-8)中，化简得到式(5-9)：

$$\begin{cases} \alpha = \arctan \frac{\sin(\alpha_1 + \theta) \sin(\alpha_2) - \sin(\alpha_1) \sin(\alpha_2)}{\sin(\alpha_1) \cos(\alpha_2) + \cos(\alpha_1 + \theta) \sin(\alpha_2)} \\ \rho = \frac{\sin[(\theta - \alpha) + \alpha_1]}{\sin(\alpha_2)} * r \end{cases} \quad (5-9)$$

式(5-9)中 α 为被动接收信号无人机极角； ρ 表示被动接收信号无人机极径。得到极坐标为 (ρ, α) 。

模型四：选定的两个发射信号无人机之间相隔 0 个无人机，如 $FY01$ 、 $FY02$ 。对被动接收信号无人机位于极角等于 $(2\pi - \alpha)$ 进行求解。无人机分布图如图 5-7 所示：

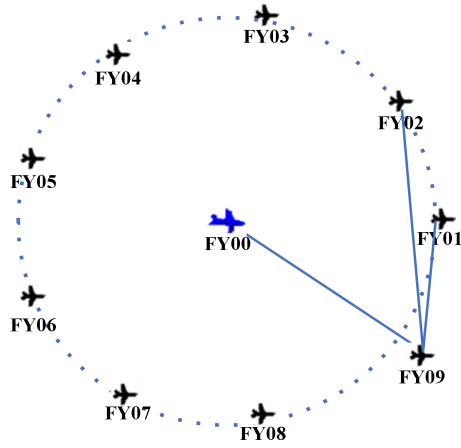


图 5-7 无人机分布图

图 5-7 中，被动接收信号无人机未位于理想位置，其极角 $\gamma=2\pi-\alpha$ 。同模型一中，作连线，构建无人机之间的三角形，角度之间的关系如图 5-8 所示：

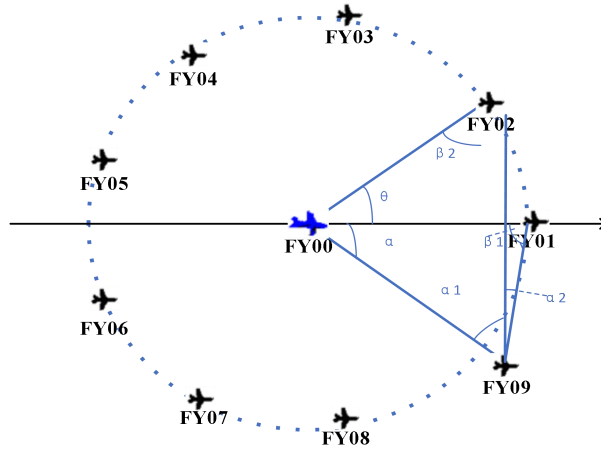


图 5-8 无人机位置关系

由图 5-8 中， $0k$ 代表编号为 $FY0k(k \in \{7, 8, 9\})$ 的无人机以 $00-01$ 之间连线为极轴， $\alpha_1 = \angle 00-0k-01$ ； $\alpha_2 = \angle 01-0k-02$ ； $\beta_1 = \angle 00-01-0k$ ； $\beta_2 = \angle 00-02-0k$ ； $\alpha = \angle 02-00-0k$ ； $\theta = \angle 01-00-02$ 。由三角形内角和为 π ，建立式(5-10)：

$$\begin{cases} \beta_1 = \pi - (\alpha_1 + \alpha_2 + \alpha) \\ \beta_2 = \pi - [(\theta + \alpha) + \alpha_1] \end{cases} \quad (5-10)$$

利用正弦定理建立极径与角度之间的关系式(5-11)：

$$\begin{cases} \frac{\sin(\alpha_1 + \alpha_2)}{r} = \frac{\sin(\beta_1)}{\rho} \\ \frac{\sin(\alpha_1)}{r} = \frac{\sin(\beta_2)}{\rho} \end{cases} \quad (5-11)$$

式(5-11)中, r 表示位于圆周上发射信号无人机极径; 将联立式(5-10)、式(5-11)得到式(5-12):

$$\begin{cases} \alpha = \arctan \frac{[\sin(\alpha_1) - \sin(\alpha_1 + \theta)] \sin(\alpha_1 + \alpha_2)}{\cos(\alpha_1 + \theta) \sin(\alpha_1 + \alpha_2) - \cos(\alpha_1 + \alpha_2) \sin(\alpha_1)} \\ \rho = \frac{\sin[(\theta + \alpha) + \alpha_1]}{\sin(\alpha_1)} * r \end{cases} \quad (5-12)$$

式(5-12), $(2\pi - \alpha)$ 表示被动接收信号无人机极角, ρ 被动接收信号无人机极径。得到极坐标 $(\rho, 2\pi - \alpha)$ 。

