

无人机协同避障最优航迹规划

摘要

本问题构建了一个两架无人机协同避障的场景,要求在避障的同时,兼顾“时间最少”与“不能碰面”两个条件,给出最优的航迹。

针对问题一,要求“先到达的无人机用时最少”。本文将问题转化为只有一架无人机需要越障,找出此时用时最短的航线。结合题目其他的约束条件,给出了一组符合题目要求的两架无人机的航迹。

针对问题二,要求“后到达的无人机用时最少”。本文深度挖掘题给几何条件,通过几何分析将位于两个半平面的无人机航线整合到一个半平面上研究。由此,题给约束条件发生变化。在新的约束条件下,找出用时最少的临界条件,得到了航线补偿长度的普遍计算方法及航线补偿方案一。

针对问题三,基站 B 距离障碍圆圆心距离不断变化的情况,本文提出了另外两种航线补偿方案。由此实现了对任意补偿长度的精确补偿。最后,给出了轨迹分配的方法。利用遍历取值,将问题三拆分为多个问题二进行研究。最后,就不同的航线补偿长度,给出了航迹设计方案。

针对问题四,B 无人机速率不断变化的情况,通过对两无人机最短路程图像的分析,区分开碰面与不碰面的情况,并就碰面情况下的不同航线补偿长度给出了航迹设计方案。

针对问题五,本文综合前四问,分左右两个半平面,做出航线补偿长度随 B 基站距障碍圆圆心距离 S_B 及 B 无人机飞行速率 V_B 的图像。通过对待补偿航线长度 ΔS 的分析,给出了碰面与不碰面对应的区间以及碰面情况下,不同航线补偿长度所采用的方案。

关键词: 协同避障 几何分析 航线补偿 航迹设计

目录

1	符号说明	3
2	问题提出	3
3	模型假设	4
4	本文讨论的基本情景	4
5	问题一的分析	5
6	问题一模型的建立与求解	5
6.1	只有一架无人机的最短航线	5
6.2	问题一的求解	7
7	问题二分析	9
8	问题二模型的建立与求解	10
8.1	几何推导及分析	10
8.2	问题二的求解	12
8.2.1	航程补偿方案一	13
9	问题三分析	14
10	问题三模型的建立与求解	15
10.1	问题三模型的建立	15
10.1.1	航线补偿方案二	15
10.1.2	航线补偿方案三	15
10.1.3	补偿方案二、三的伪代码	16
10.1.4	轨迹分配	18
10.2	问题三的求解	19
11	问题四分析	20
12	问题四的求解	20
13	问题五分析	21
14	问题五的求解	21
A	function_singleLoop.m	25
B	newCircumPath_v3.m	29

1 符号说明

符号	备注	符号	备注
S_A	基站 A 到障碍圆圆心距离	S_B	基站 B 到障碍圆圆心距离
V_A	无人机 A 匀速飞行速率	V_B	无人机 B 匀速飞行速率
S_F	先到达无人机的飞行距离	S_L	后到达无人机的飞行距离
S_D	两无人机飞行距离差	R	障碍圆半径
ΔS	航线补偿长度		

2 问题提出

本题首先建立了一个半径为 500m 的障碍圆，在其两边直径的延长线上设置有两个无人机站点。五个约束条件如下：

1. A 站距离障碍圆圆心 1km，B 站距离圆心 3.5km。
2. 两架飞机均以恒定速率 10m/s 分别从 A，B 两站同时出发去对方站点执行任务。
3. 无人机在飞行过程中必须避开障碍圆。
4. 两架无人机不得碰面。
5. 无人机的转弯半径不小于 30m。

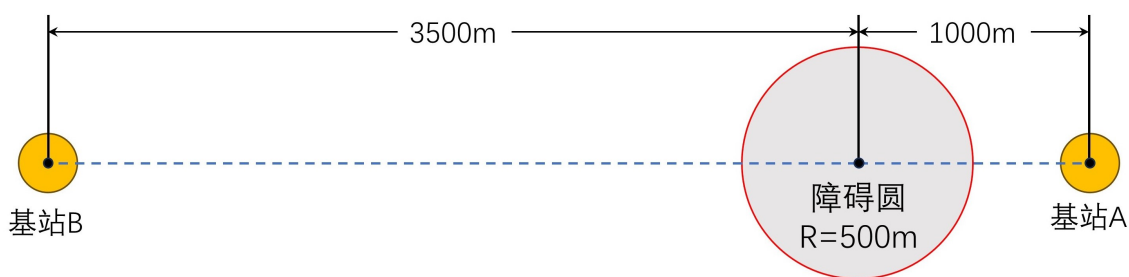


图 1: 题目简图

问题一要求在“先到达的无人机用时最少”的条件下, 给出两架无人机的最优飞行轨迹方案。

问题二要求在“后达到的无人机用时最少”的条件下,给出两架无人机的最优飞行轨迹方案。

问题三要求在“B 站点到圆心的距离在 $[0.5,10]$ km 变化 (其他参数保持不变)”时,说明问题一和问题二中最优航迹的变化。

问题四要求在“B 机的恒定速率在 $[10,30]$ m/s 内变化 (其他参数保持不变) 时”,说明问题一和问题二中最优航迹的变化。

问题五要求在“B 机的恒定速率在 $[10,30]$ m/s 内变化、B 站点到圆心的距离在 $[0.5,10]$ km 内变化 (其他参数保持不变) 时”,说明问题二中最优航迹的变化。

3 模型假设

1. 将飞机视为质点,不考虑其自身形状及体积。
2. A, B 两基站简化为分别距离障碍圆圆心 1km 和 3.5km 的点。
3. 不考虑转弯过程中飞机可能发生的轻微速率改变,认为飞机始终保持匀速率飞行。
4. 飞机可以紧贴障碍圆边缘弧线飞行,也即航线可以与障碍圆边缘弧线重叠。
5. 不考虑飞机能耗问题,认为飞机全程能量充足。

4 本文讨论的基本情景

题目中要求两无人机在飞行过程中不碰面,简单分析可以知道,下面两种飞行状况是符合题意的。

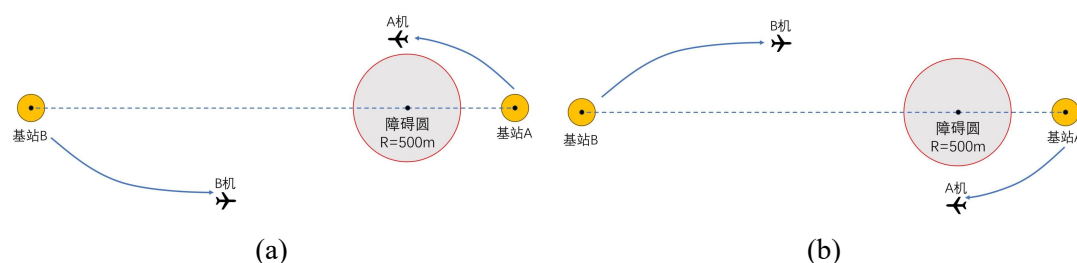


图 2: 两种飞行情况

(a): A 无人机的航道位于上半平面,B 无人机的航道位于下半平面。

(b): A 无人机的航道位于下半平面,B 无人机的航道位于上半平面。

a,b 两种航线实质上是相互对称的, 考虑一种即可。本文选择 d 种航线展开讨论, 也即 A 无人机的航道位于下半平面,B 无人机的航道位于上半平面。

5 问题一的分析

题目一要求“先到达的无人机用时最少”时, 给出两架无人机的飞行航迹方案。

不难知道, 一架无人机的航线确定之后, 平面上一定可以找到满足题目条件的另一架无人机的航线。针对问题一, 本文首先不考虑后到达的无人机, 优先保证先到达的无人机用时最短。这种情形下, 问题就变成了只有一架无人机需要跨越障碍圆, 找出它的最短航线。把这条最短航线作为先到达无人机的航线, 再结合题目约束条件, 就可以确定出后到达无人机的航线。

