

1. 实验名称及目的

1.1. 实验名称

避障控制器设计实验 (设计实验)

1.2. 实验目的

在第 6 章 6.2 节设计的跟踪控制器以及 8.2 节设计的避障算法基础上, 设计多机合作避障算法, 完成两个递进的实验。

(1) 实验一: 给定两架多旋翼, 设计控制器引导它们避免碰撞, 并飞向对方初始位置, 同时记录多旋翼避障轨迹。

(2) 实验二: 给定两架多旋翼和一个障碍物, 设计控制器引导多旋翼避开障碍物飞向对方初始位置, 并记录多旋翼避障轨迹。

图 8.24 为两个实验的示意图, 给定两架多旋翼初始位置为(0,0)和(30,0), 障碍物位置随机摆放, 此处可定为(18,0), 障碍物半径定为 2m, 安全半径定为 3m。此外, 偏航角需要始终保持为 0 使机头保持一个方向, 高度始终保持在 1m。本实验具体目标包括以下几点。

- (1) 理解与熟悉人工势场法的理论与推导过程;
- (2) 实现多旋翼的多机避障控制实验;
- (3) 使用相同的控制器进行仿真 2.0 实验, 对比两种模型的仿真结果, 并进行硬件在环仿真。

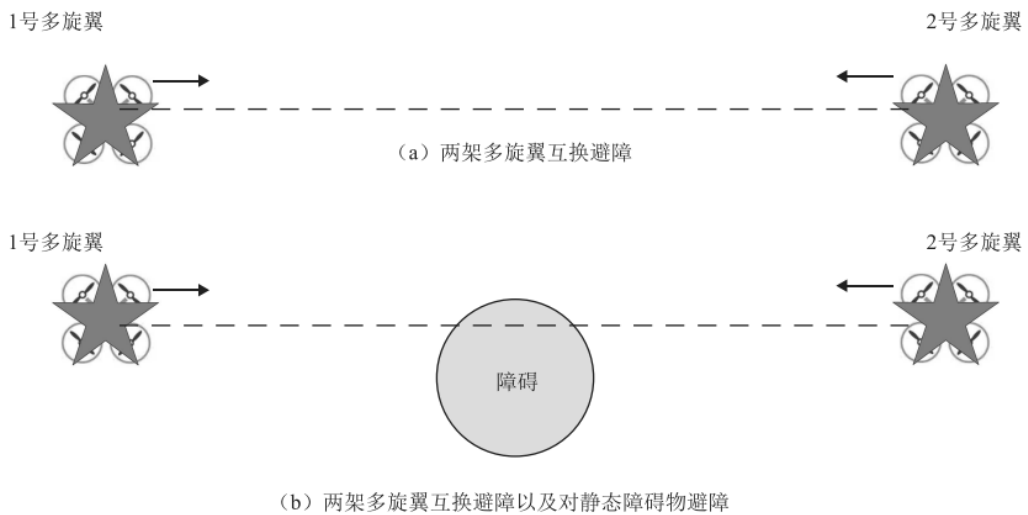


图 8.24 多机合作避障设计实验示意图

1.3. 关键知识点

在给农作物喷洒农药时, 使用多架多旋翼可以增加效率。这不仅要求多旋翼能够避开静态障碍物, 还要求多机之间合作避障。这个问题可以描述为: 基础实验中的静态障碍物具备速度, 变成移动的障碍物。

令 $\mathbf{p}_i \in \mathbb{R}^2$ 和 $\mathbf{v}_i \in \mathbb{R}^2$ 表示第 i 架多旋翼的当前位置和当前速度, $\mathbf{p}_{\text{wp},i} \in \mathbb{R}^2$ 表示第 i 架多旋翼的目标航路点, 根据质点模型

$$\begin{aligned}\dot{\mathbf{p}}_i &= \mathbf{v}_i \\ \dot{\mathbf{v}}_i &= \mathbf{u}_i\end{aligned}\quad (8.13)$$

设计虚拟控制输入 \mathbf{u}_i , 引导多旋翼从各自起飞点抵达目标位置 $\mathbf{p}_{\text{wp},i}$, 同时避开环境中位于 $\mathbf{p}_{o,j} \in \mathbb{R}^2$, $j = 1, \dots, m_o$, 速度为 $\mathbf{v}_{o,j} \in \mathbb{R}^2$ 且影响半径为 $r_{o,j} > 0$ 的 m_o 个障碍物。

