

1. 实验名称及目的

1.1. 实验名称

路径跟随控制基础实验

1.2. 实验目的

给定多旋翼传递函数模型，该模型已经包含之前设计的跟踪控制器。设计路径跟随控制器让传递函数模型输出进行直线路径跟随。直线初始点为(5,-3)，终点为(5,10),多旋翼初始位置随机，可以定为(0,0)。此外，偏航角通道需要始终保持为 0，高度始终保持在 2m。

- (1) 学习并掌握路径跟随的理论基础及原理；
- (2) 实现多旋翼直线路径跟随控制。

1.3. 关键知识点

在多旋翼执行喷洒药物或者电线巡线任务时，会要求多旋翼能够严格压着航线进行飞行，如图 7.1 所示。该问题可以描述为：当前多旋翼水平位置为 $\mathbf{p} \in \mathbb{R}^2$ ，当前航路点为 $\mathbf{p}_{wp} \in \mathbb{R}^2$ ，上一个航路点为 $\mathbf{p}_{wp,last} \in \mathbb{R}^2$ ，点 $\mathbf{p}_{wp,last}$ 和点 \mathbf{p}_{wp} 可以构成一条直线；设计出实时航路点 $\mathbf{p}_d \in \mathbb{R}^2$ 引导多旋翼到达该直线并沿直线飞行，最终到达目标航路点 \mathbf{p}_{wp} 。

可以将多旋翼视为质点，其满足牛顿第二定律：

$$\begin{aligned} \dot{\mathbf{p}} &= \mathbf{v} \\ \dot{\mathbf{v}} &= \mathbf{u} \end{aligned} \tag{7.1}$$

其中， $\mathbf{u} \in \mathbb{R}^2$ 表示虚拟控制量。如图 7.1 所示，多旋翼位置 \mathbf{p} 到这条航线的距离为 $\|\mathbf{p} - \mathbf{p}_{wp,perp}\|$ ，其中 $\mathbf{p}_{wp,perp}$ 表示垂足，为

$$\mathbf{p}_{wp,perp} = \mathbf{p}_{wp} + (\mathbf{p}_{wp,last} - \mathbf{p}_{wp}) \frac{(\mathbf{p} - \mathbf{p}_{wp})^T (\mathbf{p}_{wp,last} - \mathbf{p}_{wp})}{\|\mathbf{p}_{wp} - \mathbf{p}_{wp,last}\|^2} \tag{7.2}$$