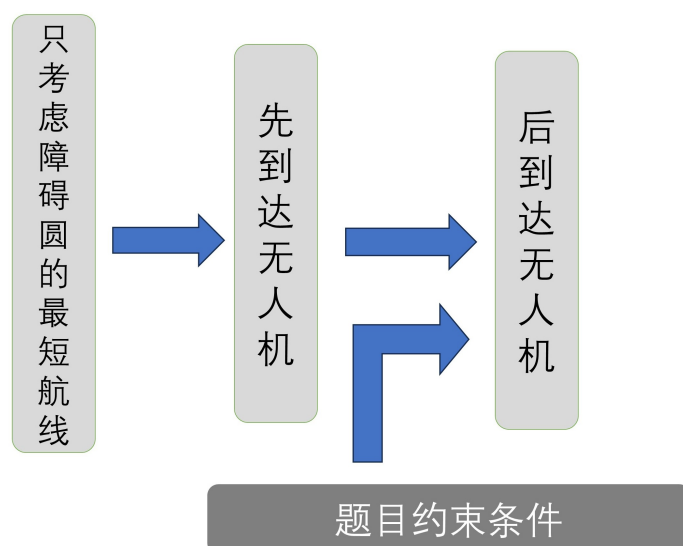


图 3: 问题一分析

6 问题一模型的建立与求解

6.1 只有一架无人机的最短航线

本部分给出 A、B 两基站之间, 在只有一架无人机越障时的最短航线。

对于本问最短航线的证明, 由于本题在拓扑关系上具有对称性, 所以我们只需要分析出 $\frac{1}{4}$ 平面中最短航线的特征就可以推广至整个图形。不失一般性, 选取了三种具有代表性的航线作比较, 分别记为 a,b,c 航线。

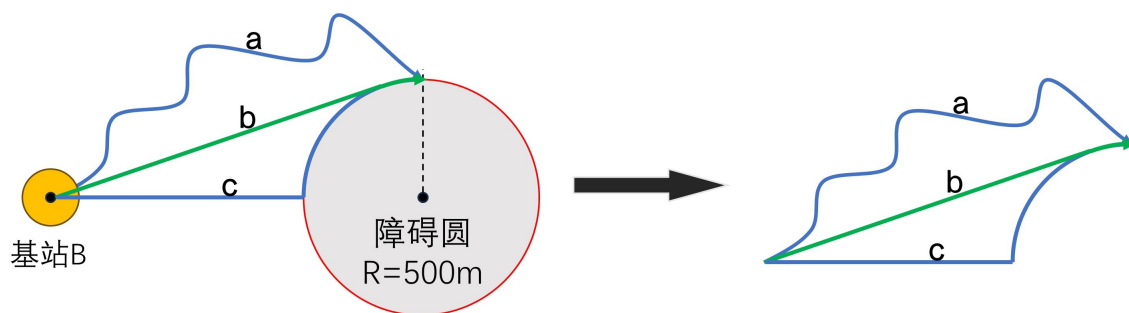


图 4: 示例航线

a 航线: a 航线是一段连接 B 点与障碍圆上顶点的自由曲线。

b 航线: b 航线由基站 B 到障碍圆的切线及切点到障碍圆上顶点的一段圆弧共同构成。

c 航线: c 航线是由基站 B 沿 X 轴正向到达障碍圆边缘及 $\frac{1}{4}$ 障碍圆圆周构成。

a 曲线是所有高于 b 曲线集合的代表;c 曲线是所有低于 b 曲线的集合的代表。容易知道,a 航线一定比 b 航线更长。本文比较的重点集中在 b 和 c 航线的长短。b,c 两条曲线如图 4。

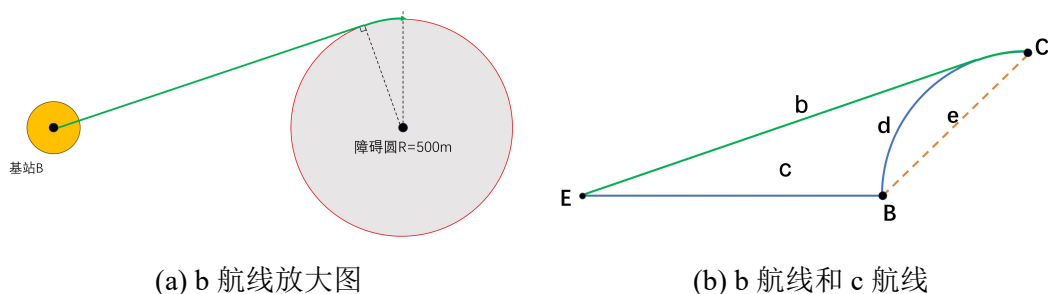


图 5: b 航线和 c 航线示意图

在图 4(b) 中, 给出 b 航线长度小于 c 航线的证明。为了便于证明, 认为曲线 b 近似等于直线 EC。

在三角形 BCE 中,

$$EB + BC > EC \quad (1)$$

$$d > BC \quad (2)$$

$$EB + d > EC \quad (3)$$

也就是, b 航线的长度小于 c 航线, 也即 b 航线为本问的最短航线。

$\frac{1}{4}$ 平面上最短航线的特征: 基站点向障碍圆做切线, 切线段和由切点到障碍圆上顶点的圆弧组成的曲线。

推广到整个上半平面, 可以得到一条本问中 A/B 无人机的到达对方基站的最短航线。

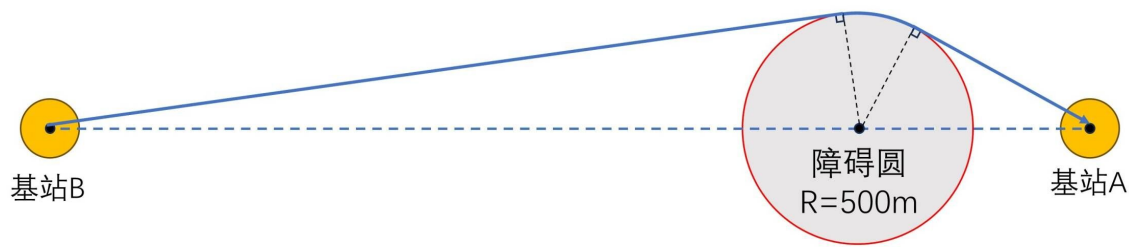


图 6: 第一问中的最短航线

最短航线的长度计算公式为:

$$S_0 = \sqrt{S_B^2 - R^2} + \sqrt{S_A^2 - R^2} + R \cdot \left(\arcsin \frac{R}{S_A} + \arcsin \frac{R}{S_B} \right) \cdot \frac{\pi}{180} \quad (4)$$

注释

为了避免引起歧义, 在第二至五问中, 本文将这种只考虑障碍圆的最短航线称为“极限轨迹”。

6.2 问题一的求解

首先先给出本问中两架无人机各自的最短航线。B 无人机的最短航线记作 S_b ; A 无人机的最短航线记作 S_a 。

在本问中, 如果让 A 无人机航线为最短航线, 由于基站 A 距离障碍圆较近, 很快就跨过了障碍圆, 要在满足题目五个约束条件的基础上规划 B 无人机航线, 难度较大。所以, 本文选取 B 无人机航线为最短航线, A 无人机航线在满足题目要求的情况下, 配合 B 无人机。

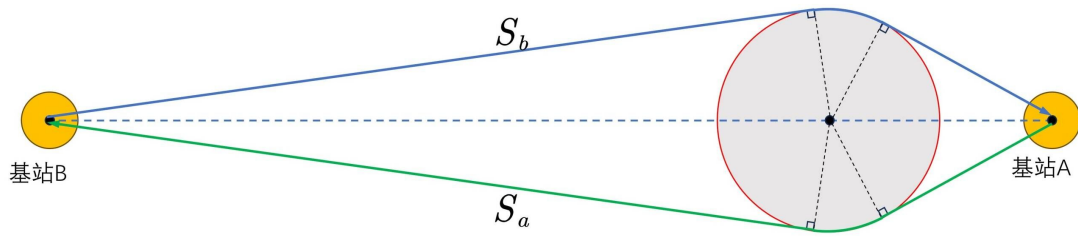


图 7: 本问中两无人机各自的最短航线

注释

这里的分析为第二问中选取 B 无人机航线为极限轨迹埋下了伏笔。

根据后续的模型和计算, 这里直接给出一组符合题目条件的 A、B 无人机航线:
具体路径:

B 无人机: 第一问的最短航线。

A 无人机: 第一问的最短航线 +6 个半径为 30m 的整圆 +1 个半径为 42.7m 的整圆。

下面, 集中给出本问各种结论性的表达式。

无人机 B 的飞行距离:

$$S_B = S_0 \quad (5)$$

无人机 B 的飞行时间:

$$T_B = \frac{S_B}{V_B} \quad (6)$$

无人机 A 的飞行距离:

$$S_A = S_B + 6 \cdot 2\pi \cdot 30 + 2\pi \cdot 42.7 \quad (7)$$

无人机 A 的飞行时间:

$$T_A = \frac{S_A}{V_A} \quad (8)$$

给出此种情形下两架无人机各自的飞行距离和飞行时间:

通过仿真分析验证了本部分的正确性。

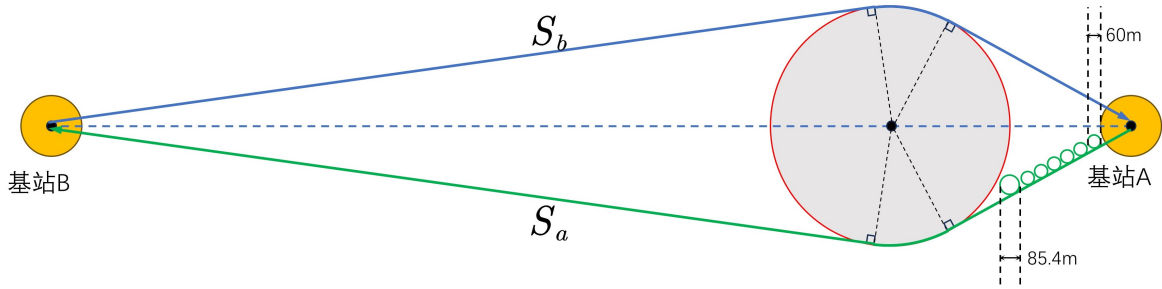


图 8: 本问中两无人机各自的具体路径

表 1: 问题一情形下两架无人机各自的飞行距离及时间

无人机名称	飞行距离	飞行时间
A 无人机	6062.1778m	606.22s
B 无人机	4663.6002m	466.36s

7 问题二分析

问题二要求在“后到达的无人机用时最少”条件下,给出两架无人机的飞行航迹方案。

$$S_L = S_F + S_D \quad (9)$$

其中, S_L 代表后到达无人机的飞行距离, S_F 代表先到达无人机的飞行距离, S_D 代表两架飞机飞行距离之差。

本问的求解思路:

$$\min(S_L) = \min(S_F) + \min(S_D) \quad (10)$$

由问题一可知,应该选取 B 无人机航线为极限轨迹,来达到最小的 S_F 。

按照前文预设的基本情景,两架无人机在不同的半平面内运行,这考虑起来是复杂困难的。如果能将两架无人机的运行状态整合到一个半平面上,分析的难度就会大大下降。

首先,为了达到“后到达的无人机用时最少”这个目标,自然希望两架无人机的位置关系能处于“刚好不碰面”这种临界条件下。不难分析得出,这种临界条件对应于两无人机连线与障碍圆恰好相切。本部分从切线入手,通过对几何条件的深度分析,将两个半平面中的运动整合到一个半平面内,建立全新的约束条件,再次进行求解。

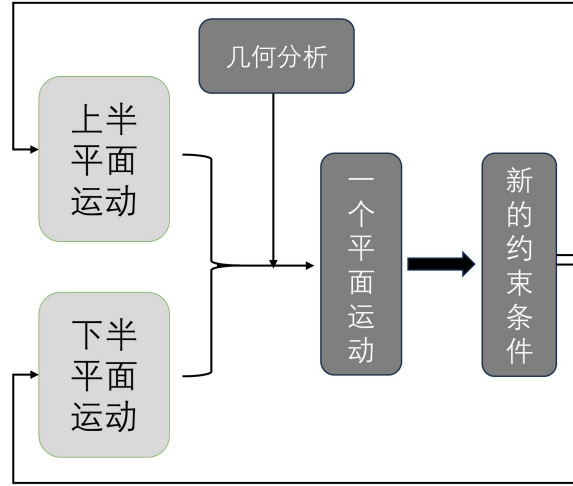


图 9: 问题二分析

8 问题二模型的建立与求解

8.1 几何推导及分析

本文将 A、B 两架无人机的运行模式分为两类：左半平面碰面、右半平面碰面。分别做运动示意图。

如图, B_oC 、 B_oF 、 $B_1B'_1$ 分别切 $\odot O$ 于点 C 、 F 、 H , 障碍圆半径为 r 。本部分对两种情况下的六个变量的表达式进行推导。 f_{A1} 、 f_{B1} 、 $f_{B1'}$ 、 f_{A2} 、 f_{B2} 、 $f_{A2'}$ 。

f_{A1} : A1 点从图 10(a) 中 A0 点出发, f_{A1} 代表 A1 走过的路程长度。

f_{B1} : B1 点从图 10(a) 中 B0 点出发, f_{B1} 代表 B1 走过的路程长度。

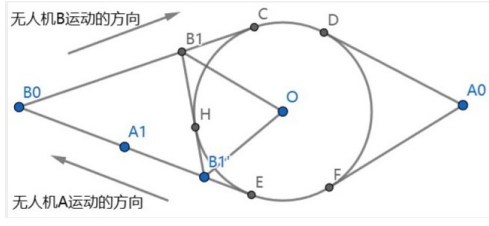
$f_{B1'}$: B1' 点是 B1 点通过切线关系在下半平面极限轨迹上的映射。 $f_{B1'}$ 代表其从 E 点出发走过的路程。

f_{A2} : A2 点从图 10(b) 中 A0 点出发, f_{A2} 代表 A2 点从 A0 点出发走过的路程。

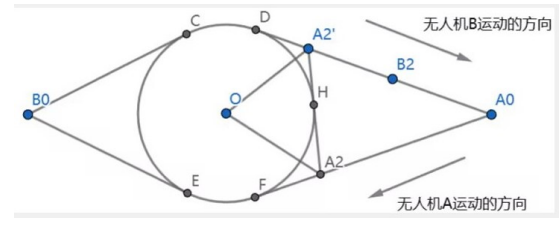
f_{B2} : B2 点从图 10(b) 中 B0 点出发, f_{B2} 代表 B2 点从 B0 点出发走过的路程。

$f_{A2'}$: A2' 点是 A2 点通过切线关系在上半平面极限轨迹上的映射。 $f_{A2'}$ 代表其从 D 点出发走过的路程。

$$f_{A1} = V_A \cdot t = V_A \cdot t_0 - \left| \widehat{EF} \right| - \left| \overrightarrow{A_0F} \right| = V_A \cdot t_0 - d_1 \quad (11)$$



(a) 左半面碰面示意图



(b) 右半面碰面示意图

图 10: 两种运动示意图

$$f_{B1} = V_B \cdot t \quad (12)$$

下面主要推导 $f_{B1'}$ 的表达式。

容易知道: B_1O 平分 $\angle COH$, $B_1'O$ 平分 $\angle HOE$

$$\angle B_1OB_1' = \frac{1}{2}\angle COE = \angle B_0OE \quad (13)$$

$$\angle B_1B_0O = \angle B_1'OE \quad (14)$$

$$f_{B1'} = |EB_1'| = r \cdot \tan \angle B_1'OE \quad (15)$$

又因为:

$$\angle B_1'OE = \angle B_1B_0O \quad (16)$$

由余弦定理:

$$\cos \angle B_1'OB_0 = \frac{\overline{B_1'O}^2 + \overline{B_0O}^2 - 2\overline{B_0O} \cdot \overline{B_1'O}}{2 \times \overline{B_0O} \times \overline{B_1'O}} \quad (17)$$

由正弦定理:

$$\frac{\overline{B_1O}}{\sin \angle B_1B_0O} = \frac{\overline{B_0B_1}}{\sin \angle B_1OB_0} = \frac{f_{B1}}{\sin \angle B_1OB_0} \quad (18)$$

综上, $f_{B1'}$ 可以表示为 f_{B1} 的函数。

$$f_{B1'} = g_1(f_{B1}) \quad (19)$$

同理:

$$f_{A2} = V_A \cdot t_0 \quad (20)$$

$$f_{B2} = V_B \cdot t_0 - \left| \overrightarrow{B_o\hat{C}} \right| - \left| \widehat{CD} \right| = V_B \cdot t - d_2 \quad (21)$$

$$f_{B2'} = g_2(f_{B2}) \quad (22)$$

由上述推导中式 19,22 可知, 无人机在一个半平面上的航行可以通过几何关系映射到另一个半平面上。如此, 两个半平面上无人机的飞行关系就可以整合到一个半平面上。碰不碰面的约束条件就转变为一架无人机能不能追上另一架无人机的映射。

本问中 B 无人机映射的函数表达式:

$$f_{B1'} = R \cdot \tan \left(\arcsin \frac{R}{S_B} \right) \quad (23)$$

注释

针对本问, 两架无人机一定会在左半平面碰面。无论在左半平面碰面还是在右半平面碰面, 由于对称性, 不难分析出, 通过几何关系将一架无人机的真实航线映射整合到另一架无人机的航线上是可行的。

8.2 问题二的求解

图11中, 红色曲线为 B 无人机在下半平面映射的路程随时间图像。蓝色直线为 A 无人机在下半平面的路程随时间图像。

符合题目的图像应该有以下约束:

1. A 无人机的函数图像不能高于 B 无人机在下半平面映射的函数图像。
2. A 无人机的函数图像应尽可能高, 使得 Y 轴截距离终点距离尽可能小。

为了满足这两个约束下, 需要对 A 无人机的函数图像做上下平移, 当 A 无人机的图像与 B 无人机的图像恰好相切时, 向下平移的距离就是所求的两架无人机飞行距离最小的差。

如果能使得 A 无人机的飞行距离至少增加这个差值, 就能使得 A、B 两架无人机满足“不碰面”的约束条件。无人机 A 用时最短的情况, 也就是其航线刚好精确地增加这个最小的差值。本文认为, 此时无人机 A 的航线就是符合题目要求的“后达到的无人机用时最短”。

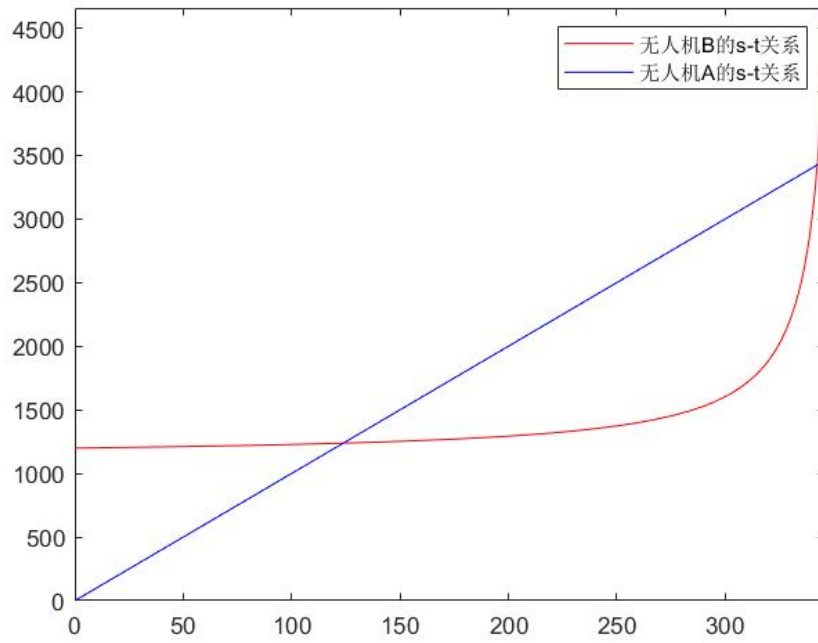


图 11: B 无人机映射与 A 无人机路程随时间变化图像

8.2.1 航程补偿方案一

上文中提到, 航线不是极限轨迹的无人机, 需要补一段航程。如何补偿这段航程, 本部分提出转整圆的方法, 作为航程补偿的第一种方法, 来调节本问中后达到无人机的航程。

航程补偿方法一:

$$\Delta s = \begin{cases} n \cdot 2 \cdot \pi \cdot 30 + 2 \cdot \pi \cdot r_1 & \Delta s > 2 \cdot 2\pi \cdot 30 \\ 2 \cdot \pi \cdot r_2 & \Delta 2\pi \cdot 30 < s < 2 \cdot 2\pi \cdot 30 \end{cases}$$

其中, 要求 n 尽可能大, r_1 和 r_2 均大于 30m。

航线补偿方案一伪代码

IF $\Delta s > 2 \cdot 2\pi \cdot 30$

将待补偿航线长度拆分为 n 个半径 30m 的整圆, 一个半径大于 30m 的整圆, n 尽可能大。

ELSE $2\pi \cdot 30 < \Delta s < 2 \cdot 2\pi \cdot 30$

将待补偿的航线长度整合为一个圆, 其周长为待补偿航线长度。

举例

假设现在需要补偿的航线长度有 2.7 个半径 30m 的圆周长, 那么第一种航线补偿方案会把它拆分成一个半径 30m 的整圆周长和一个周长等于 $1.7 \times 2 \times \pi \times 30$ 的整圆。假设现在需要补偿的航线长度有 1.7 个半径 30 的圆周长, 那么第一种航线补偿方案会把它处理为一个一个周长等于 $1.7 \times 2 \times \pi \times 30$ 的圆。

注释

本文共有三种航线补偿方案, 此处为第一种航线补偿方案。它主要针对待补偿航线长度大于一个半径 30m 整圆周长的情况。关于待补偿航线长度小于一个半径 30m 整圆周长的情况, 后续的问题求解将陆续提出。

利用上述方法对后到达的无人机进行航线补偿, 最终得到后到达的无人机需要补偿 6 个半径 30m 的整圆和一个半径为 42.7m 的整圆。

具体路径:

B 无人机: 第一问的最短航线。

A 无人机: 第一问的最短航线 +6 个半径为 30m 的整圆 +1 个半径为 42.7m 的整圆。

表 2: 问题二最优航迹两架无人机各自的飞行距离及时间

无人机名称	飞行距离	飞行时间
A 无人机	6062.1778m	606.22s
B 无人机	4663.6002m	466.36s

9 问题三分析

问题三要求当 B 站点到圆心的距离在 $[0.5, 10]$ km 变化 (其他参数保持不变) 时, 指出问题一二中最优航迹的变化。通过第二问的几何证明, 本文依旧延续第二问的研究思路。

本文在 $[0.5, 10]$ km 范围内遍历取值, 拆分为多个第二问进行求解。如此, 对于需要补偿的航线长度计算只需沿用第二问的解法即可。

不同于问题二, 由于 B 站点到圆心的距离开始变化。出现了两个问题:

1. 待补偿的航线长度可能会出现小于一个半径 30m 整圆周长的情况, 需要给出新的航线长度补偿方案作为补充。

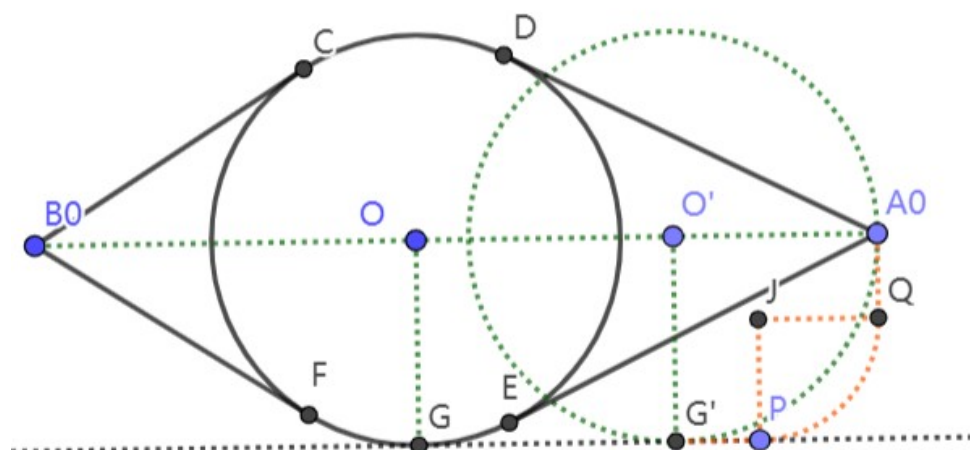


图 13: 航线补偿方案三

变化。同样地, 这种方案航线的增加也是连续的。经过计算, 这种方案下最大的航线补偿长度超过了半径为 30m 的整圆的周长。

注释

至此, 本文完全呈现了三种航线补偿方案。上述着重讨论了对 A 无人机航线的补偿方案。同理, 三种航线补偿方案也适用于 B 无人机。它们的结合可以实现任意航线补偿长度的精确补偿。

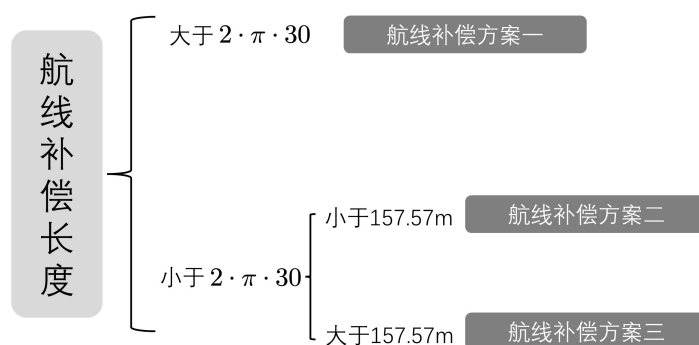


图 14: 三种航线补偿方案

10.1.3 补偿方案二、三的伪代码

如图 13,14, 如果想将轨迹 $\overrightarrow{A_0E} + \overrightarrow{EF}$ 切换成一条长度为 $|\overrightarrow{A_0E}| + |\overrightarrow{EF}| + \Delta S$ 的轨迹 (其中, $0 < \Delta S < 2\pi \cdot 30$), 可进行如下操作。

STEP1: 如图 13, 将圆心 O 向右平移 $|\overrightarrow{OO'}|$ 得到圆心 O' , 以 O' 为圆心, 500m 为半径作圆, 新轨迹为 $\overrightarrow{A_0E'} + \widehat{E'G_1} + \overrightarrow{G'_G}$, (其中, $|\overrightarrow{OO'}| + r \leq |\overrightarrow{OA_0}|$)

下面求解 $y_1 = |\overrightarrow{A_0E'}| + |\widehat{E'G'}| + |\overrightarrow{G'_G}|$ 及 $y_2 = |\overrightarrow{A_0E}| + |\overrightarrow{EG}|$ 的表达式。

$$\angle E'O'A_0 = \arccos \left(\frac{r}{|OA_0| - |\overrightarrow{OO'}|} \right) \quad (24)$$

$$\therefore y_1 = r \cdot \tan \angle E'O'A_0 + 2\pi r \cdot \frac{90^\circ - \angle E'O'A_0}{360^\circ} + |\overrightarrow{OO'}| \quad (25)$$

$$\angle EOA_0 = \arccos \left(\frac{r}{|OA_0|} \right) \quad (26)$$

$$\therefore y_2 = r \cdot \tan \angle EOA_0 + 2\pi r \cdot \frac{90^\circ - \angle EOA_0}{360^\circ} \quad (27)$$

$$\therefore \Delta S = y_1 - y_2 \text{ 可表示为 } f_{\Delta s} = g(|OO'|)$$

STEP2: 如果 $y_1 < \max(y_1 - y_2) < 2\pi \cdot 30$ 可进行如下操作。新轨迹为 $\overrightarrow{A_0Q} + \widehat{QP} + \overrightarrow{PG}$

下面推导 $y_3 = |\overrightarrow{A_0Q}| + \widehat{QP} + \overrightarrow{PG}$

设

$$|\overrightarrow{A_0Q}| = |\overrightarrow{PG'}| = \Delta r \quad (28)$$

显然,

$$|\widehat{QP}| = 2\pi (r - \Delta r) \cdot \frac{1}{4} \quad (29)$$

$$\left| \overrightarrow{G'G} \right| = |OO'| = \left| \overrightarrow{OA_0} \right| - r \quad (30)$$

$$\therefore y_3 = 2\Delta r + 2\pi(r - \Delta r) \cdot \frac{1}{4} + \left| \overrightarrow{OA_0} \right| - r \quad (31)$$

$$\therefore \Delta_{s2} = y_3 - y_2 \text{ 可表示为 } f_{\Delta s2} = g_4(\Delta r)$$

综上, 通过 ΔS 可以反推 $|OO'|$ 和 Δr , 即可确定新轨迹。

10.1.4 轨迹分配

轨迹的分配依赖于在左半平面碰面还是在右半平面碰面。如果在左半平面碰面, 简单来说就是 A 无人机跨越障碍圆这段距离飞行距离太短, 需要补偿一定的航线长度。此时, B 无人机的航线为极限轨迹。反之, B 无人机的航线需要补偿一定的航线长度, A 无人机的航线为极限轨迹。

首先, 假定两架无人机会在左半平面碰面, 如果假设成立的话, 以左半平面碰面为假设解算出来的 ΔS 应为正值, 如果解算出来的值为负值, 说明假设不成立。此时, 有两种情况, 两架无人机实际在右半平面碰面或者两架无人机均保持极限轨迹时不会碰面。

当以左半平面碰面为假设解算出的 ΔS 为负时, 就需要做出两架无人机在右半平面碰面的假设, 如果在这个假设中解的 ΔS 为正, 说明两无人机均保持极限轨迹是会在右半平面碰面。如果这里的 ΔS 其值也为负值, 说明两无人机均保持极限轨迹时不会发生碰面。

注释

问题三中涉及的“航线补偿方案”和“轨迹分配方案”对于后续四、五的求解也具有重要的意义。



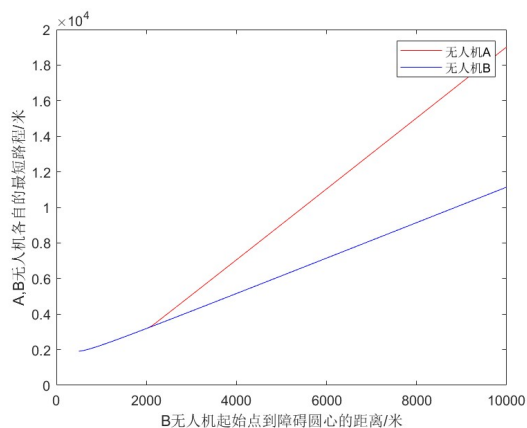
图 15: 轨迹分配方案

10.2 问题三的求解

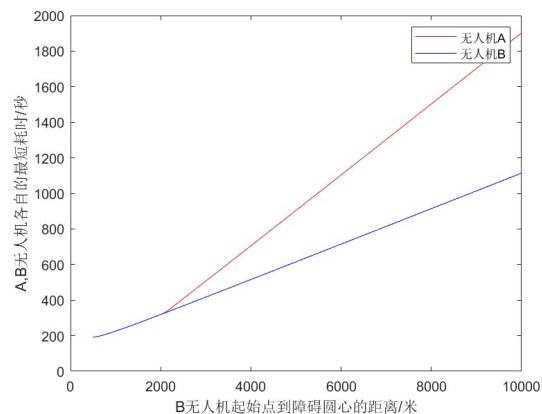
对距离在 $[0.5, 10]$ km 上进行遍历, 最终得到了 A、B 两架无人机最短路程图像, 用最短路程图像除以各自的速率得到最短耗时图像, 如图16。

分析最短路程图像, 在基站 B 与障碍圆圆心距离为 500m 到 2115m 时, 两无人机的最短路程相等。此种情况下, 两无人机均保持极限轨迹, 不碰面。

2115m 之后, A、B 无人机在左半平面碰面。此时, B 无人机保持极限轨迹航行, A 无人机航线需要补偿长度。如果补偿的长度大于一个半径 30m 的整圆周长, 则采用航线补偿方案一; 如果补偿的长度小于一个半径 30m 的整圆周长但是大于 157.57m, 则采用航线补偿方案三; 如果补偿的长度小于 157.57m, 则采用航线补偿方案二。



(a) A、B 两无人机最短路程与基站 B 距离圆心距离的关系



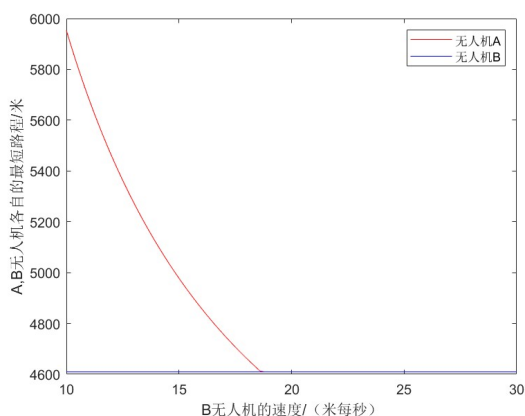
(b) A、B 两无人机最短耗时与基站 B 距离圆心距离的关系

图 16: A、B 两无人机从最短距离与基站 B 距离圆心距离的关系

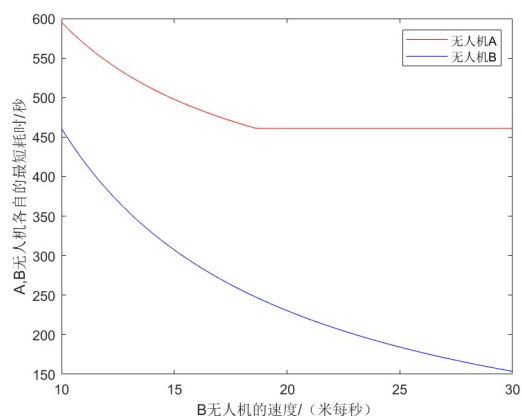
11 问题四分析

问题四要求 B 无人机的恒定速率在 $[10,30]$ m/s 内变化 (其他参数保持不变), 指出第一二问中最优航迹的变化。对 B 无人机速度在 $[10,30]$ m/s 上进行遍历, 首先得到 A、B 两架无人机的最短路程。之后, 用最短路程除以各自的恒定速率得到两架无人机的最短耗时。

12 问题四的求解



(a) A、B 两无人机最短路程与 B 无人机速度的关系



(b) A、B 两无人机最短耗时与 B 无人机速度的关系

图 17: A、B 两无人机参数随无人机 B 速度变化的函数图像

在本问中, B 无人机始终保持极限轨迹。观察最短路程图像, 当 B 无人机的速率大于

10m/s 小于 18.6m/s 时, 无人机 A 和 B 的最短路程不同, 两架无人机会在左边碰面。此时, 无人机 A 需要补偿航线长度, 如果需要补偿的航线长度大于一个半径 30m 的整圆周长, 采用航线补偿方案一; 如果需要补偿的航线长度小于一个半径 30m 的整圆周长但大于 157.57m, 采用航线补偿方案三; 如果需要补偿的航线长度小于 157.57m, 采用航线补偿方案二。

当无人机 B 的速率大于 18.6m/s 时, 此时两架无人机均保持极限轨迹, 不会碰面。

13 问题五分析

第五问综合第三问和第四问, 要求在 B 基站距离障碍圆圆心距离和无人机 B 速率两个参数均变化的情况下, 指出问题二最优航迹的变化。本部分使用两层循环, 分别在距障碍圆圆心距离和无人机 B 速率两个参量的取值范围上遍历取值, 先得到两架无人机最短路程, 与各自的恒定速率相除后得到最短耗时与 V 、 S_B 的图像。

14 问题五的求解

图 18 为假设 A、B 两无人机在右半平面碰面时, ΔS 关于 B 无人机基站距障碍圆圆心距离 S_B 和无人机 B 速率 V_B 的图像。其中, ΔS 取值为正的部分, 颜色为深蓝色。也即两无人机保持极限轨迹, 会在右半平面碰面的情形。

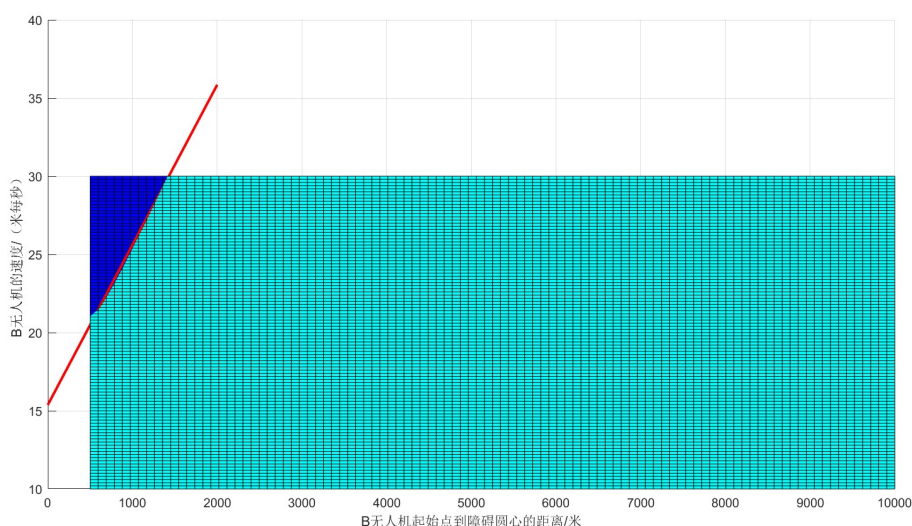


图 18: 右半面碰面分析图像

图 18 中红色直线方程解析式为:

$$V = 0.0102 \cdot S + 15.3738 \quad (32)$$

如果 B 基站距离障碍圆圆心的距离 S 与无人机 B 飞行速率 V 位于图 18 中红色直线上方, 则无人机 A 和无人机 B 在右半平面碰面。

这种情况下, A 无人机的航线为极限轨迹, B 无人机的航线需要增加补偿长度。如果补偿长度大于一个半径 30m 的整圆周长, 按照航线补偿方案一补偿; 如果补偿长度小于一个半径 30m 的整圆周长但大于 157.57m, 则按照航线补偿方案三补偿; 如果补偿长度小于 157.57m, 则按照航线补偿方案二补偿。

图 19 为假设 A、B 两无人机保持极限轨迹会在左半平面碰面时, ΔS 关于 B 无人机基站距障碍圆圆心距离 S_B 和无人机 B 速率 V_B 的图像。其中, ΔS 取值为正的部分, 颜色为深蓝色。也即两无人机保持极限轨迹, 会在左半平面碰面的情形。

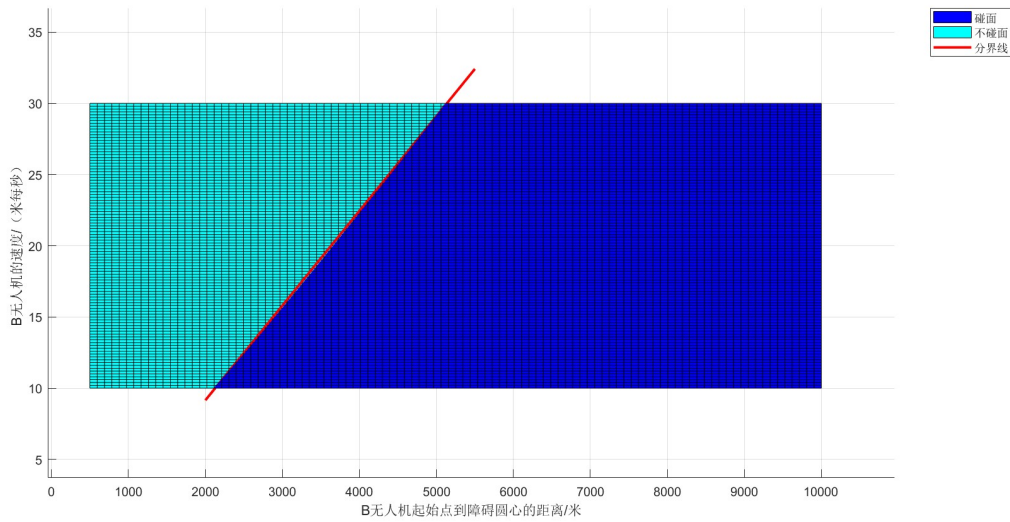


图 19: 左半面碰面分析图像

图 19 中红色直线方程解析式为:

$$V = 0.0066 \cdot S - 4.1417 \quad (33)$$

如果 B 基站距离障碍圆圆心的距离 S 与无人机 B 飞行速率 V 位于图 19 中红色直线下方, 则无人机 A 和无人机 B 在左半平面碰面。

这种情况下, B 无人机的航线为极限轨迹, A 无人机的航线需要增加补偿长度。如果补偿长度大于一个半径 30m 的整圆周长, 按照航线补偿方案一补偿; 如果补偿长度小于一个半径 30m 的整圆周长但大于 157.57m, 则按照航线补偿方案三补偿; 如果补偿长度小

于 157.57m, 则按照航线补偿方案二补偿。

最后, 如果 B 基站距离障碍圆圆心的距离 S 与无人机 B 飞行速率 V 位于图 18 和图 19 中两红色直线中间, 则无人机 A 和无人机 B 不会碰面, 两者航线均为极限轨迹。

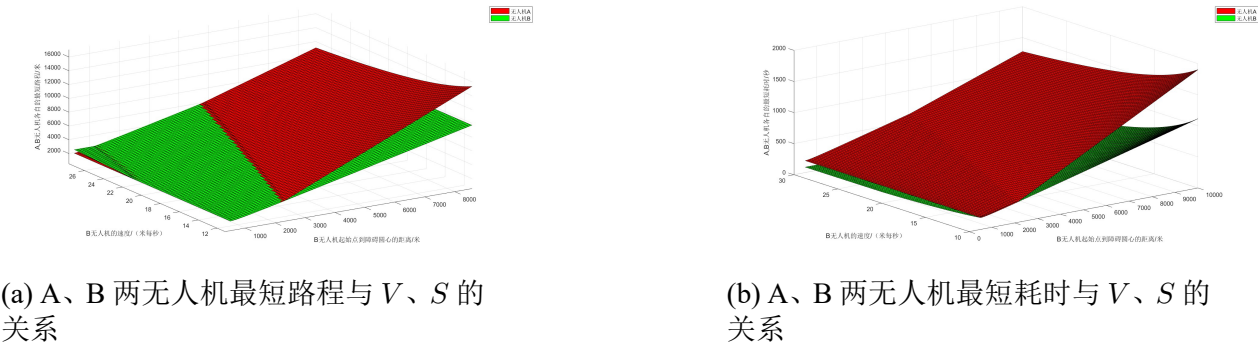


图 20: A、B 两无人机参数随 V 、 S 变化的函数图像

参考文献

- [1] 黄科薪, 黄金昌, 张燕芳等. 我国无人机发展现状及趋势展望 [J]. 厦门科技, 2023(01):59-62.
- [2] 丁肖倩. 多无人机协同避障规划系统研究 [D]. 哈尔滨工业大学, 2022. DOI:10.27061/d.cnki.ghgdu.2022.003369.

附录

A function_singleLoop.m

```

1      function [delta_SS,delta_S22]=function_singleLoop(Vbb,Sobb)
2
3      %      clear,clc;
4
5      %%%定义变量
6      t=sym('t');
7      Vb=sym('Vb');
8      Sob=sym('Sob');
9      delta_S=sym('delta_S');
10     delta_S2=sym('delta_S2');
11
12     %%%定义常量
13     r=500;
14     Va=10;
15     Soa=1000;
16     specific_Sob=Sobb;
17     specific_Vb=Vbb;
18
19     %%%计算中间变量
20     %左半平面
21     Sbe=sqrt(Sob.^2-r.^2);
22     Snc=Va*t;
23     ANGLEnoc=atan(Snc./r);
24     ANGLEboc=acos(r/Sob);
25     ANGLEmoe=ANGLEboc-ANGLEnoc;
26     ARCCd=2*pi*r*(pi-ANGLEboc-pi/3)/(2*pi);
27     Sad=sqrt(3)*r;
28     %右半平面
29     San=Va*t;
30     Son=sqrt(Soa^2+San^2-2*Soa*San*cos(pi/6));
31     ANGLEnoa=asin(San*sin(pi/6)/Son);
32     Sfm=r*tan(ANGLEnoa);
33     %最短路程极限
34     minLimit_distance=ARCCd+Sad+Sbe;

```

```

35
36
37
38    %%%最终表达式
39    %左半平面
40    Sbm=Sbe-r*tan(ANGLEmoe);
41    Ya=Sbm;
42    Yb=Vb*t+(Sad+ARCcd)/Va*Vb;
43    EndTime_left=Sbe/Va;
44    %右半平面
45    Ya2=Sfm;
46    Yb2=Vb*t-(Sbe+ARCcd);
47    EndTime_right=Sad/Va;
48
49
50    %%%代入数据
51    %左半平面
52    Ya=subs(Ya,[Vb,Sob],[specific_Vb,specific_Sob]);
53    Yb=subs(Yb,[Vb,Sob],[specific_Vb,specific_Sob]);
54    diff_Ya=diff(Ya);
55    EndTime_left=subs(EndTime_left,Sob,specific_Sob);
56    EndTime_left=double(EndTime_left);
57    %右半平面
58    diff_Ya2=diff(Ya2);
59    Yb2=subs(Yb2,[Vb,Sob],[specific_Vb,specific_Sob]);
60    EndTime_right=subs(EndTime_right,Sob,specific_Sob);
61    EndTime_right=double(EndTime_right);
62    %最短路程极限
63    minLimit_distance=double(subs(minLimit_distance,Sob,specific_Sob));
64
65
66
67    % %%%绘图
68    % figure;
69    % subplot(2, 1, 1);
70    % fplot(Ya,[0,EndTime_left],'r');
71    % hold on;
72    % fplot(Yb,[0,EndTime_left],'b');
73    % hold on;

```

```

74 % fplot(diff_Ya,[0,EndTime_left],'g');
75 % legend('无人机A的s-t关系','无人机B的s-t关系',
76
77 '无人机A的ds/dt-t关系');
78 % title('左半平面');
79 % subplot(2, 1, 2);
80 % fplot(Ya2,[0,EndTime_right],'r');
81 % hold on;
82 % fplot(Yb2,[0,EndTime_right],'b');
83 % hold on;
84 % fplot(diff_Ya2,[0,EndTime_right],'g');
85 % legend('无人机A的s-t关系','无人机B的s-t关系',
86
87 '无人机A的ds/dt-t关系');
88 % title('右半平面');
89
90
91 %%求数值解
92 %左半平面
93 eqns=[Ya-delta_S-Yb==0,diff_Ya-specific_Vb==0];
94 answer=solve(eqns,[t,delta_S]);
95 index = (real(double(answer.t)) >= 0) & (real(double(answer.t))
    <= EndTime_left & imag(double(answer.t))==0);
96 if (real(answer.t) >= EndTime_left) | (real(answer.t) <= 0)
97 answer_t=EndTime_left;
98 answer_deltaS=double(subs(Ya,t,answer_t)-subs(Yb,t,answer_t));
99 else
100 answer_t = double(answer.t(index));
101 answer_deltaS=double(answer.delta_S(index));
102 end
103 if EndTime_left==0
104 answer_deltaS=0;
105 end
106 %右半平面
107 eqns2=[Ya2-delta_S2-Yb2==0,diff_Ya2-specific_Vb==0];
108 answer2=solve(eqns2,[t,delta_S2]);
109 index2 = (real(answer2.t) >= 0) & (imag(answer2.t)==0);
110 if real(answer2.t) >= EndTime_right
111 answer_t2=EndTime_right;

```

```

112 answer_deltaS2=double(subs(Ya2,t,answer_t2)-subs(Yb2,t,answer_t2))
113 else
114 answer_t2 = double(answer2.t(index2));
115 answer_deltaS2=double(answer2.delta_S2(index2));
116 end
117
118 %%% 结论
119 if answer_deltaS>0
120 fprintf('B起点到障碍圆心距离为%.4f米\n',specific_Sob)
121 fprintf('B的速度为%.4f米\n',specific_Vb)
122 fprintf('为了避免在左半平面碰面\n')
123 fprintf('无人机B的最短路程为: %.4f米\n',minLimit_distance)
124 fprintf('无人机B的最短耗时为: %.4f秒\n',
125
126 minLimit_distance/specific_Vb)
127 fprintf('无人机A的最短路程为: %.4f米\n',minLimit_distance
128
129 +abs(answer_deltaS))
130 fprintf('无人机B的最短耗时为: %.4f秒\n',(minLimit_distance
131
132 +abs(answer_deltaS))/10)
133
134 elseif answer_deltaS2<0
135 fprintf('B起点到障碍圆心距离为%.4f米\n',specific_Sob)
136 fprintf('B的速度为%.4f米\n',specific_Vb)
137 fprintf('为了避免在右半平面碰面\n')
138 fprintf('无人机B的最短路程为: %.4f米\n',minLimit_distance
139
140 +abs(answer_deltaS2))
141 fprintf('无人机B的最短耗时为: %.4f秒\n',(minLimit_distance
142
143 +abs(answer_deltaS2))/specific_Vb)
144 fprintf('无人机A的最短路程为: %.4f米\n',minLimit_distance)
145 fprintf('无人机B的最短耗时为: %.4f秒\n',(minLimit_distance)/10)
146
147 else
148 fprintf('B起点到障碍圆心距离为%.4f米\n',specific_Sob)
149 fprintf('B的速度为%.4f米\n',specific_Vb)
150 fprintf('不会碰面\n')

```

```

151 fprintf('无人机B的最短路程为：%.4f米\n',minLimit_distance)
152 fprintf('无人机B的最短耗时为：%.4f秒\n',(minLimit_distance)
153
154 /specific_Vb)
155 fprintf('无人机A的最短路程为：%.4f米\n',minLimit_distance)
156 fprintf('无人机B的最短耗时为：%.4f秒\n',(minLimit_distance)/10)
157
158 end
159 delta_SS=answer_deltaS/Vbb*10;
160 delta_S22=answer_deltaS2;
161 end

```

B newCircumPath_v3.m

```

1 function
2     [final_So1o,final_deltaR,error]=newCircumPath_v3(Delta_S,
3
4     So1o_ceil)
5
6     So1o=[0];%一个决策变量并初始化为0
7     A=[];%
8     b=[];%线性不等式约束，默认A*x0<=b
9     Aeq=[];%
10    beq=[];%线性等式约束，默认Aeq*x0<=beq
11    lb=[0];%决策变量下限
12    ub=[];%决策变量上限
13    options = optimoptions('fmincon',...
14    'Algorithm','interior-point','ConstraintTolerance',1e-6 );
15    [answer_Solo,error]=fmincon(@object_fuc,So1o,A,b,Aeq,beq,lb,ub,
16
17    @condition_fuc,options)
18    function f=object_fuc(So1o)
19
20    r=500;
21    Soa=1000;

```

```

22
23     if So1o+r<=So1o_ceil
24
25         ANGLEao1e1=acos( (r)/(So1o_ceil-So1o) );
26         f3=r*tan(ANGLEao1e1)+2*pi*r*(pi/2-ANGLEao1e1)/(2*pi)+So1o;
27
28         ANGLEaoe=  acos( (r)/(So1o_ceil) );
29         f4=r*tan(ANGLEaoe)+2*pi*r*(pi/2-ANGLEaoe)/(2*pi);
30
31         f=(f3-f4-Delta_S)^2;
32
33     else
34
35         %             ANGLEao1e1=acos( (r)/(So1o_ceil-r) );
36         %
37         temp_f3=r*tan(ANGLEao1e1)+2*pi*r*(pi/2-ANGLEao1e1)/(2*pi)+So1o;
38         Delta_r=So1o-(So1o_ceil-r);
39         f3=2*Delta_r+2*pi*(r-Delta_r)/4+(So1o_ceil-r);
40
41         ANGLEaoe=  acos( (r)/(So1o_ceil) );
42         f4=r*tan(ANGLEaoe)+2*pi*r*(pi/2-ANGLEaoe)/(2*pi);
43
44         f=(f3-f4-Delta_S)^2;
45
46     end
47
48     function [c,ceq]=condition_fuc(x)
49     c=[];
50     %c为非线性不等式约束，默认c<=0
51     %若有多个约束条件，使用c=[c ...]添加约束
52     ceq=[];
53     %ceq为非线性等式约束，默认ceq=0
54     %若有多个约束条件，使用ceq=[ceq ...]添加约束
55     end
56
57     r=500;
58     if answer_Solo+r<=So1o_ceil
59         final_Solo=answer_Solo;
60         final_deltaR=0;

```

```
60     else
61         final_Solo=Solo_ceil-r;
62         final_deltaR=answer_Solo-final_Solo;
63     end
64
65
66     end
```