5.1.2 第一问模型求解

由题所给条件可知, 以 FY00-FY01 连线为极轴, FY00 无人机为圆心, 其余无人机均匀分布在圆周上, 假设圆周极径为 100m, 则所得无人机的极坐标如表 5-1 所示:

表 5-1 无人机极坐标参数表

无人机编号	极坐标(m, °)
00	(0, 0)
01	(100, 0)
02	(100, 40)
03	(100, 80)
04	(100, 120)
05	(100, 160)
06	(100, 200)
07	(100, 240)
08	(100, 280)
09	(100, 320)

模型一: 已知模型建立在位置偏离的被动接收信号无人机的基础上, 当无人机位于理想情况时, 验证模型的可行性。理想情况无人机分布图如下图 5-9 所示:

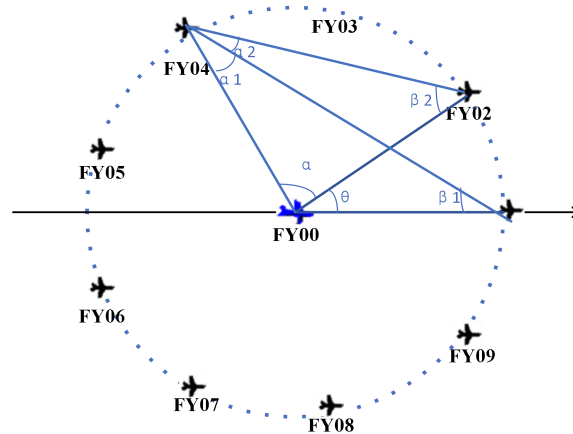


图 5-9 理想情况无人机分布

图 5-9 中，图中各角度值如下表 5-2 所示：

表 5-2 各角度分布表

θ	α	$\alpha 1$	$\alpha 2$
40°	80°	30°	20°

表 2 表示了选用编号 01、编号 02 无人机作为发射信号无人机时，测量编号为 04 无人机位置各角度的角度值。此模型下，理想极坐标($\rho=100$, $\gamma=\alpha+\theta=120^{\circ}$)，将表 2 中数据代入式(5-3)中，解得极径 $\rho=100\text{m}$ ，极角 $\gamma=120^{\circ}$ ，符合表中数据。模型一同样适用于测量编号为 03 和编号为 05 的无人机位置。

模型二：在此模型下，理想情况无人机分布图如下图 5-10 所示：

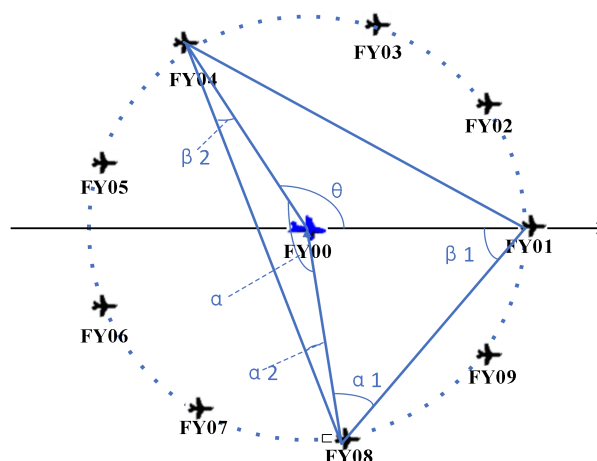


图 5-10 理想情况无人机分布

图 5-10 中，图中各角度值如下表 5-3 所示：

表 5-3 各角度分布表

θ	α	$\alpha 1$	$\alpha 2$
160°	120°	50°	30°

表 3 表示了选用编号 01、编号 04 无人机作为发射信号无人机时，测量编号 08 无人机各角度值。理想极坐标($\rho=100$, $\gamma=\alpha+\theta=280^\circ$)，将表 3 中数据代入式(5-6)中，解得极径 $\rho=100\text{m}$ ，极角 $\gamma=280^\circ$ ，符合表中数据。模型二同样适用测量编号为 06、编号为 07 无人机位置。

模型三：在此模型下，理想情况无人机分布图如下图 5-11 所示：

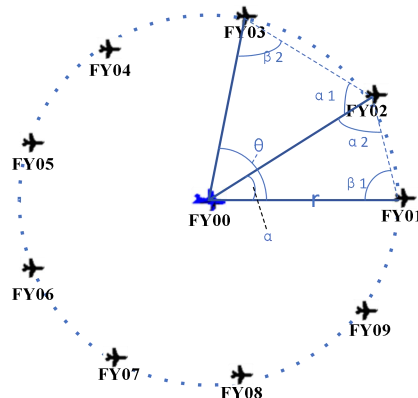


图 5-11 理想情况无人机分布

图 5-11 中，图中各角度值如下表 5-4 所示：

表 5-4 各角度分布表

θ	α	$\alpha 1$	$\alpha 2$
80°	40°	70°	70°

表 4 表示了选用编号 01、编号 03 无人机作为发射信号无人机时，测量编号 02 无人机各角度值。理想极坐标($\rho=100$, $\gamma=40^\circ$)，将表 4 中数据代入式(5-9)中，解得极径 $\rho=100\text{m}$ ，极角 $\gamma=40^\circ$ ，符合表中数据。模型三同样适用当选编号 02、编号 04 时或选用编号 01、编号 04 时，测量处于二者中间的无人机位置。