全局势场可通过叠加目标航路点和所有障碍物的势场获得, 除自身外其他多旋翼视为移动的障碍物。根据势场的叠加结果, 可将局部期望位置 $\mathbf{p}_{d,i} \in \mathbb{R}^2$ 表示为

$$\mathbf{p}_{d,i} = \mathbf{p}_i + \mathbf{p}_{d,o}$$

这里

$$\mathbf{p}_{d,o} = \text{sat}_{\text{gd}} \left(k_1 (\mathbf{p}_{\text{wp},i} + k_0 \mathbf{v}_{\text{wp},i} - \mathbf{p}_i - k_0 \mathbf{v}_i) + \sum_{j=1}^{m_o} b_j (\mathbf{p}_i + k_0 \mathbf{v}_i - \mathbf{p}_{o,j} - k_0 \mathbf{v}_{o,j}), a_0 \right)$$

$$b_j = k_2 \frac{1}{(\|\mathbf{p}_i + k_0 \mathbf{v}_i - \mathbf{p}_{o,j} - k_0 \mathbf{v}_{o,j}\| - r_{o,j} - k_0 a_0)^2} \frac{1}{\|\mathbf{p}_i + k_0 \mathbf{v}_i - \mathbf{p}_{o,j} - k_0 \mathbf{v}_{o,j}\|}, j = 1, \dots, m_o$$

其中, $k_0, k_1, k_2 > 0$, 并且假设目标航路点是静态的, 即 $\mathbf{v}_{\text{wp},i} = \mathbf{0}_{2 \times 1}$ 。

虽然本章提出的避障算法简单实用, 且能有效地生成平滑的可飞路径, 但是在避障方面还存在一些问题。当目标航路点离障碍物较近的时候, 多旋翼可能无法抵达目标航路点。当环境中存在多个相近的障碍物, 并且这些障碍物的排斥力合力与目标航路点的吸引力大小相等而方向相反时, 多旋翼可能会陷入局部稳定点而无法逃离。

1) 整体模型

如图 8.25 所示, 整体模型分成两部分, 对应 1 号多旋翼和 2 号多旋翼。每个部分都包括控制输入模块、控制器模块和多旋翼模块。

2) 控制输入模块

因为两架多旋翼的算法设计几乎一致, 因此只考虑其中的 1 号多旋翼。打开图 8.25 中的“Controller Input1”模块, 该控制输入模块内部如图 8.26 所示。根据 8.4.2.1 节的原理, 当两架多旋翼对飞时, 全局势场可通过叠加目标位置和障碍物势场获得。根据势场的叠加结果, 局部期望位置不仅与自身的速度、位置反馈有关, 还与移动障碍物(其他的多旋翼)的速度、位置反馈有关。因此, 与基础实验不同, 这里引入了 2 号多旋翼的位置和速度反馈。进一步打开该控制输入模块, 可以看到如图 8.27 所示的内部结构。本次实验只考虑二维水平的避障问题, 并且让机头朝一个方向不变, 因此, 将高度设置为 1m, 偏航设置为 0, 只对水平通道进行避障控制器设计。

打开图 8.27 中的“X1”函数模块, 关键代码如表 8.5 所示。该代码给定了目标位置、障碍物位置和半径, 并对期望轨迹方程相关的参数进行了赋值。此外, 为了保证多旋翼到达

目标位置后不会因为斥力场的存在出现较大偏差，这里限制了斥力场的作用范围。当多旋翼与障碍物中心的距离超过一定值时，斥力场作用失效。在失效前到完全失效这一中间过程应当尽量平滑，因此也设计了一个三阶函数用于过渡。在飞行过程中，局部期望位置不断进行迭代更新，作为图 8.25 中“Controller1”模块的输入。图 8.27 的“Cache”变量用来打包需要的数据并发送到工作区，用于作图显示。