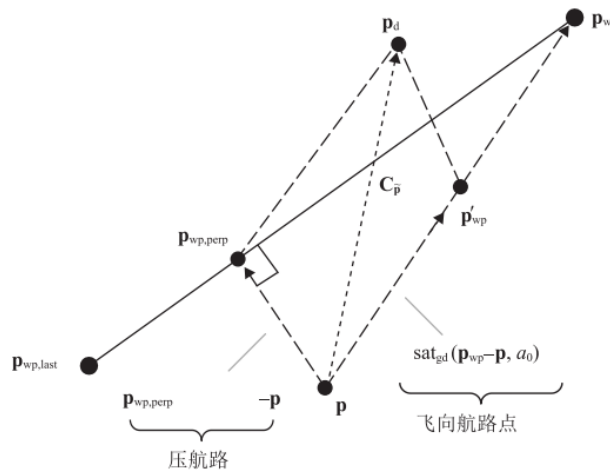


图 7.1 基于人工势场压航线的路径规划

关键知识点 1：实验整体流程

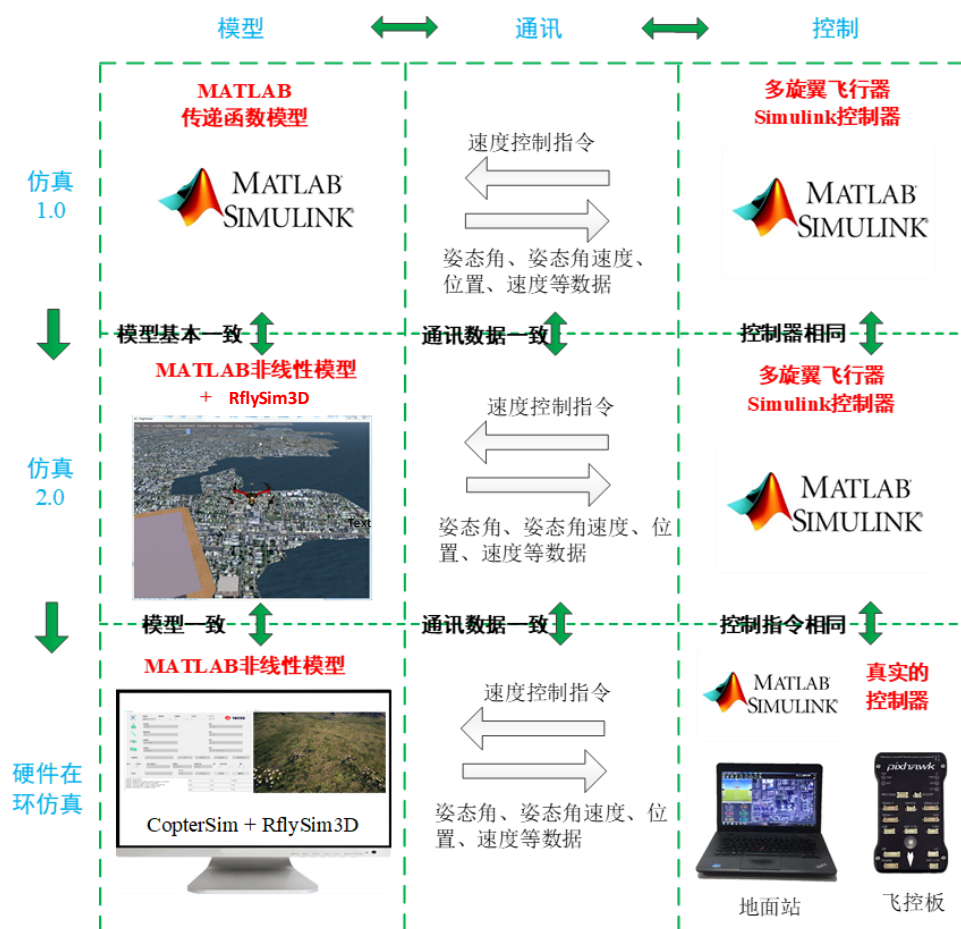
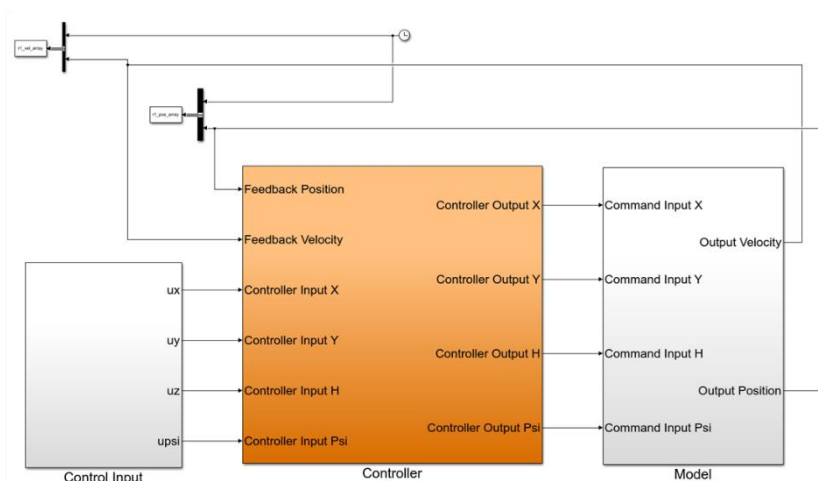


图 仿真阶段流程

针对设计所需要的仿真实验平台，如下图所示。仿真 1.0、仿真 2.0 和硬件在环仿真三者的区别在于多旋翼模型模块不一样：仿真 1.0 的多旋翼模型模块内部主要包含通过“系统辨识”实验得到的传递函数模型，即设计模型；仿真 2.0 的多旋翼模型模块内部包含一个给定的非线性模型和 RflySim3D 显示模块；硬件在环仿真的多旋翼模型模块内部包含了与 CopterSim 联合仿真的通信接口。