模型四：在此模型下，理想情况无人机分布图如下图 5-12 所示：

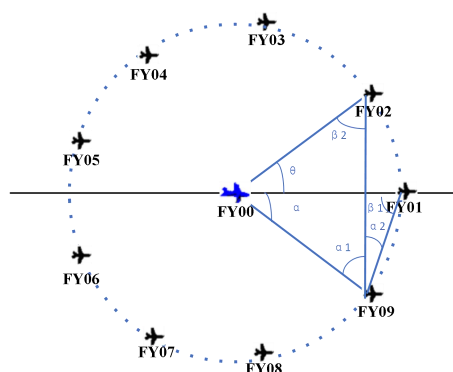


图 5-12 理想情况无人机分布

图 5-12 中，图中各角度值如下表 5-5 所示：

表 5-5 各角度分布表

θ	α	α_1	α_2
40°	40°	50°	20°

表 5 表示了选用编号 01、编号 02 无人机作为发射信号无人机时，测量编号 09 无人机时各角度值。编号 09 无人机理想极坐标($\rho=100$, $\gamma=2\pi-\alpha=40^\circ$)，将表 5 中数据代入式 (5-12)中，解得极径 $\rho=100\text{m}$ ，极角 $\gamma=40^\circ$ ，符合表中数据。模型四同样适用当选用编号 02 无人机、编号 03 无人机作为发射信号无人机时或选用编号 01 无人机、编号 03 无人机时，测量编号 08 无人机位置。

5.2 问题一的模型建立与求解

5.2.1 第二问的模型建立

在第一小问的基础上，由题意已知编号 FY00、FY01 的无人机为无偏差的信号发射无人机，现要从其余的无人机中选取若干架无人机实现有效定位。

首先，考虑在编号 FY00、FY01 两架无人机的基础上增加 1 架无人机用作信号发射无人机，故该待增无人机的选取可以分为以下 4 种情况，如图 5-13 所示。

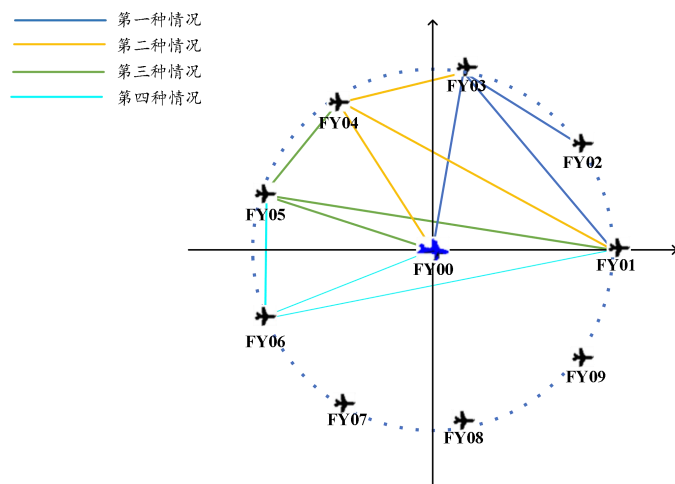


图 5-13 增加 1 架无人机可选择的情况

这四种情况的选取与第一问的四种模型一致，均可在已知三个无偏差的信号发射无人机的位置时，定位待定的无人机坐标。但是，当某一待定位无人机的位置出现偏差，与编号 FY00、FY01 的无人机处于同一直线上时，待定位无人机无法确定信号来自编号 FY00 还是来自 FY01，故无法求出待定无人机的位置坐标。如图 5-14 所示。