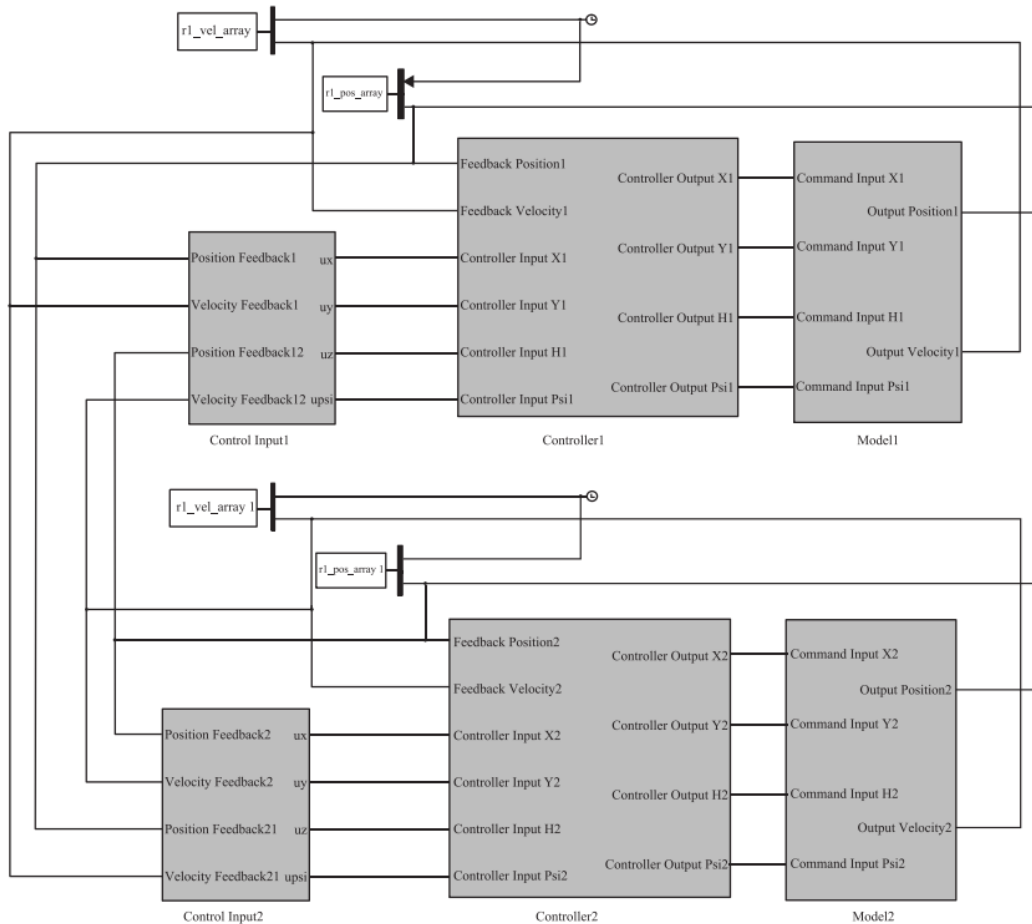


图 8.25 整体模型，Simulink 模型详见“e5\5.3\sim1.0\ e5_3_Avoidance_Segment.slx”