关键知识点 2: Simulink 模型整体框架



期望输入模块—Control Input: 用于输入期望的飞行器位置信息;

控制器模块—Controller: 用于设计控制器, 实现飞行器的有效位置控制;

多旋翼模型模块—Model: 飞行器仿真模型模块, 用于接收控制器输出的控制量, 并输出飞行器的运动状态 (这里是位置信息和速度信息;)

飞行器状态数据收集: 这里拥有两个变量——“r1_pos_array”与“r1_vel_array”, 分别用于收集单次仿真过程中飞行器的位置信息与速度反馈, 并以矩阵的形式保存在 MATLAB 工作空间。矩阵的每一行信息均是以“时间, 水平前向通道位置/速度, 水平侧向通道位置/速度, 高度通道位置/速度, 偏航通道角度/偏航角速度”方式自左而右排列的一组值, 矩阵的列按时间顺序自上而下排列。

关键知识点 3: 开环线性分析和校正

关键知识点 4: 模型模块详解

“Control Input”期望输入模块

整体结构

该模块包含了三种不同的期望输入, 分别是阶跃信号、圆以及“8”字, 圆以及“8”字信号有三种不同的周期。

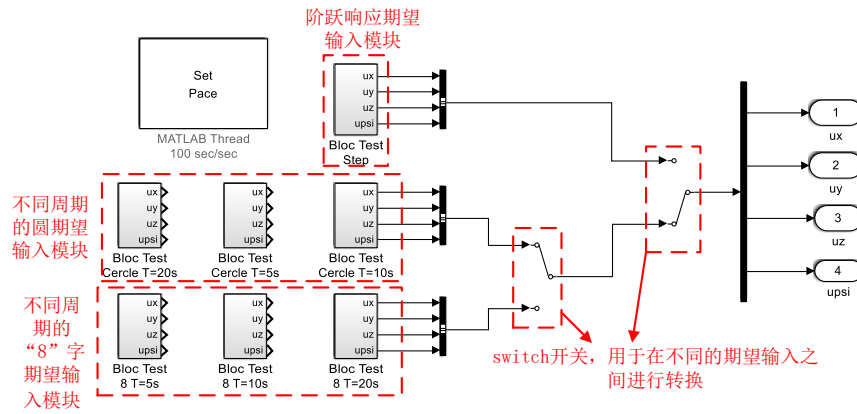


图 1 期望输入模块

期望输入的具体结构

以周期为 10s 的“8”字轨迹期望输入为例

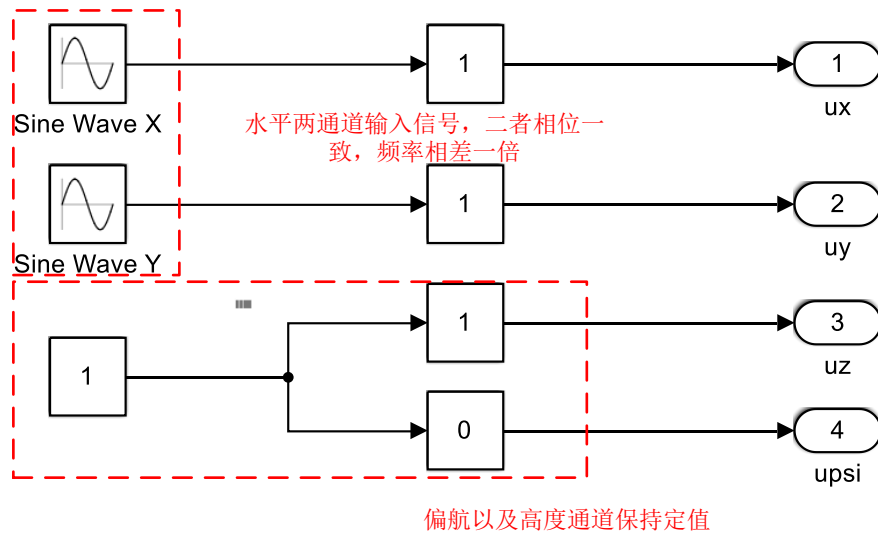


图 2 “8”字轨迹期望输入

“Controller”控制器模块

整体结构

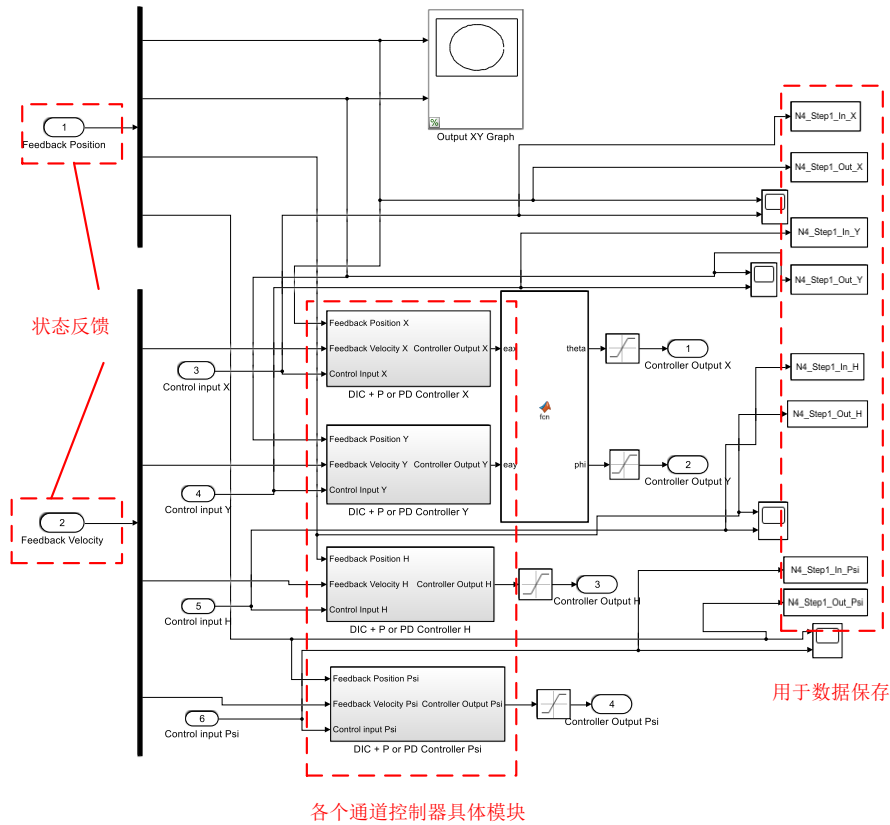


图 3 控制器模块

控制器具体结构

四个通道的 PID 控制器具体结构基本一致，以水平前向通道控制器为例

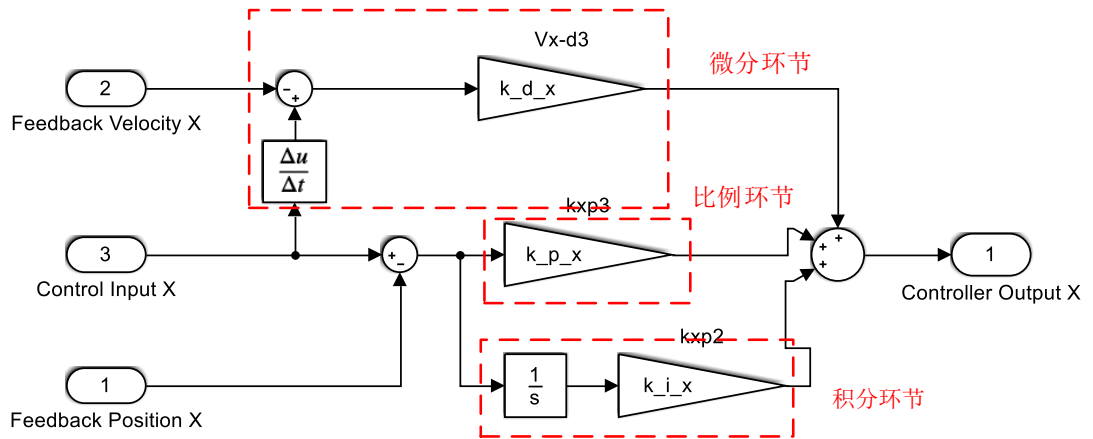


图 4 水平前向通道的 PID 控制器具体结构

三通道模型

	控制器	期望