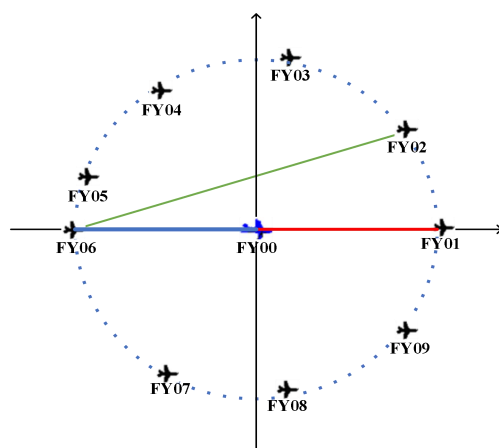


图 5-14 特殊情况下无法求得待定无人机坐标

因此，在编号 FY00、FY01 两架无人机的基础上再增加 2 架无人机作为信号发射无人机。如图 5-15 所示。

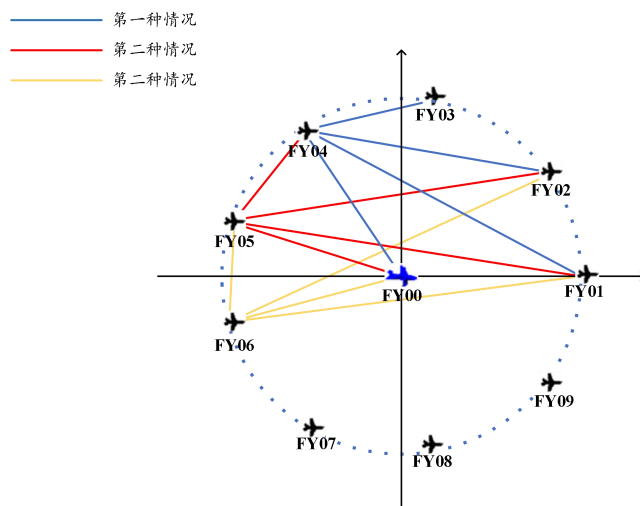


图 5-15 增加 2 架无人机可选择的情况

当未出现特殊情况时，可以建立以下三种模型：

模型一：选定与编号 FY00，FY01 完全相邻的两架无人机 FY02，FY03 作为信号发射无人机。

模型二：选定与编号 FY00，FY01 未完全相邻且间隔 1 个无人机的两架无人机 FY02，FY04 作为信号发射无人机。

模型三：选定与编号 FY00，FY01 未完全相邻且间隔 2 个无人机的两架无人机 FY02，FY05 作为信号发射无人机。

当出现特殊情况时，如图 5-16 所示。

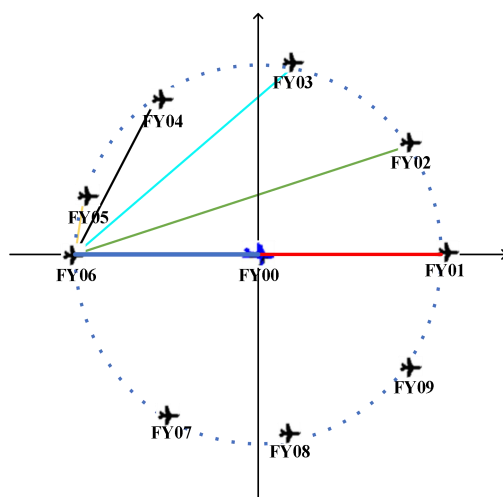


图 5-16 特殊情况下选定 2 个无人机的可能性

可以建立以下三个模型：

模型四：选定与编号 FY00，FY01，FY02，FY03 作为信号发射无人机。

模型五：选定与编号 FY00，FY01，FY02，FY04 作为信号发射无人机。

模型六：选定与编号 FY00，FY01，FY02，FY05 作为信号发射无人机。

5.2.2 第二问的模型求解

由 5.2.1 中建立的一、二、三模型可知，并不需要运用 4 个无偏差信号发射无人机的坐标，故从 4 个无偏差信号发射无人机中选用 3 个即可得到待定目标的位置，与问题一的第一小问相似。再由四、五、六模型可知，编号 FY00，FY01 的无偏差信号发射无人机处于同一直线上，故可看作一个无偏差的信号发射无人机，再结合选定的 2 个无偏差信号发射无人机可构成 3 个无偏差信号发射无人机，通过这 3 个位置信息即可确定待定目标的位置。故再选定 2 个无偏差信号发射无人机即可实现无人机的有效定位。

5.3 问题一第三问的模型建立与求解

5.3.1 第三问的模型建立

无人机在飞行的过程中，其编队机制与鸟类群体的社会活动行为存在一定的相似性，通过对鸟类的生物社会活动系统的分析，发现鸽群的群集运动与无源定位无人机的机理相似，在本问的基础上将鸽群的行为机制引入无人机的编队飞行控制系统中，进行鸽群群体行为的运动控制建模^[1]。

鸽群与绝大多数的生物总群相似，其生物群体中存在严格的层级关系，在每个层级里，处于层级最前方的头鸽有着绝对的领导地位。在第三问中，初始状态下位于圆心的无人机 FY00 是编队的核心所在，从题目的分析可知，多无人机为了避免外界信号的干