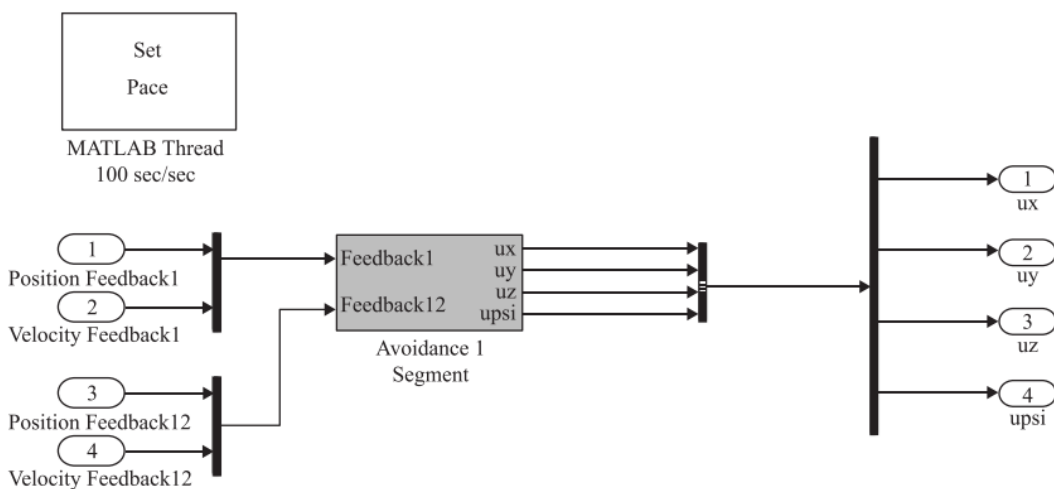


图 8.26 控制输入模块

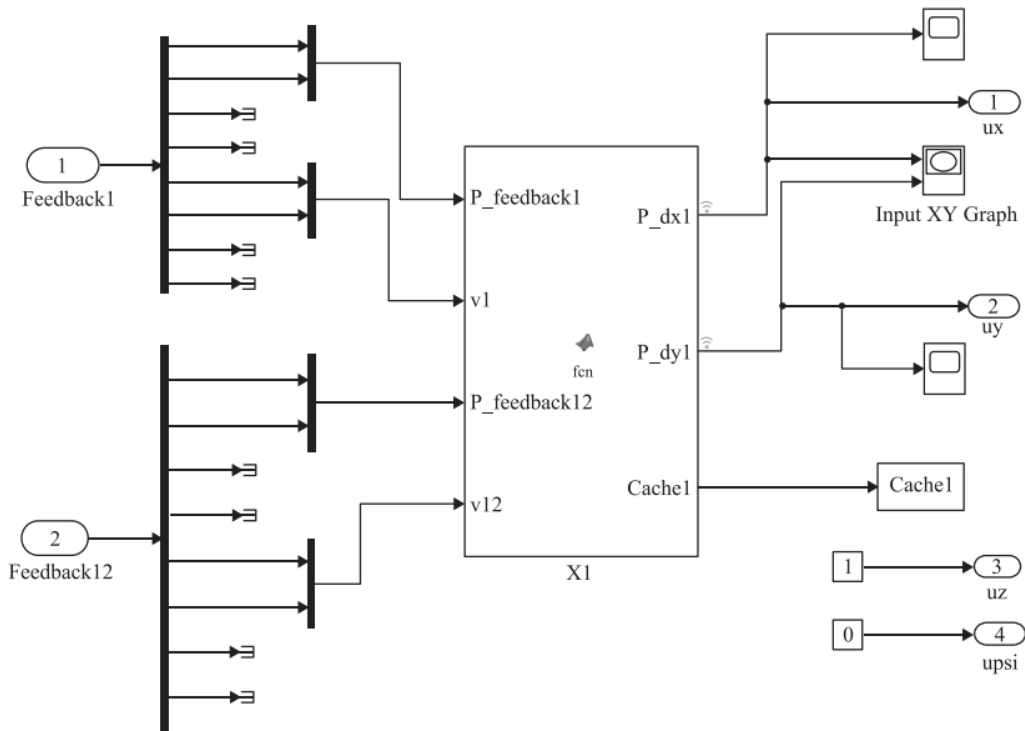


图 8.27 避障控制器模块

表 8.5 避障算法关键代码

```

1 temp1 = satgd(-a*ksi_wp+b*ksi_0,a0);
2 P_d1 = P_feedback1+ temp1; %局部期望位置
3 P_dx1=P_d1(1);
4 P_dy1=P_d1(2);

```

3) 控制器模块

打开图 8.25 中的“Controller1”模块，其内部结构如图 8.28 所示。这部分与基础实验一样，不再赘述，读者可根据需要，适当修改参数进行优化。

4) 多旋翼模块

仿真 1.0 和仿真 2.0 的多旋翼分别为各通道的传递函数模型以及非线性模型，这部分与基础实验一样。

2. 实验效果

通过示波器显示出仿真模型期望位置轨迹和实际位置轨迹对比。

3. 文件目录

例程目录：[\[安装目录\]\RflySimAPIs\6.RflySimExtCtrl\1.BasicExps\e7_MutUAVRemoteCtrl\5.Avoidance_Segment\e5.3\](#)

文件夹/文件名称		说明
Sim1.0	e5_3_Avoidance_code.m	控制器参数文件
	e5_plot.m	绘图文件

	e5_3_Avoidance_Segment_2017b.slx	多旋翼避障控制器 simulink 仿真 1.0 文件
	startSimulation.m	初始化参数文件
	SaveData.m	保存文件
sim2.0	e5_3_Avoidance_code.m	控制器参数文件
	e5_plot.m	绘图文件
	e5_3_Avoidance_Segment_2017b.slx	多旋翼避障控制器 simulink 仿真 2.0 文件
	startSimulation.m	初始化参数文件
	SaveData.m	保存文件
sim1.1	e5_3_Avoidance_code.m	控制器参数文件
	e5_3_Avoidance_Segment_2017b.slx	多旋翼避障控制器 simulink 仿真 1.1 文件
	e5_plot.m	绘图文件
	SaveData.m	保存文件
	startSimulation.m	初始化参数文件
sim2.1	e5_3_Avoidance_code.m	控制器参数文件
	e5_3_Avoidance_Segment_2017b.slx	多旋翼避障控制器 simulink 仿真 2.1 文件
	e5_plot.m	绘图文件
	SaveData.m	保存文件
	startSimulation.m	初始化参数文件
sim1.0_vs_sim2.0	e5_compare_plot.m	比较结果文件
sim1.1_vs_sim2.1	e5_compare_plot.m	比较结果文件
HIL	e5_3_Avoidance_code.m	控制器参数文件
	e5_3_Avoidance_Segment_HITL_R2017b.slx	多旋翼避障控制器硬件在环文件
	e5_plot.m	绘图文件
	startSimulation.m	初始化参数文件

4. 运行环境

序号	软件要求	硬件要求	
		名称	数量
1	Windows 10 及以上版本	笔记本/台式电脑 ^①	1
2	RflySim 工具链	Pixhawk 6C ^②	1
3	MATLAB 2022b 及以上	遥控器 ^③	1
		遥控器接收器	1
		数据线、杜邦线等	若干