高度通道	$u_T(t) = -k_{p_z p}(p_{z_e}(t) - p_{z_{ed}}(t)) - k_{p_z d}(\dot{p}_{z_e}(t) - \dot{p}_{z_{ed}}(t)) - k_{p_z i} \int_0^t (p_{z_e}(\tau) - p_{z_{ed}}(\tau)) d\tau$	$p_{z_{ed}} \in \mathbb{R}$

偏航通道	$\begin{aligned} u_{\omega_z}(t) = & -k_{\psi p}(\psi(t) - \psi_d(t)) - k_{\psi d}(\omega_z(t) - \dot{\psi}_d(t)) \\ & - k_{\psi i} \int_0^t (\psi(\tau) - \psi_d(\tau)) d\tau \end{aligned}$	$\psi_d \in \mathbb{R}$
水平通道	$\begin{aligned} \mathbf{u}_h(t) = & -\mathbf{K}_{hp} \mathbf{R}_{\psi}^T(\mathbf{p}_h(t) - \mathbf{p}_{hd}(t)) - \mathbf{K}_{hd} \mathbf{R}_{\psi}^T(\dot{\mathbf{p}}_h(t) - \dot{\mathbf{p}}_{hd}(t)) \\ & - \mathbf{K}_{hi} \int_0^t \mathbf{R}_{\psi}^T(\mathbf{p}_h(\tau) - \mathbf{p}_{hd}(\tau)) d\tau \end{aligned}$	$\mathbf{p}_{hd} \in \mathbb{R}^{2 \times 2}$

多旋翼模型模块

Sim1.0 (系统辨识得到的各通道传递函数模型)

Sim2.0 (使用辨识参数的多旋翼机理模型)

2. 实验效果