扰，都会尽可能的保持信号静默，少向外界的控制系统发射电磁波信号，为了让无人机在控制系统的作用下实现稳定的编队队形，采用鸽群机制的层级关系对无人机进行编号。

设无人机的层级结构的可用有向图用公式（5-13）描述， 顶点集合为第 n 只鸽子，描述如公式（5-14），边集集合为上下层级无人机的领导关系，描述如公式（5-15）。

$$D_n = (V_n, E_n) \quad (5-13)$$

$$V_n = \{v_1, v_2, v_3, \dots, v_n\} \quad (5-14)$$

$$E_n = \{e_1, e_2, e_3, \dots, e_n\} \quad (5-15)$$

对于第三问的模型建立，可以事先定义好无人机之间的层级关系，处于圆心的无人机属于团队的绝对核心。无人机的领导关系用 N_i 表示，如公式（5-16）。

$$N_i = \begin{cases} \emptyset, i=1, \\ 1, i=2, \\ 1, 2, i=3, \\ i-3, i-2, i-1, i=2k, k=\{3, 4, \dots\}, \\ i-4, i-3, i-2, i-1, i=2k, k=\{2, 3, \dots\}. \end{cases} \quad (5-16)$$

FY00 无人机不受任何的无人机的领导约束，在图 5-17 中，矩形框表示的是无人机编队的编号，即是有向图中的顶点集合，指向箭头代表编队中的层级关系。

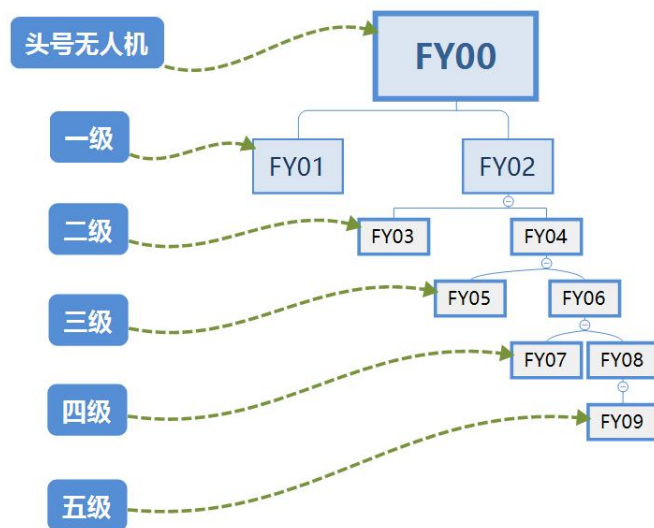


图 5-17 无人机层级关系有向图

从平面空间的编队问题中来说，对于第三问，期望的编队队形是均匀分布在圆周模型上的，基于鸽群生物总群中的领导关系，可使得无人机实现规定状态的编队、稳定的飞行要求。以 FY00 无人机为原点，从第三问给出的真实数据仿真得到如图 5-18。

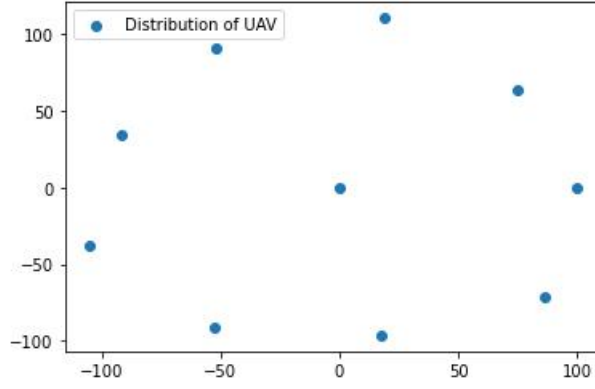


图 5-18 无人机真实数据分布图

为了消除无人机初始状态的误差，使其均匀的移动分布在圆周模型上，建立被接收信号无人机的动力学模型，如公式（5-17）。

$$\begin{cases} X_i = v_i, \\ V_i = v_i - k_i v_i, i = 1, \dots, n. \end{cases} \quad (5-17)$$

其中， X_i 表示无人机在笛卡尔坐标系第*i*个无人机的位置矢量， V_i 表示第*i*个无人机的速度矢量， k_i 为速度的阻尼系数，当 k_i 大于0时，代表无人机的速度衰减增益， X_{ij} 表示无人机*i*与*j*之间的相对欧式距离。

人工势场法是在移动控制机器人中最常采用的一种简单有效的路径规划算法，其主要的基本思想是模拟控制系统的机器人在工作环境中的，构造一个人工创造的势能场景，势中包含有斥力级、引力级，让控制机器人原理不在规定路径的斥力极，靠近目标位置的引力级。在本文中，理想状态下的无人机位置定义为引力级点，尽可能的让无人机根据其余无人机的领导信息调整自身的控制函数，朝向目标方向运动^[3]。势函数的建立如公式(5-18)所示。

$$P_{ij}(\|x_{ij}\|) = \ln \|x_{ij}\|^2 + \frac{d_{ij}^2}{\|x_{ij}\|^2} \quad (5-18)$$

其中， $\|x_{ij}\|$ 表示第*i*个无人机和第*j*个无人机之间的欧氏距离^[2]， $\|d_{ij}\|$ 表示为第*i*个无人机和上级领导层次无人机*j*的期望距离。