- ①：推荐配置请见：<https://doc.rflysim.com>
- ②：须保证平台安装时的编译命令为：`px4_fmuv6c_default`，固件版本为：1.13.3。其他配套飞控请见：<http://doc.rflysim.com>
- ③：本实验演示所使用的遥控器为：天地飞 ET10、配套接收器为：WFLY RF209S。遥控器相关配置见：..\e11_RC-Config\Readme.pdf

5. 实验步骤

5.1. 仿真 1.0(sim1.0)实验步骤

Step 1: 初始化参数

在完成参数设计之后，可以开始仿真。打开文件夹“e5\e5.3\sim1.0”，运行“startSimulation.m”文件初始化参数。同时，“c5_3_Avoidance_code.m”文件也会自动运行，其中含有实验模型所需要的必要参数信息以及多旋翼起始位置信息，具体内容如表 8.6 所示。之后只需要打开模型文件“e5_3_Avoidance_Segment_2017b.slx”，按照实验目标设定避障控制器中多旋翼目标位置、障碍物位置及半径。最后，运行该模型文件，获得相应的结果。



表 8.6 实验一仿真 1.0 部分参数

```

1 e5_3_Avoidance_Segment
2 P0_x1=0; P0_y1=-0.1; % 1号多旋翼初始位置
3 P0_x2=30; P0_y2=0.1; % 2号多旋翼初始位置
4 k1_p_x=1/0.8; k1_d_x=1/1; % 控制器参数
5 k2_p_x=1/0.8; k2_d_x=1/1;
6 k1_p_y=1/0.8; k1_d_y=1/1;
7 k2_p_y=1/0.8; k2_d_y=1/1;
8 k1_p_z=1; k1_d_z=10;
9 k2_p_z=1; k2_d_z=10;
10 k1_p_yaw=1.6; k1_d_yaw=10;
11 k2_p_yaw=1.6; k2_d_yaw=10;

```

注意，在多旋翼相遇之前，对于每架多旋翼，都与目标位置以及另一架多旋翼处于同一条直线上，理论上存在“死锁”问题，也就是多旋翼稳定到速度为零的状态，但未能到达目标点。但实际上，根据 8.1 节的原理分析可知，多旋翼不会严格在同一条直线上飞行，尤其是在三维测度上，“死锁”问题不可能存在或者说概率为 0。本实验考虑的是二维测度仿真，针对“死锁”情况，将起点位置设置为(0,-0.1)和 (30,0.1)，目标位置设置为(30,-0.1)和(0,0.1)。这样两架多旋翼在两条十分接近的平行线上进行对飞避障。

Step 2: 查看结果

在调试后，选取参数 $a_1 = 1$ 、 $k_1 = 0.2$ 、 $k_2 = 0.11$ 进行实验，仿真时长为 50s。仿真结束

后，通过运行“e5_plot.m”文件绘制两架多旋翼的轨迹，结果如图 8.29 所示。该图以水平前向通道位置值为横坐标，以水平侧向通道位置值为纵坐标构成二维平面图形。图中包含起点位置 1、起点位置 2、两架多旋翼的期望位置轨迹与实际位置轨迹。从图中可以看出，从初始位置开始，多旋翼的期望位置轨迹与实际位置轨迹十分吻合，并且成功地进行了相互避障到达目标位置，可以认为该避障算法是可行的。

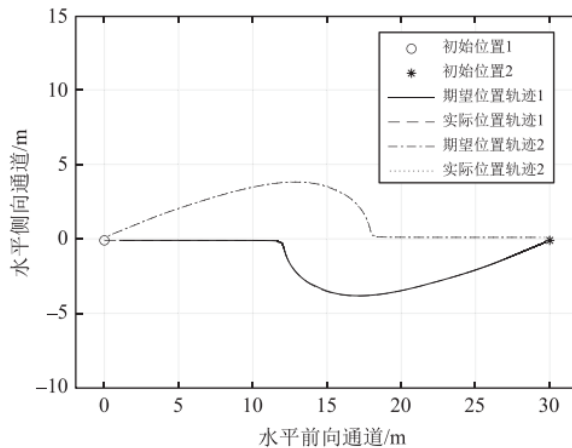


图 8.29 实验一仿真 1.0 期望位置轨迹和实际位置轨迹对比

Step 3: 保存结果

Simulink 仿真完成后，可运行文件“SaveData.m”将仿真 1.0 的实验数据保存为“.mat”格式文件，用于和仿真 2.0 数据对比。

5.2. 仿真 2.0(sim2.0)实验步骤

接下来，我们将进行仿真 1.0 和仿真 2.0 的对比。对于在传递函数模型中已经完成的实验过程，在这里需要进一步在非线形模型上进行验证。

打开文件夹“e5\5.3\sim2.0”，其中的文件与“e5\5.3 \sim1.0”基本相同，区别在于这里替换多旋翼模型为非线形模型。注意，与仿真 1.0 不同，这里需要在初始文件“startSimulation.m”中修改两架多旋翼的初始位置。相关代码见表 8.8。采用与仿真 1.0 完全相同的步骤进行实验，最后运行文件“e5_plot.m”获得如图 8.31 所示结果。

表 8.8 实验一仿真 2.0 两架多旋翼初始位置

```

1 %多旋翼初始状态
2 ModelInit.PosE1=[0,-0.1,0]; %1号多旋翼初始位置（单位：m）
3 ModelInit.PosE2=[30,0.1,0]; %2号多旋翼初始位置（单位：m）

```

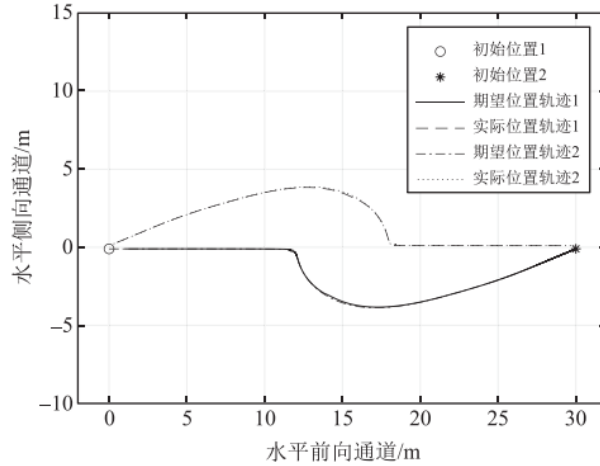


图 8.31 实验一仿真 2.0 期望位置轨迹和实际位置轨迹对比

若要进行对比，运行文件“e5\5.3\ sim1.0_vs_sim2.0\5_compare_plot.m”文件即可，对比效果如图 8.32 所示。从图中可以看出，这四条轨迹对应重合在一起，意味着采用系统辨识获得的模型进行实验时，所获得的避障控制器可以很好地模拟真实模型的避障情况，并且可以达到较好的效果，因此在接下来的实验中可以利用传递函数模型进行实验。此外，仿真 2.0 也可以在 FlightGear 中查看仿真飞行效果。

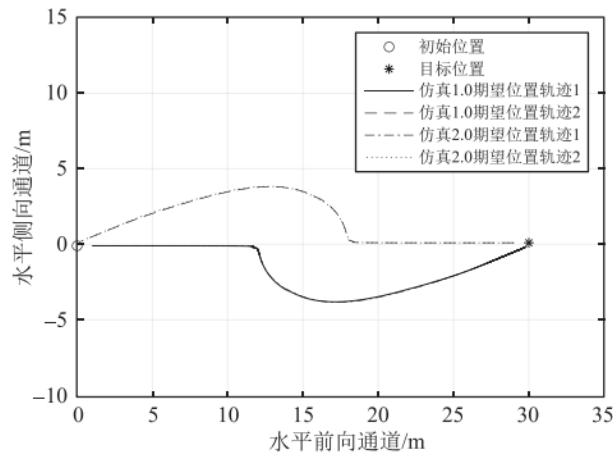


图 8.32 实验一仿真 2.0 不同模型期望位置轨迹对比

5.3. 仿真 1.1(sim1.1)实验步骤

Step 1: 初始化参数

在完成了参数设计之后，可以开始仿真。打开文件夹“e5\5.3\sim1.1”，运行“startSimulation.m”文件初始化参数，“e5_3_Avoidance_code.m”文件也会自动运行，具体内容如表 8.7 所示。之后打开模型文件“e5_3_Avoidance_Segment_2017b.slx”，按照实验目标设定避障控制器中多旋翼目标位置、障碍物位置及半径。最后，运行该模型文件，获得相应的结果。同样地，针对“死锁”情况，将起点位置设置为(0,0.1)和(30,-0.1)，目标位置设置为(30,0)和(0,-0.1)。

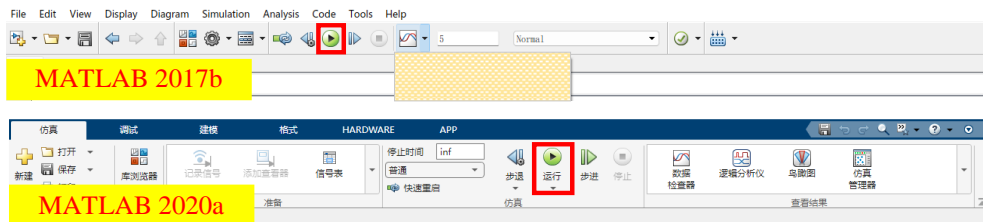


表 8.7 实验二仿真 1.0 控制器参数

```

1 e5_3_Avoidance_Segment
2 P0_x1=0; P0_y1=-0.1;           % 1号多旋翼初始位置
3 P0_x2=30; P0_y2=0.1;          % 2号多旋翼初始位置
4 k1_p_x=1/0.8; k1_d_x=1/0.25; % 控制器参数
5 k2_p_x=1/0.8; k2_d_x=1/0.25;
6 k1_p_y=1/0.8; k1_d_y=1/0.25;
7 k2_p_y=1/0.8; k2_d_y=1/0.25;
8 k1_p_z=1; k1_d_z=10;
9 k2_p_z=1; k2_d_z=10;
10 k1_p_yaw=1.6; k1_d_yaw=10;
11 k2_p_yaw=1.6; k2_d_yaw=10;

```

Step 2: 查看结果

在调试后，选取参数 $a_1=0.5$ 、 $k_1=0.1$ 、 $k_2=10$ 进行实验，仿真时长为 70s。仿真结束后，通过运行“e5_plot.m”文件绘制两架多旋翼的轨迹，结果如图 8.30 所示。从图中可以看出，从初始位置开始，多旋翼的期望位置轨迹与实际位置轨迹十分吻合，并且成功地进行了相互避障，绕开静态障碍物到达目标位置，可以认为该避障算法是可行的。

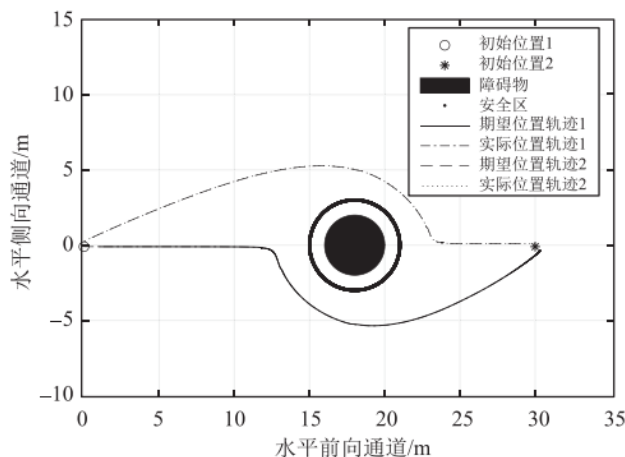


图 8.30 实验二仿真 1.0 期望位置轨迹和实际位置对比

Step 3: 保存结果

Simulink 仿真完成后，可运行文件“SaveData.m”将仿真 1.0 的实验数据保存为“.mat”格式文件，用于和仿真 2.0 数据对比。

5.4. 仿真 2.1(sim2.1)实验步骤

打开文件夹“e5\5.3\sim2.1”，其中的文件与上面的“e5\5.3\sim1.1”基本相同，区别在

于这里的多旋翼模型为非线性模型。两架多旋翼的初始位置同样需要在初始文件“startSimulation.m”中进行修改。相关代码见表 8.9。采用与仿真 1.0 完全相同的实验步骤，运行文件“e5_plot.m”获得结果如图 8.33 所示。

表 8.9 实验二仿真 2.0 初始位置设置

```
%多旋翼初始状态
ModelInit.PosE1=[0,-0.1,0]; %1号多旋翼初始位置（单位：m）
ModelInit.PosE2=[30,0.1,0]; %2号多旋翼初始位置（单位：m）
```

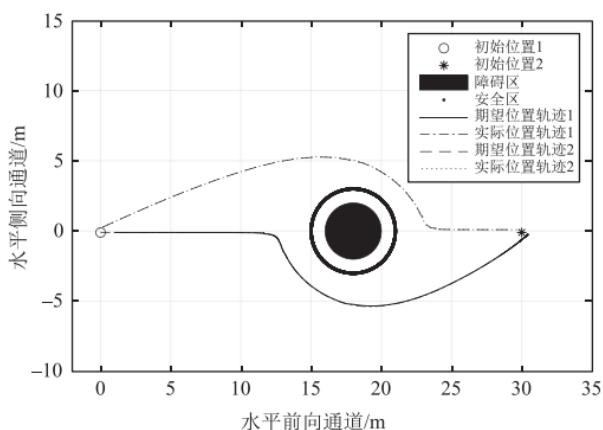


图 8.33 实验二仿真 2.0 期望位置轨迹和实际位置轨迹对比

从图中可以看出，实际位置轨迹跟踪期望位置轨迹，不仅避开了障碍物和另一架多旋翼，而且完美地到达了目标位置。若要进行对比，保存数据后运行文件“c5\5.3\sim1.1_vs_sim2.1\5_compare_plot.m”文件即可，对比效果如图 8.34 所示。从图中可以看出，两个实验的轨迹对应地重合在一起。这意味着采用系统辨识获得的模型进行实验时，所设计的避障算法可以很好地模拟真实模型的避障情况，并且达到期望的效果。

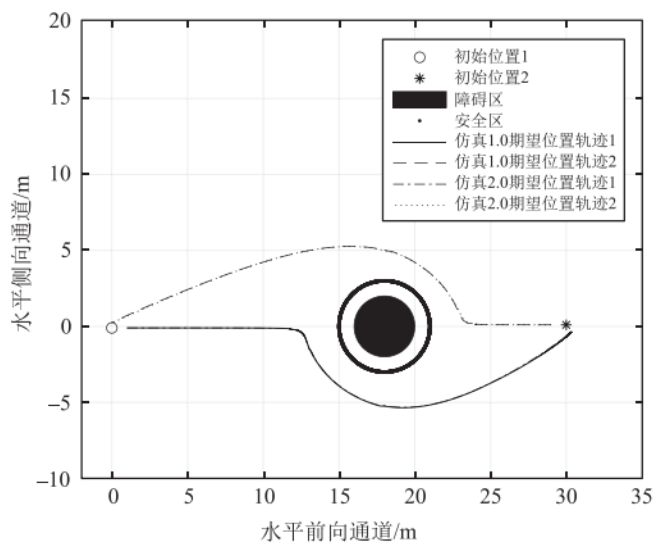


图 8.34 实验二仿真 2.0 不同模型避障轨迹对比

5.5. 硬件在环仿真

准备好两个自驾仪并打开 RflySim 平台。注意，此处是多机仿真，因此要打开名称为“HTLRunUdpFull.bat”的文件，在双击打开之前需要修改多旋翼的位置。右击该文件选择编辑，此时该文件将由记事本打开。找到设置多旋翼间距的地方，如表 8.10 所示，将多旋翼间距从 2m 修改为 30m。

表 8.10 实验二仿真 2.0 控制器参数

```
1 % 设置多旋翼间距
2 REM Set the interval between two vehicle , unit:m
3 SET /a VEHICLE_INTERVAL=2
```

修改完成之后双击打开该文件。运行“startSimulation.m”初始化参数并运行 Simulink 模型文件“e5_3_Avoidance_Segment_HUIL_2017b.slx”开始仿真。此时，可以在 Rfly-Sim3D 软件中看到硬件在环仿真的效果，如图 8.35 所示。实验数据保存至工作区中，单击运行“e5_plot.m”文件可以得到如图 8.36 所示的结果。从图中可以看出，两架多旋翼在互相躲避的同时完成了对静态障碍物的避障，实现了预期目标，验证了人工势场法的正确性和可行性。



图 8.35 高度通道硬件在环仿真的效果图

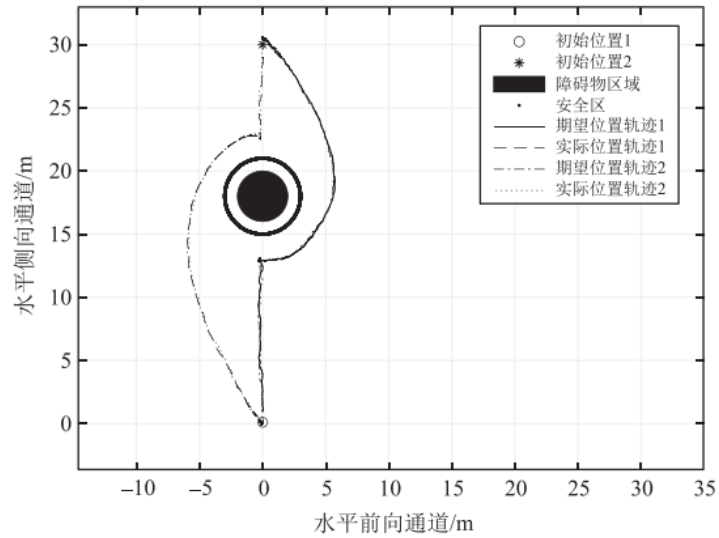


图 8.36 硬件在环仿真的避障结果

6. 参考资料

- [1]. Quan Quan. Introduction to Multicopter Design and Control. Springer, Singapore, 2017
- [2]. 全权 杜光勋 赵峙尧 戴训华 任锦瑞 邓恒译 多旋翼飞行器设计与控制 [M] 电子工业出版社 2018.
- [3]. 全权 戴训华 王帅 多旋翼飞行器设计与控制 实践 [M] 电子工业出版社 2020.
- [4]. 全权 等.多旋翼无人机远程控制实践[M].电子工业出版社,2022.

7. 常见问题

Q1: ****

A1: ****