通过示波器显示出仿真模型路径跟随控制器响应时间的曲线，输出轨迹与期望轨迹的对比图。

3. 文件目录

例程目录: [\[安装目录\]\RflySimAPIs\6.RflySimExtCtrl\1.BasicExps\e7_MutUAVRemoteCtrl\4.TrajectoirePlanning\e4.1\](#)

	文件夹/文件名称	说明
Sim1.0	e4_1_TF_TrajectoirePlanning_code.m	控制器参数文件
	satgd.m	饱和函数文件
	e4_1_TF_TrajectoirePlanning_Segmentr2017b.slx	多旋翼路径跟踪 simulink 仿真 1.0 文件
	startSimulation.m	初始化参数文件
sim2.0	e4_1_Model_TrajectoirePlanning_code.m	控制器参数文件
	satgd.m	饱和函数文件
	e4_1_Model_TrajectoirePlanning_Segmentr2017b.slx	多旋翼路径跟踪 simulink 仿真 2.0 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		
3	MATLAB 2022b 及以上版本		

① : 推荐配置请见: <https://rflysim.com/doc/zh/HowToInstall.pdf>

5. 实验步骤

5.1. 必做实验：基于多旋翼线性模型的数值仿真 1.0(sim1.0)

Step 1: 参数初始化

打开本章所附带文件夹中的“e4\4.1\sim1.0”文件夹，打开“e4\4.1\sim1.0\startSimulation.m”文件，并单击“运行”按钮，初始化参数。

打开“e4\4.1\sim1.0\ e4_1_TF_TrajectoirePlanning_Segmentr2017b.slx”模型文件，将看到多旋翼路径跟踪 Simulink 仿真平台，如图 7.2