在笛卡尔坐标系中，将无人机的运动视为质点运动，且在二维的平面空间内，假设无人机的运动满足高斯分布，对无人机水平方向的速度控制量进行建模，如公式(5-19)所示。

$$u_i = (-Kp \cdot x_{ij} \cdot p_{ij} \cdot x - Kv \cdot (u_i - u_j) \cdot w_1) + ku_i \quad (5-19)$$

式（5-1x）中， K_p 表示为势场的常数因子， K_v 表示为速度的反馈常数因子， K_p ， K_v 均大于 0， w_1 为水平速度控制调节常数因子。

在本问中，假设无人机具备有初始速度，航向角两个回环的自动驾驶器，则可以将无人机在运动过程中的控制系统简化为四态模型，如公式（5-20）。

$$\begin{cases} x_i = V_i \cos(\alpha), \\ y_i = V_i \sin(\alpha), \\ V_i = \frac{1}{t_v}(V_i^c - V_i), \\ \alpha_i = \frac{1}{t_\alpha}(\alpha_i^c - \alpha_i). \end{cases} \quad (5-20)$$

无人机的运动模型建立，需要将无人机控制系统输出状态的数据转化为无人机层级模型的输入数据，解算公式如式（5-21）所示。

$$\begin{cases} V_i^c = t_v(u_1^i \cos(\alpha_i) + u_2^i \sin(\alpha_i)) + V_i, \\ \alpha_i^c = \frac{t_\alpha}{V_i^i}(u_2^i \cos(\alpha_i) + u_1^i \sin(\alpha_i)) + \alpha_i, \\ X_i = (x_i, y_i), \\ v_i = (V_i \cos(\alpha), V_i \sin(\alpha)). \end{cases} \quad (5-21)$$

5.3.2 第三问的模型求解

对第三问题中所给的数据进行仿真，假设有 10 架无人机在二维的笛卡尔坐标系中飞行，其参数如表 5-6 所示。

表 5-6 无人机控制参数设置

参数符号	描述	初始值
t_v	速度时间因子	3s
t_a	无人机航向角时间因子	0.75s
K_p	势场常数因子	120
K_v	速度的反馈常数因子	1
w_1	水平速度控制调节常数因子	0.1

六、结果分析

综上所述，最终无人机的圆周模型拟合轨迹如图 6-1 所示。

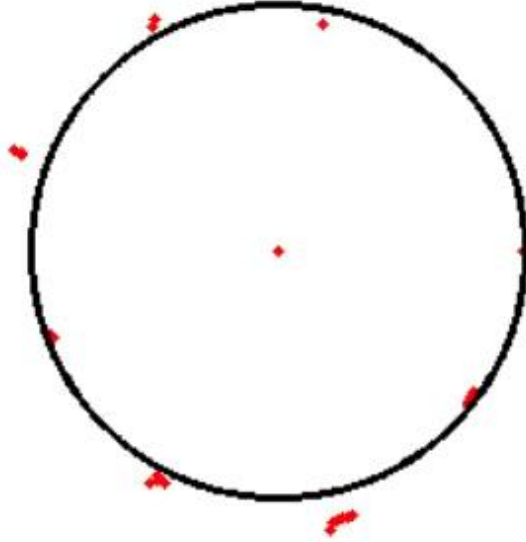


图 6-1 无人机圆周模型拟合轨迹图

七、模型评价及推广

7.1 模型的优点

1.问题一中的第一问，第二问均采用几何方法对圆周模型进行求解，可实现在二维平面的精准定位。

2.基于生物模型中的鸽群模型对无人机的编号进行层次编号，每个层次均有一个领导无人机固定发射信号源，再由临近的两个无人机进行辅助信号发送，解决无人机编队过程中信号干扰的问题。

3.模型具有良好的鲁棒性，不仅适用于圆周模型，也实用于锥形模型，只需要更改好无人机之间的间距与模型航向角，即可完成仿真模型。

7.2 模型的缺点

1.在实际场景下无人机的编队任务中，不仅是在二维的仿真模型下进行，而是在复杂的三维空间中运行，需要考虑到无人机的高度信息等，并且在三维空间内的航向角也由原来的一个航向角增加到三个航向角，分别是翻滚角，偏航角，俯仰角。在本文的模型建立中并没有考虑到三维空间的仿真模型情况，在后续的无人机编队模型中加以改进。

2. 鸽群模型的层次模型依照"就近原则", 并非一个优解, 在后续模型改进中针对不同的层次模型进行仿真模拟, 找到最优解。

3. 人工势场模型的仿真结果不稳定, 无人机的飞行姿态不佳, 后续考虑向模型的控制系统加入 PID 调制, 使用卡尔曼滤波对无人机的飞行位置进行预测, 以求达到更稳定的飞行姿态。

7.3 模型的推广

模型可对与圆周模型的无人机编队调整有一定的帮助, 并且适用于其他的模型。模型使用了多种的几何方法进行求解, 并且利用的生物种群模型中的鸽群模型进行模型仿真, 均取得良好的效果。

参考文献

- [1] 邱华鑫, 段海滨, 范彦铭. 基于鸽群行为机制的多无人机自主编队[J]. 控制理论与应用, 2015, 32(10): 1298-1304.
- [2] 邹先雄. 无人机目标无源定位方法研究[D]. 电子科技大学, 2018.
- [3] 吴芬. 单无人机纯角度目标被动定位与跟踪问题研究[D]. 南京理工大学, 2017.

附录

一、支撑材料列表

1. porint_1.py

2.san_2.py

3. pigeon_3.py

二、源代码

问题一第三问的源代码 1:

文件名: porint_1.py

```
import numpy as np
import cv2 as cv
import matplotlib.pyplot as plt
import matplotlib.patches as mpatches
# table 3 极坐标
img = np.zeros((600,600,3))
point = np.array(
    [
        [0, 0, 0, 0, 0, 0],
        [100, 0, 100, 0, 1, 0],
        [98, 40.10, 100, 40, 2, 0],
        [112, 80.21, 100, 80, 3, 0],
        [105, 119.75, 100, 120, 4, 0],
        [98, 159.86, 100, 160, 5, 0],
        [112, 199.96, 100, 200, 6, 0],
        [105, 240.07, 100, 240, 7, 0],
        [98, 280.17, 100, 280, 8, 0],
        [112, 320.28, 100, 320, 9, 0]
    ])
point_ = np.zeros((point.shape[0],2))
# x = rcos%,y=rsin%
```

```

#  $x^2 + y^2 = r^2$ ,  $y/x = \tan\theta$ 
for i in range(point.shape[0]):
    point_[i,0] = point[i,0] * np.cos(np.deg2rad(point[i,1]))
    point_[i,1] = point[i,0] * np.sin(np.deg2rad(point[i,1]))
p_ = []
for i in range(1,5):
    x_ = 100*np.cos(np.deg2rad((i-1)*40))
    y_ = 100*np.sin(np.deg2rad((i-1)*40))
    p_.append([x_,y_])

plt.plot(x_, y_ )
plt.scatter(point_[:,0], point_[:,1], lw=1, label='Distribution of UAV')
plt.legend()
plt.show()

```

问题一第二问的源代码 2:

文件名: san_2.py

```

import sympy
import numpy as np

r = sympy.symbols('r')
a = sympy.symbols('a')
x0, y0 = sympy.symbols('x0 y0')
x1, y1 = sympy.symbols('x1 y1')
x2, y2 = sympy.symbols('x2 y2')
x3, y3 = sympy.symbols('x3 y3')
x01, x12, x13 = sympy.symbols('x01 x12 x13')
k12, k10, k_ = sympy.symbols('k12 k10 k_')
s02, p02, s10, p10 = sympy.symbols('s02 p02 s10 p10')
x0, y0 = 0, 0
x1, y1 = 100, 0

```

```

x2, y2 = 76.6044, 64.2788
# x3, y3 = 17.3648, 98.4808
r = sympy.sqrt((x0 - x1)**2 + (y0 - y1)**2)
k10 = x1**2 + y1**2 - 2 * (x1 * x0 + y1 * y0)
k12 = x1**2 + y1**2 - 2 * (x2 * x1 + y2 * y1)
k_ = 2 * r * sympy.sin(a) - (x1**2 + y1**2)
s02 = x0 - x2
s10 = x1 - x2
p02 = y0 + y2
p10 = y1 + y0
y3 = ((s10 * (k12 - k10) - s02 * (k12 - k_))/2) / (p02 * s10 - p10 * s02)
x3 = (((k12 - k10) / 2) - y3 * p10) / s02
print("x3:",x3)
print("=====")
print("y3:",y3)

```

问题一第三问的源代码 3:

文件名: pigeon_3.py

```

import numpy as np
import cv2 as cv
import matplotlib.pyplot as plt
import matplotlib.patches as mpatches
# table 3 极坐标

n = 10
v_0 = np.random.rand(10)*5
desired_dis = np.zeros((n,n))

def polar(r0,th0):
    #极坐标转换为直角坐标
    x_ = r0*np.cos(np.deg2rad((th0)))

```



```

y_ = r0*np.sin(np.deg2rad((th0)))
return x_,y_

def dis(x0,y0,x1,y1):
    #欧氏距离求解
    return np.sqrt(np.square(x0 - x1) + np.square(y0 - y1))

def p_ij(point,i,j):
    #势函数 chan 算法
    r0 = point[int(point[i,4]),0]
    th0 = point[int(point[i,4]),1]
    r1 = point[int(point[j,4]),0]
    th1 = point[int(point[j,4]),1]

    x0, y0 = polar(r0, th0)
    x1, y1 = polar(r1, th1)

    x_ij = dis(x0, y0, x1, y1)

    r00 = point[int(point[j,5]),0]
    th00 = point[int(point[j,5]),0]

    x00, y00 = polar(r00, th00)
    d_ij = desired_dis[j,int(point[j,5])]

    return np.log(np.square(x_ij)) + np.square(d_ij) / np.square(x_ij)

def uav(point,i,tv,ts,vc,sc):
    vi = point[int(point[i,4]),6]
    s = point[int(point[i,4]),1]

```

```

point[int(point[i,4]),7] = vi * np.cos(np.deg2rad(s)) + point[int(point[i,4]),0]
point[int(point[i,4]),8] = vi * np.sin(np.deg2rad(s)) + point[int(point[i,4]),0]
point[int(point[i,4]),6] = (1 / tv) * (vc - vi)
point[int(point[i,4]),1] = (1 / ts) * (sc - s) + s
x = point[int(point[i,4]),7]
y = point[int(point[i,4]),8]
v = point[int(point[i,4]),6]
s = point[int(point[i,4]),1]
return x,y,v,s

```

```
def solve_uav(point,i,tv,ts):
```

```

    #鸽群模型转 uav
    s = point[int(point[i,4]),1]
    vi = point[int(point[i,4]),6]
    ui_1 = vi * np.cos(np.deg2rad(point[int(point[i,4]),9]))
    ui_2 = vi * np.sin(np.deg2rad(point[int(point[i,4]),10]))
    point[int(point[i,4]),9] = ui_1
    point[int(point[i,4]),10] = ui_2
    vc = (tv * (ui_1 * np.cos(np.deg2rad(s)) + ui_2 * np.sin(np.deg2rad(s)))) + vi
    sc = ((ts / vi) * (ui_2 * np.cos(np.deg2rad(s)) - ui_1 * np.sin(np.deg2rad(s)))) + s + 0.01
    return vc,sc

```

```
def ui_12(point,i,kp,kv,pij,w1):
```

```

    xi_1 = point[int(point[i,4]),7]
    xi_2 = point[int(point[i,4]),8]
    ui_1 = point[int(point[i,4]),9]
    ui_2 = point[int(point[i,4]),10]
    ui_0 = point[int(point[0,4]),10]
    vi_1 = (-kp * xi_1 * pij * np.random.randn(1)[0] - kv * ui_1 - (ui_1 - ui_0) * w1) +

```

```
ui_1
```

```

    vi_2 = (-kp * xi_2 * pij * np.random.randn(1)[0] - kv * ui_2 - (ui_2 - ui_0) * w1) + ui_2

```

```

    point[int(point[i,4]),9] = vi_1
    point[int(point[i,4]),10] = vi_2
    # return vi_1,vi_2

tv = 3
ts = 0.75
kv = 1
kp = 120
w1 = 0.1
epoch = 1000
point = np.array(
    [
        [0, 0, 0, 0, 0, 0, v_0[0], 0, 0, 0, 0],
        [100, 0, 100, 0, 1, 0, v_0[1], 0, 0, 0, 0],
        [98, 40.10, 100, 40, 2, 0, v_0[2], 0, 0, 0, 0],
        [112, 80.21, 100, 80, 3, 0, v_0[3], 0, 0, 0, 0],
        [105, 119.75, 100, 120, 4, 0, v_0[4], 0, 0, 0, 0],
        [98, 159.86, 100, 160, 5, 0, v_0[5], 0, 0, 0, 0],
        [112, 199.96, 100, 200, 6, 0, v_0[6], 0, 0, 0, 0],
        [105, 240.07, 100, 240, 7, 0, v_0[7], 0, 0, 0, 0],
        [98, 280.17, 100, 280, 8, 0, v_0[8], 0, 0, 0, 0],
        [112, 320.28, 100, 320, 9, 0, v_0[9], 0, 0, 0, 0]
    ])
while epoch:
    img = np.zeros((600,600,3))
    for i in range(desired_dis.shape[0]):
        for j in range(desired_dis.shape[0]):
            x0 = point[i,0]
            y0 = point[i,1]
            x1 = point[i,2]
            y1 = point[i,3]

```

```

        desired_dis[i,j] = dis(x0,y0,x1,y1)
for i in range(1,point.shape[0]):
    for j in range(i,point.shape[0]):
        x_, y_ = polar(point[int(point[i,4]),8],point[int(point[i,4]),1])
        x0, y0 = polar(point[0,0],point[0,1])
        cv.circle(img, (int(x_ + 200),int(y_ + 200)), 2, (0,0,255),-1)
        cv.circle(img, (int(x0 + 200),int(y0 + 200)), 100, (255,255,255),2)
        cv.imshow("not people air", img)
        cv.waitKey(50)
        pij = p_ij(point, i, j)
        vc,sc = solve_uav(point, i, tv, ts)
        x,y,v,s = uav(point,i,tv,ts,vc,sc)
        # ui_12(point, i, kp, kv, pij, w1)
    print(vc,sc)
epoch -= 1
cv.imshow("not people air", img)

```