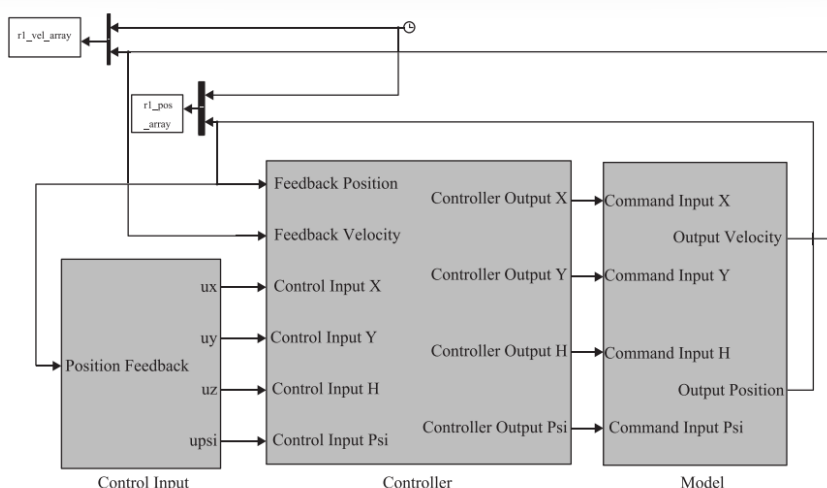


图 7.2 整体模型，Simulink 模型详见“e4\4.1\sim1.0\ e4_1_TF_TrajectoryPlanning_Segment.slx”

所示。

Step 2: 期望控制输入

打开控制输入模块，其内部结构如图 7.3 所示。进一步打开路径跟随控制器模块，可以看到如图 7.4 所示的内部结构。对于四通道中的每个通道，都设计了独立的路径跟随控制器。由于四个通道的设计几乎相同，这里仅以水平前向通道为例进行介绍。图 7.4 中加入了位置反馈，并且可以设置多旋翼需要跟随路径的初始坐标与终点坐标。设置完之后，将它们传入路径跟随控制器，产生位置期望，输出到路径跟踪控制器模块。此外这里还需要一些数据的保存与显示。关于路径跟随控制器的内部设计，可以单击该模块查看完整代码，其中的关键代码如表 7.1 所示，对应实验原理中的式(7.2)和式(7.15)。

表 7.1 路径跟随控制器关键代码

```
1 P_perp = P_final+(P_init-P_final)*((P_static-P_final)'*(P_init-P_final))/(norm(P_
   final-P_init + 0.000001))^2;
2 temp = k1_x*satgd(((k0_x/k1_x)*(P_final - P_static)+(P_perp-P_static)),a0);
3 P_d = P_static + temp;
```

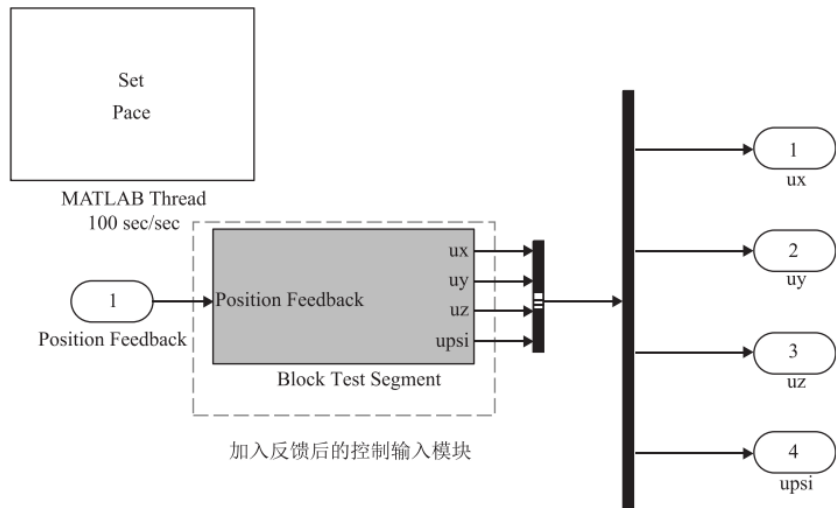


图 7.3 控制输入模块

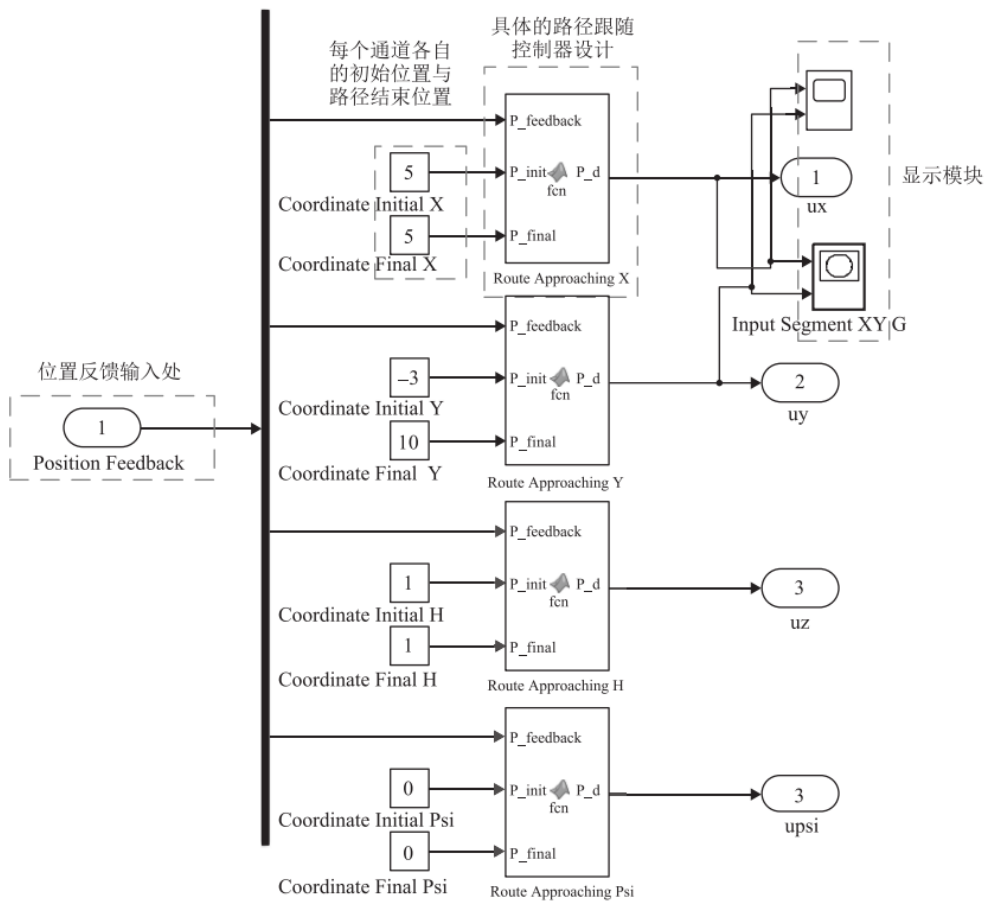


图 7.4 路径跟随控制器示意图

Step 3: 运行仿真并观测控制器响应

单击 Simulink “开始仿真” 按钮进行仿真，水平前向通道以及水平侧向通道的路径跟随控制器响应时间曲线如图 7.5 所示。从图中可以看出，多旋翼水平两通道轨迹都是从初始位置开始，然后逐渐逼近期望的轨迹，最终到达目标终点。此外，路径跟随控制器输出轨迹与期望轨迹对比图如图 7.6 所示。从图中可以看到，从初始位置开始，多旋翼非常迅速地逼近了期望轨迹，可以认为路径跟随控制器的设计是可行的。

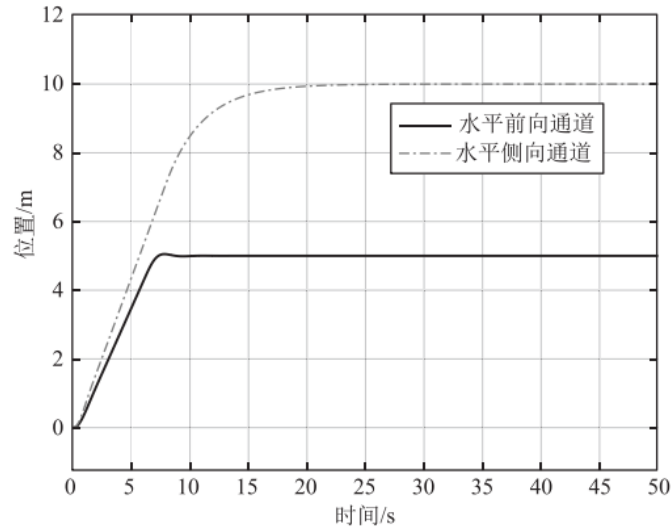


图 7.5 路径跟随控制器响应时间曲线

Step 4: 对比实际轨迹与期望轨迹

打开“e4\4.1\sim1.0\Plot.m”文件，并单击“运行”按钮，开始画图。

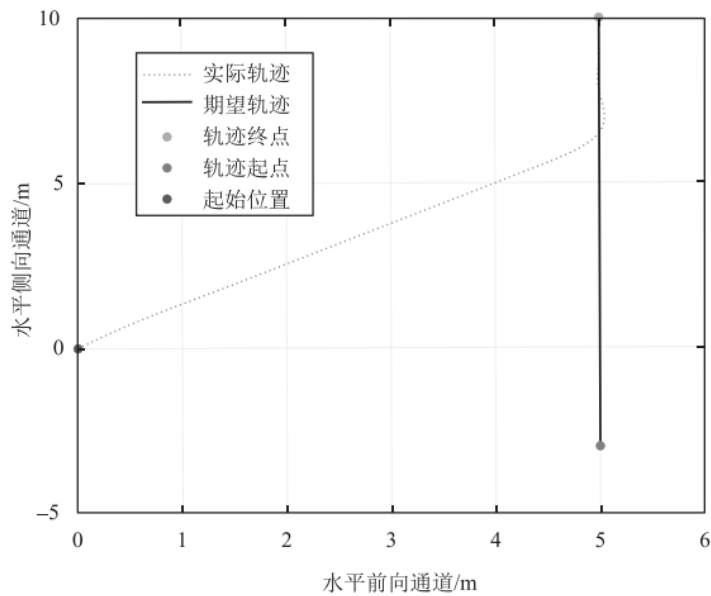


图 7.6 输出轨迹与期望轨迹对比图

5.2. 必做实验：基于多旋翼六自由度模型的仿真 2.0(sim2.0)

基于设计模型，已经完成了仿真 1.0，下一步需要在非线性模型上进行验证，即仿真 2.0。

Step 1: 参数初始化

打开本章所附带文件夹中的“e4\4.1\sim2.0”文件夹，打开“e4\4.1\sim2.0\startSimulation.m”文件，并单击“运行”按钮，初始化参数。

Step 2: 仿照 sim1.0 运行仿真

按照[仿真 1.0 中的步骤](#) step2~3 完成实验。

Step 3: 与仿真 1.0 对比路径跟随效果

打开“e4\4.1\sim2.0\Plot.m”文件，并单击“运行”按钮，开始画图。

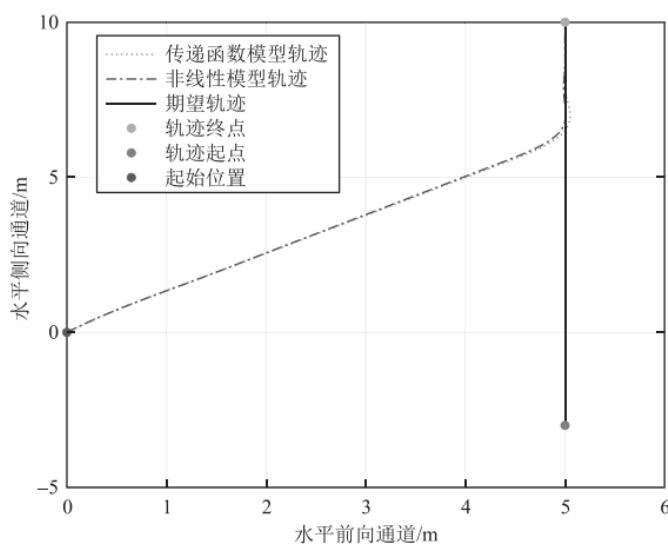


图 7.7 不同模型跟随轨迹对比图

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: