

轨迹跟踪基础实验

1. 实验目的

了解给定的多旋翼三通道线性化传递函数仿真模型和相应的轨迹跟踪控制器，进行轨迹跟踪。

2. 实验要求

- 软件要求：Windows 10及以上版本；RflySim工具链^[1]。
- 硬件要求：笔记本/台式电脑1台^[2]。

3. 实验地址

例程路径：[\[安装目录\]\RflySimAPIs\6.RflySimExtCtrl\1.BasicExps\e9_TrajectoireFollowing_Segment\e3.1](#)

- [sim1.0\Save_Cercle.m](#)：保存数据文件
- [sim1.0\startSimulation.m](#)：启动仿真文件
- [sim1.0\e3_1_TF_TrajectoireFollowing_Segment.slx](#)：Simulink仿真文件（线性模型）
- [sim1.0\N4_Step1_cercle_comparaison_T10tf.mat](#)：仿真结果数据文件
- [sim2.0\Save_Cercle.m](#)：保存数据文件
- [sim2.0\startSimulation.m](#)：启动仿真文件
- [sim2.0\Plot_Compare_cercle.m](#)：绘制比较圆轨迹文件
- [sim2.0\e3_1_Model_TrajectoireFollowing_Segment.slx](#)：Simulink仿真文件（非线性模型）
- [sim2.0\N4_Step1_cercle_comparaison_T10.mat](#)：仿真结果数据文件
- [sim2.0\N4_Step1_cercle_comparaison_T10tf.mat](#)：仿真结果对比数据文件

4. 实验内容或步骤

4.1 步骤1：基于多旋翼线性模型的数值仿真1.0(sim1.0)

打开本章所附带文件夹中的"e3\e3.1\sim1.0"文件夹，打开"e3\e3.1\sim1.0\startSimulation.m"文件，并单击"运行"按钮，初始化参数。

打开Simulink

文件"e3\e3.1\sim1.0\e3_1_TF_TrajectoryFollowing_Segment_2017b.slx"，整体模块如图所示。

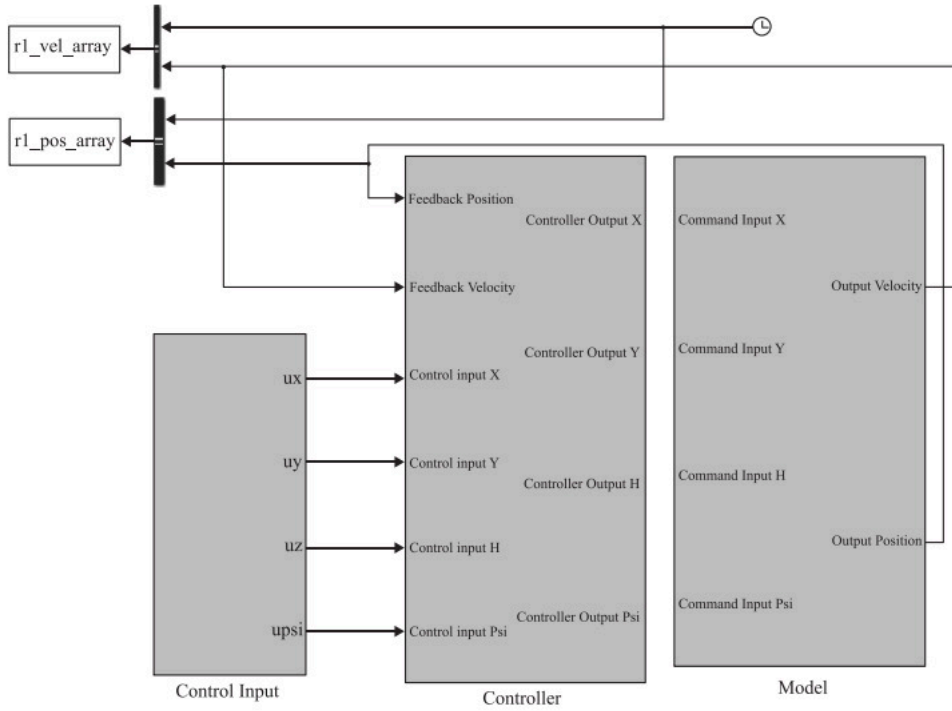
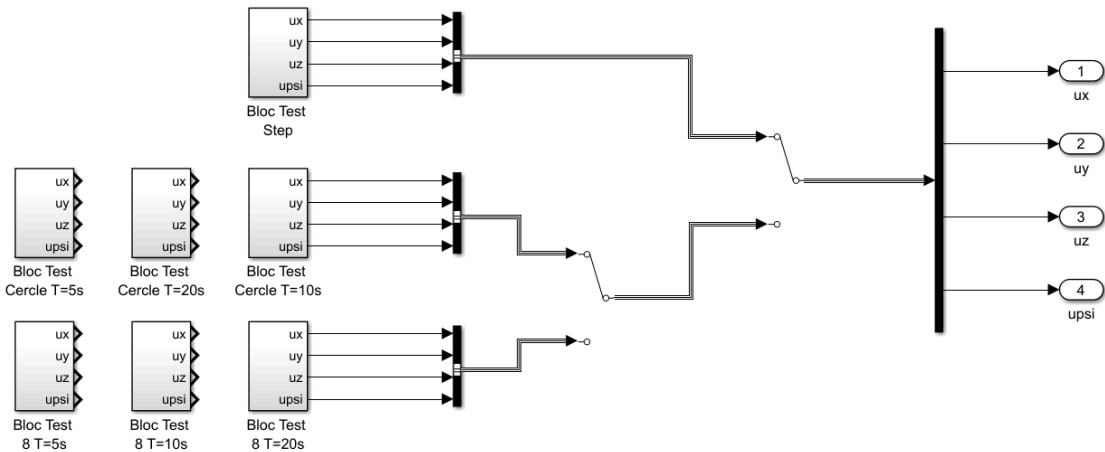
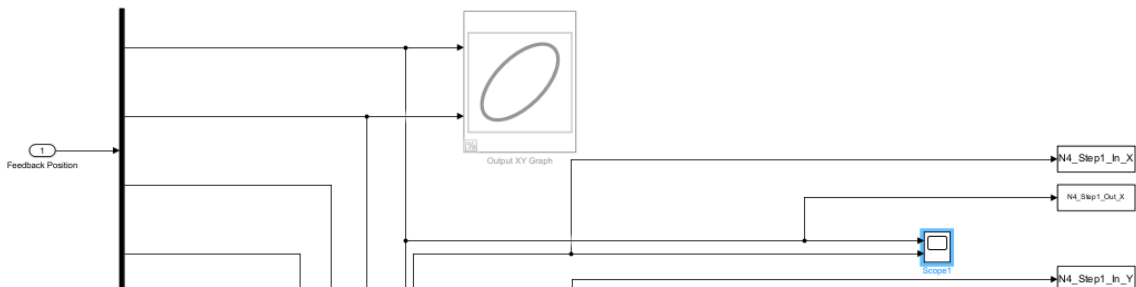


图 6.2 模块整体示意图, Simulink 模型详见 “e3_1_TF_TrajectoryFollowing_Segment.slx”

将"ControlInput"模块的按图示接入，可以得到各通道的阶跃响应输入输出时间曲线对比图。



以水平前向通道为例，仿真结果如图所示。



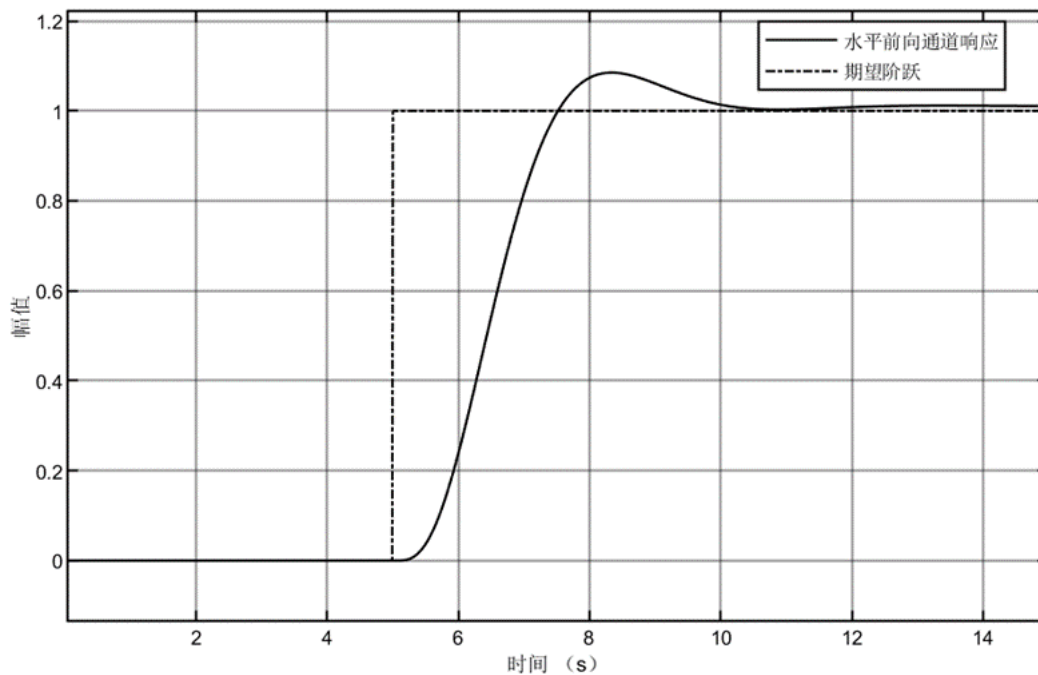
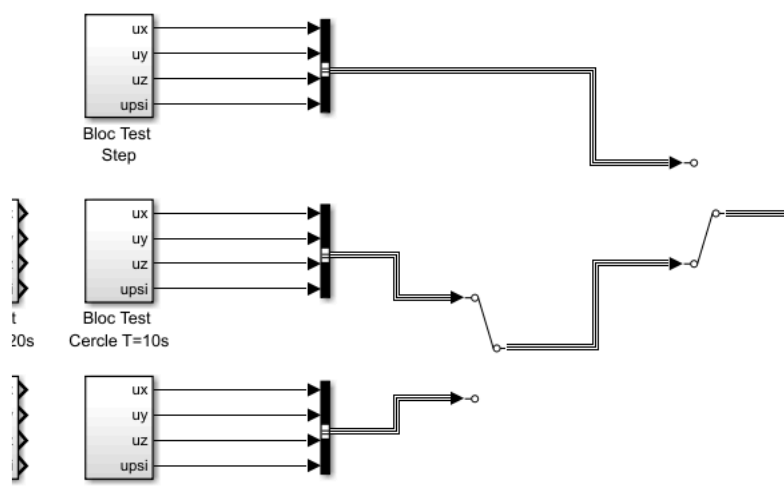


图 5期望阶跃的水平前向通道响应

将"Control Input"模块中的拨杆如图进行连接，即为圆轨迹追踪。



水平前向通道响应可以通过示波器Scope1观察

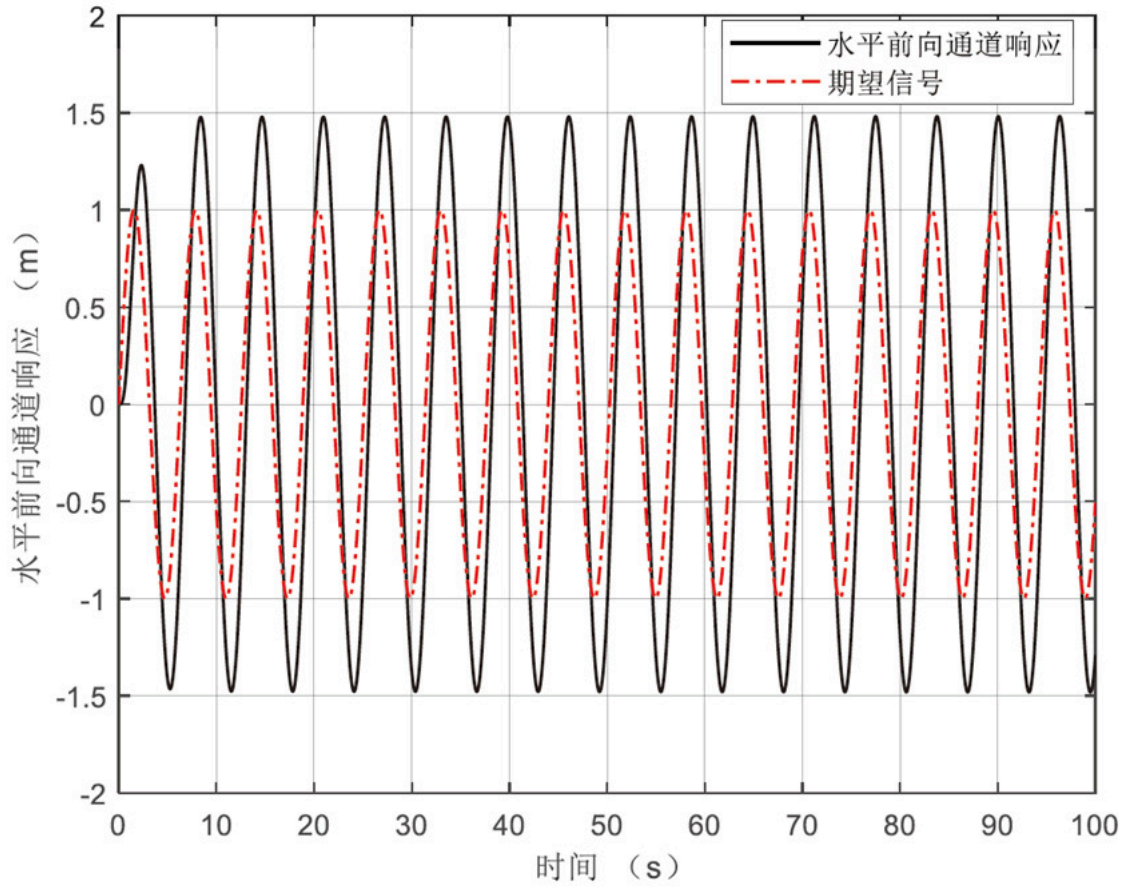
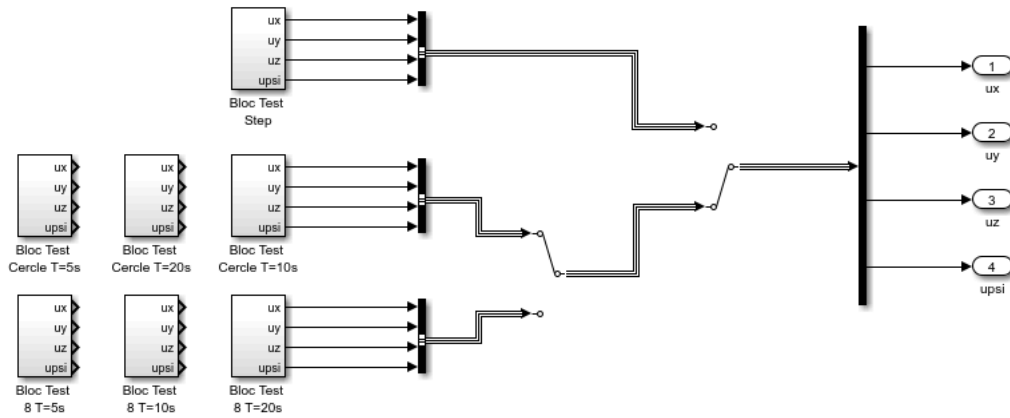


图 6圆轨迹跟踪水平前向通道 $T = 5s$ 输入输出对比图

将"Control

Input"模块的三种不同周期的圆轨迹信号接入，以水平前向通道为例，其仿真结果如图所示。



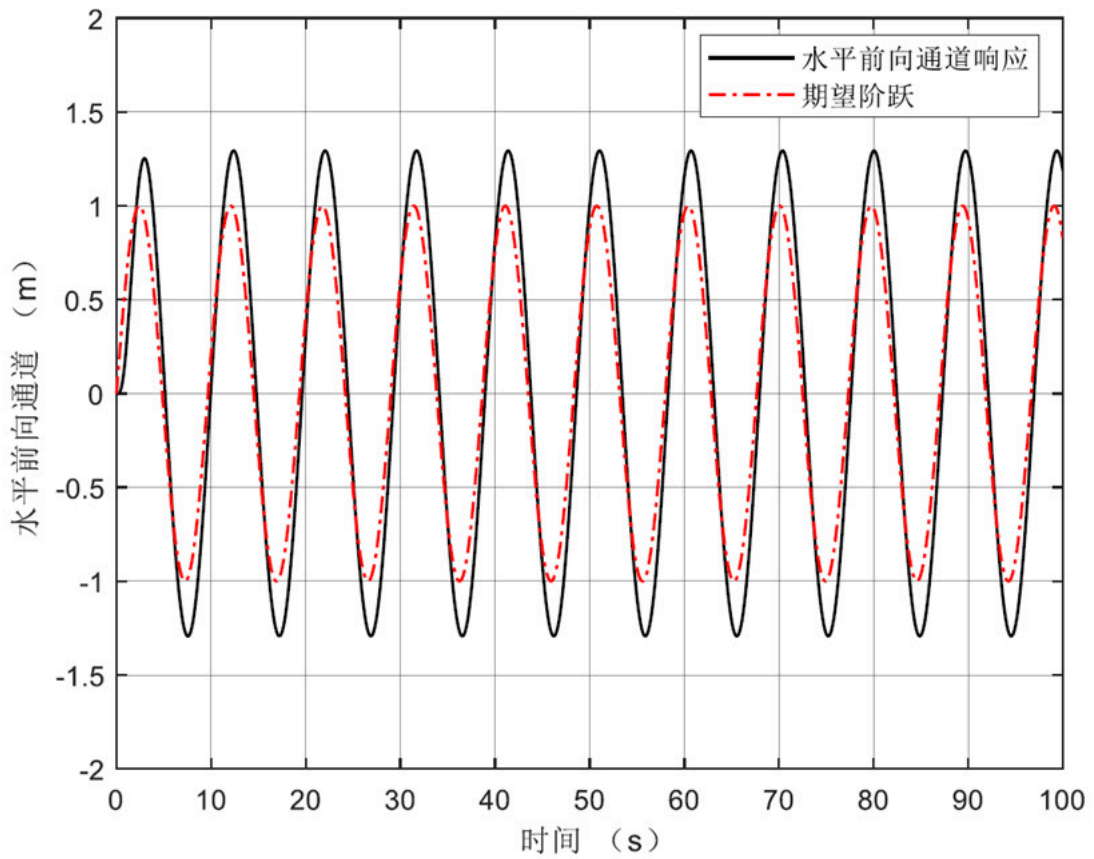


图 7圆轨迹跟踪水平前向通道 $T = 10s$ 输入输出对比图

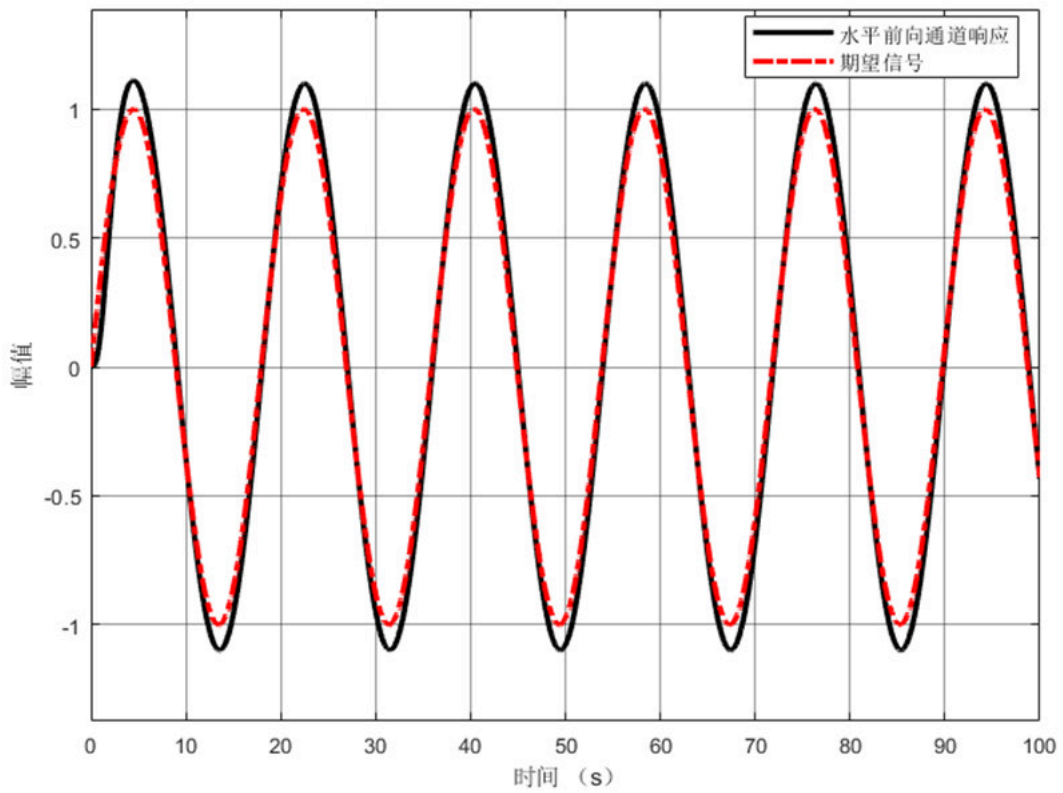
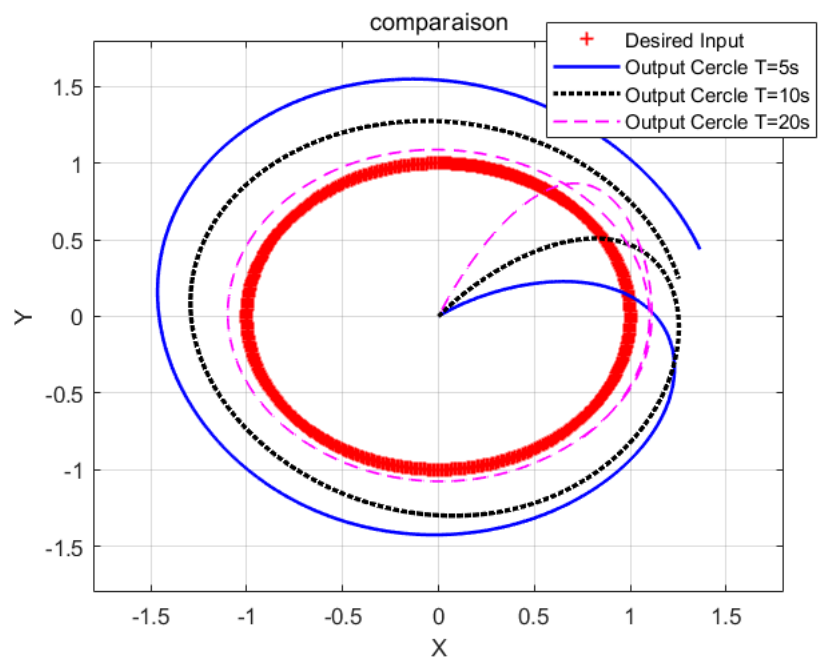
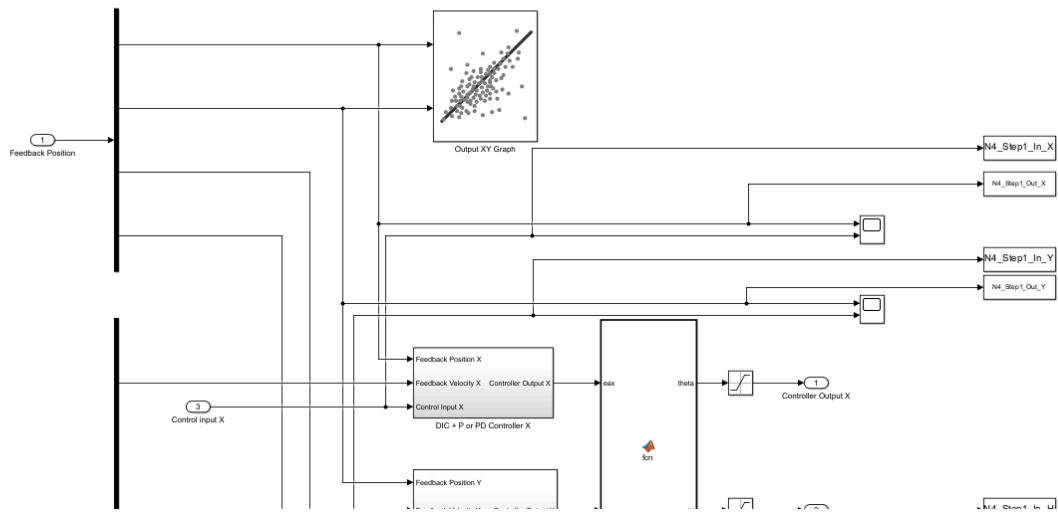
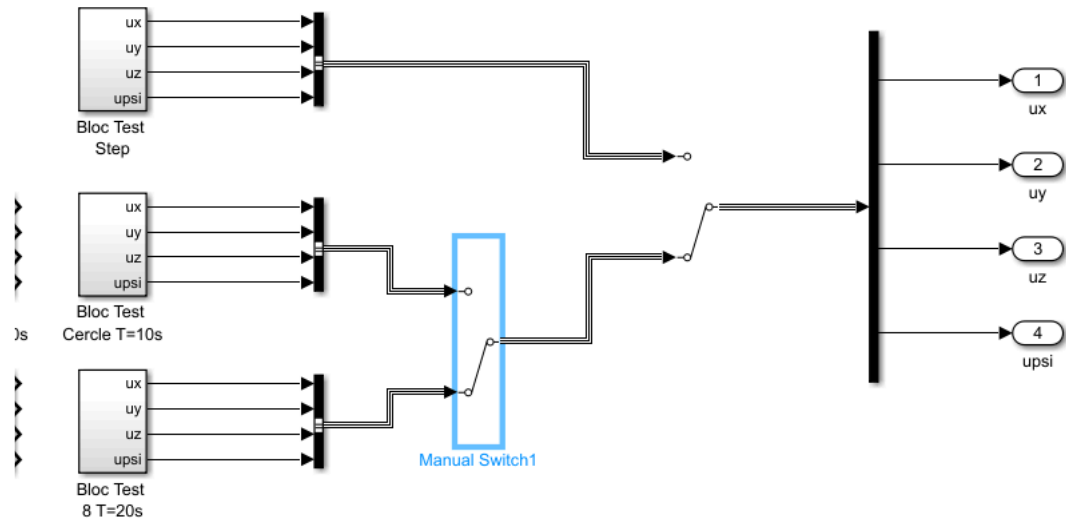


图 8圆轨迹跟踪水平前向通道 $T = 20s$ 输入输出对比图

为获得更好的对比效果，将模型水平前向和侧向响应接入Output XY Graph，点击运行，将不同周期输入激励下的水平两方向通道响应画到一张图中，如图所示。可以看出，随着周期的增大，多旋翼对期望曲线的跟踪效果更加理想。



将"Control Input"模块中的拨杆如图进行连接，即为"8"字轨迹追踪。



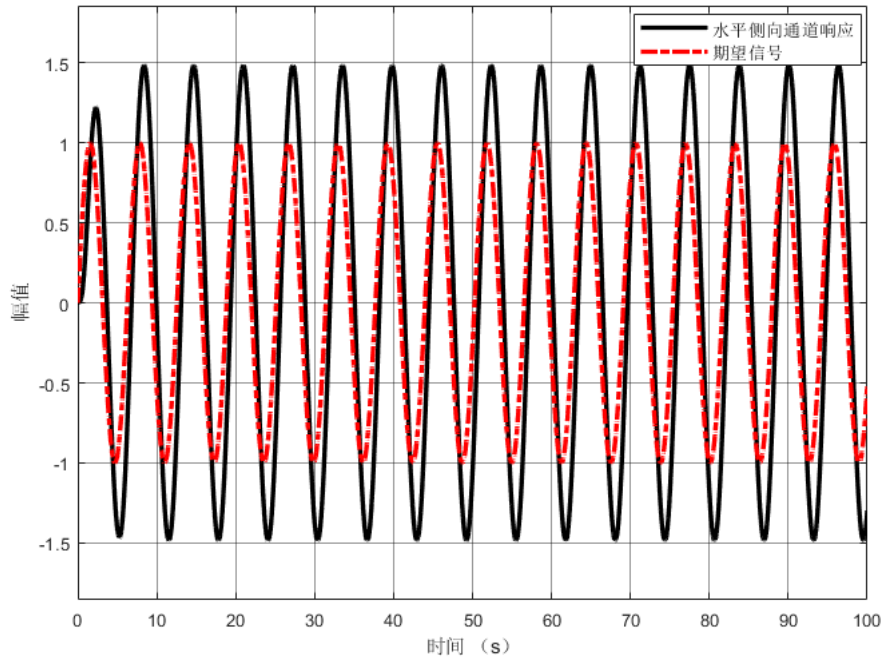


图 9 "8"字轨迹跟踪水平前向通道 $T = 5s$ 输入输出对比图

将 "Control

Input" 模块的三种不同周期的 "8" 字轨迹信号接入，以水平前向通道为例，其仿真结果如下对比图所示。

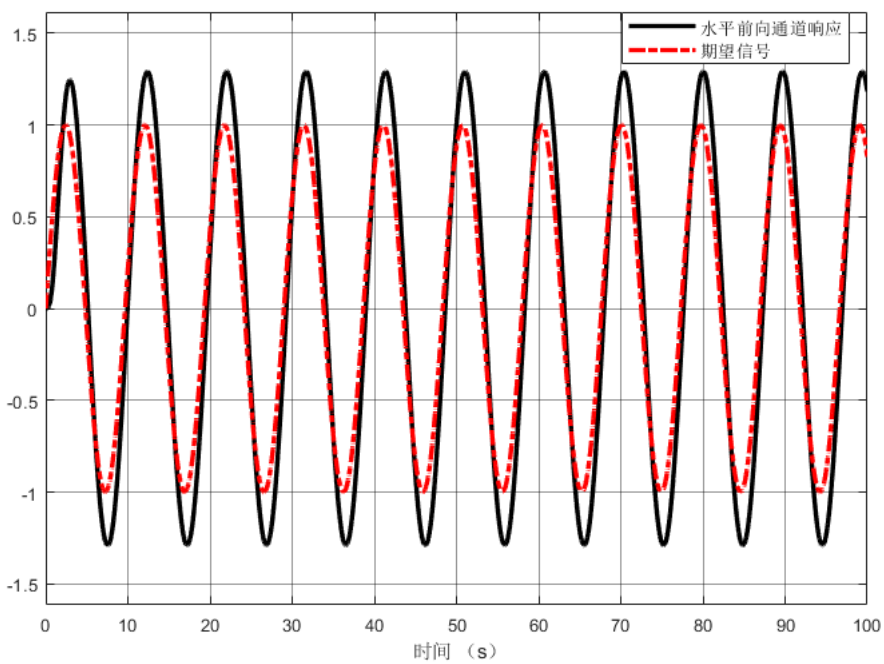
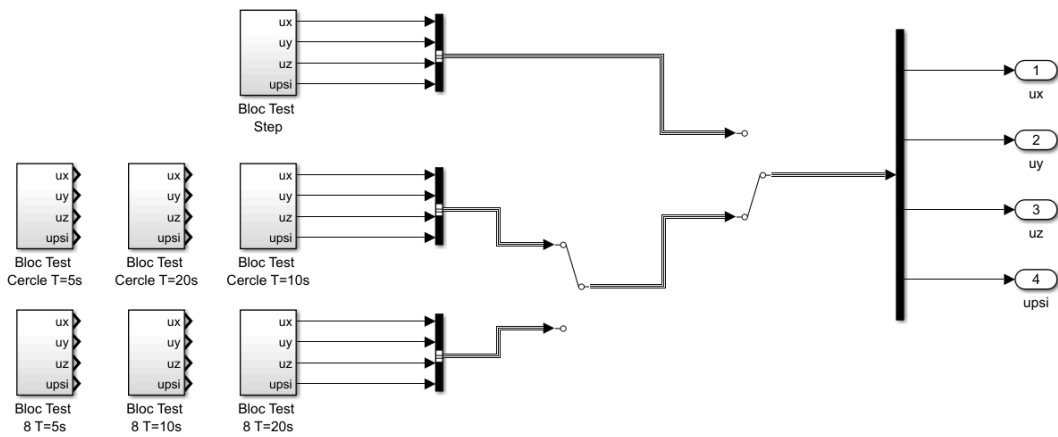


图 10"8"字轨迹跟踪水平前向通道T = 10s 输入输出对比图

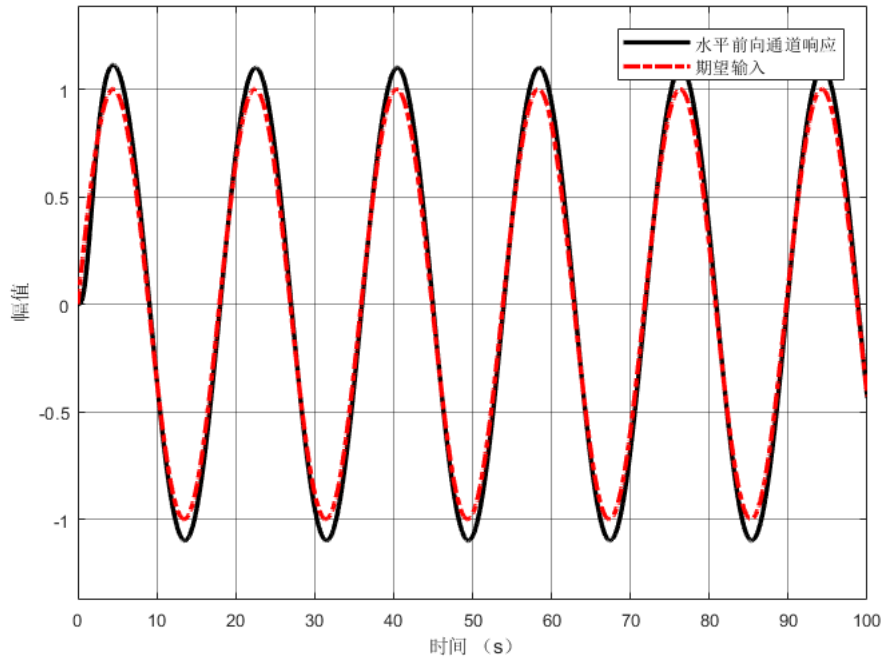


图 11"8"字轨迹跟踪水平前向通道T = 20s 输入输出对比

为获得更好的对比效果，将模型水平前向和侧向响应接入Output XY Graph。如轨迹对比图所示，可以看出，随着周期的增大，多旋翼对期望曲线的跟踪效果更加理想。

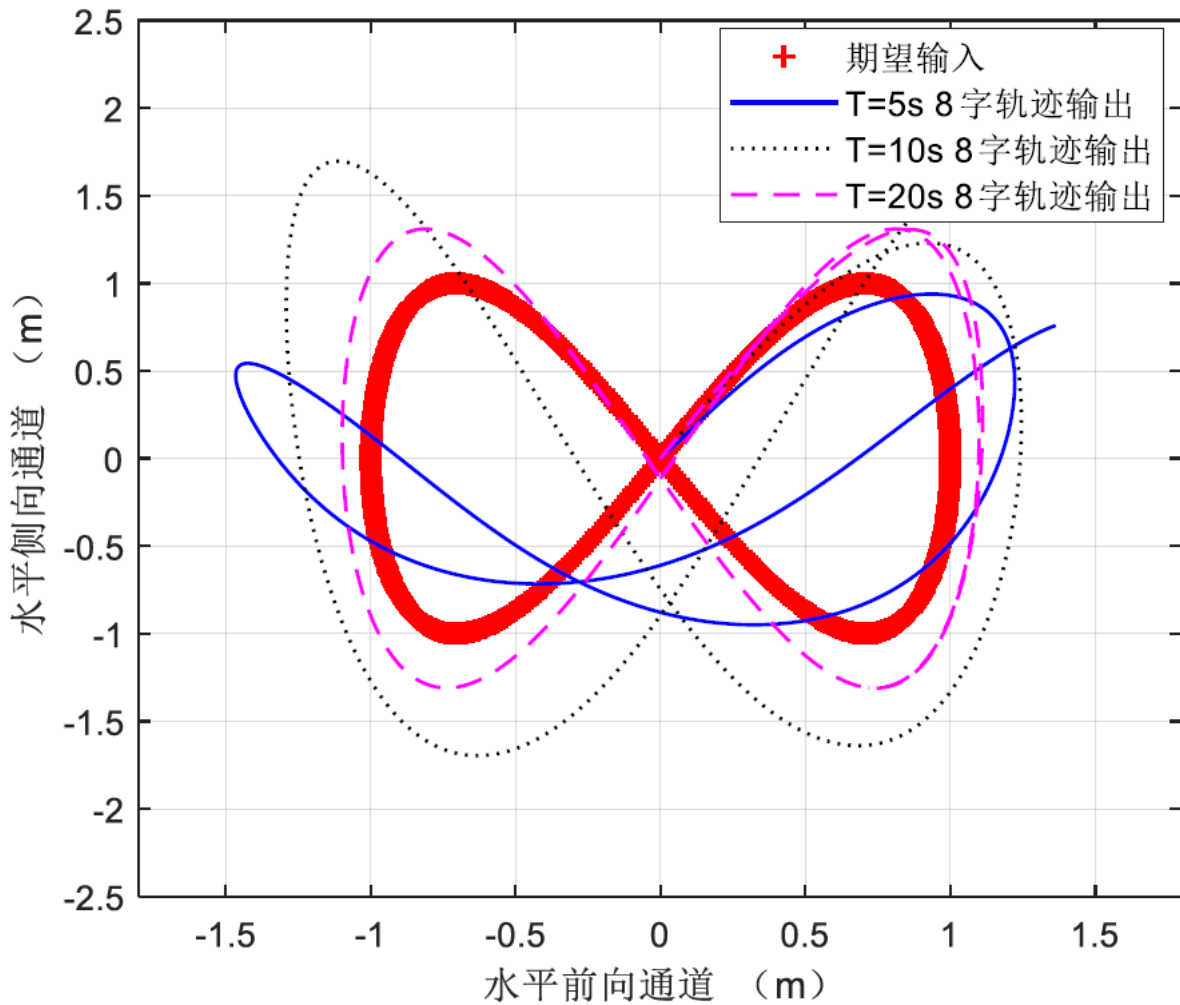


图 12不同周期"8"字轨迹跟踪曲线对比图

4.2 步骤2: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。

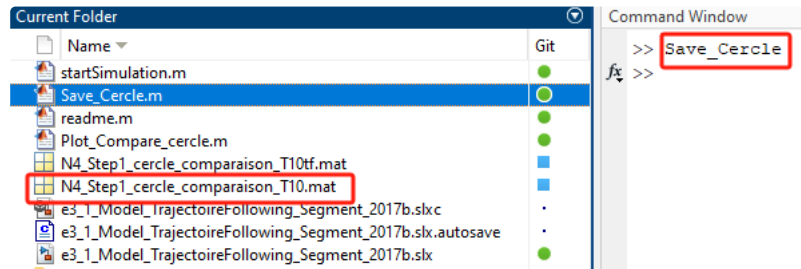
打开

Simulink文件"e3\e3.1\sim2.0"，其中的闭环系统结构与"e3\e3.1\sim1.0"相同，区别在于其中的模型为非线性模型。采用与传递函数模型实验中完全相同的实验步骤，将两种模型的关键结果进行对比。对比结果图。

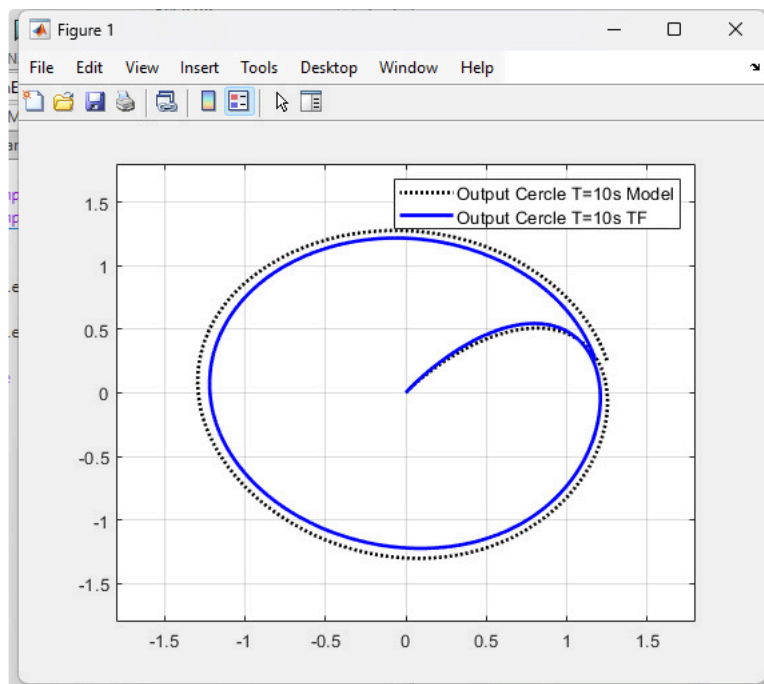
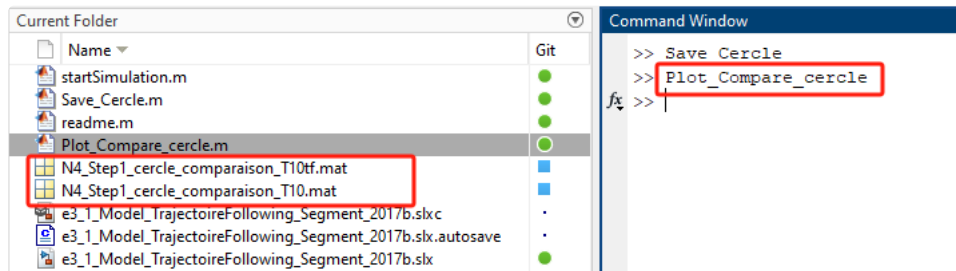
打开本章所附带文件夹中的"e3\e3.1\sim2.0"文件夹，打开"e3\e3.1\sim2.0\startSimulation.m"文件，并单击"运行"按钮，初始化参数。

按照 [仿真1.0中的步骤 step3](#)完成实验。

以 $T = 10s$ 圆轨迹期望输入，运行sim2.0\e3_1_Model_TrajectoireFollowing_Segment_2017b.slx完成后，运行sim2.0\Save_Cercle.m保存水平两方向通道响应数据（同样的方法保存sim1.0的数据）



此时运行sim2.0\Plot_Compare_cercle.m加载sim1.0和sim2.0保存的水平两方向通道响应数据，绘制圆轨迹



对比结果图可以看出，我们采用系统辨识获得的模型进行跟踪控制器设计，可以产生与基于非线性模型设计非常接近的效果。

同时也可以可以在RflySim3D中查看仿真飞行效果。

5. 关键知识点

关键知识点1: 实验整体流程

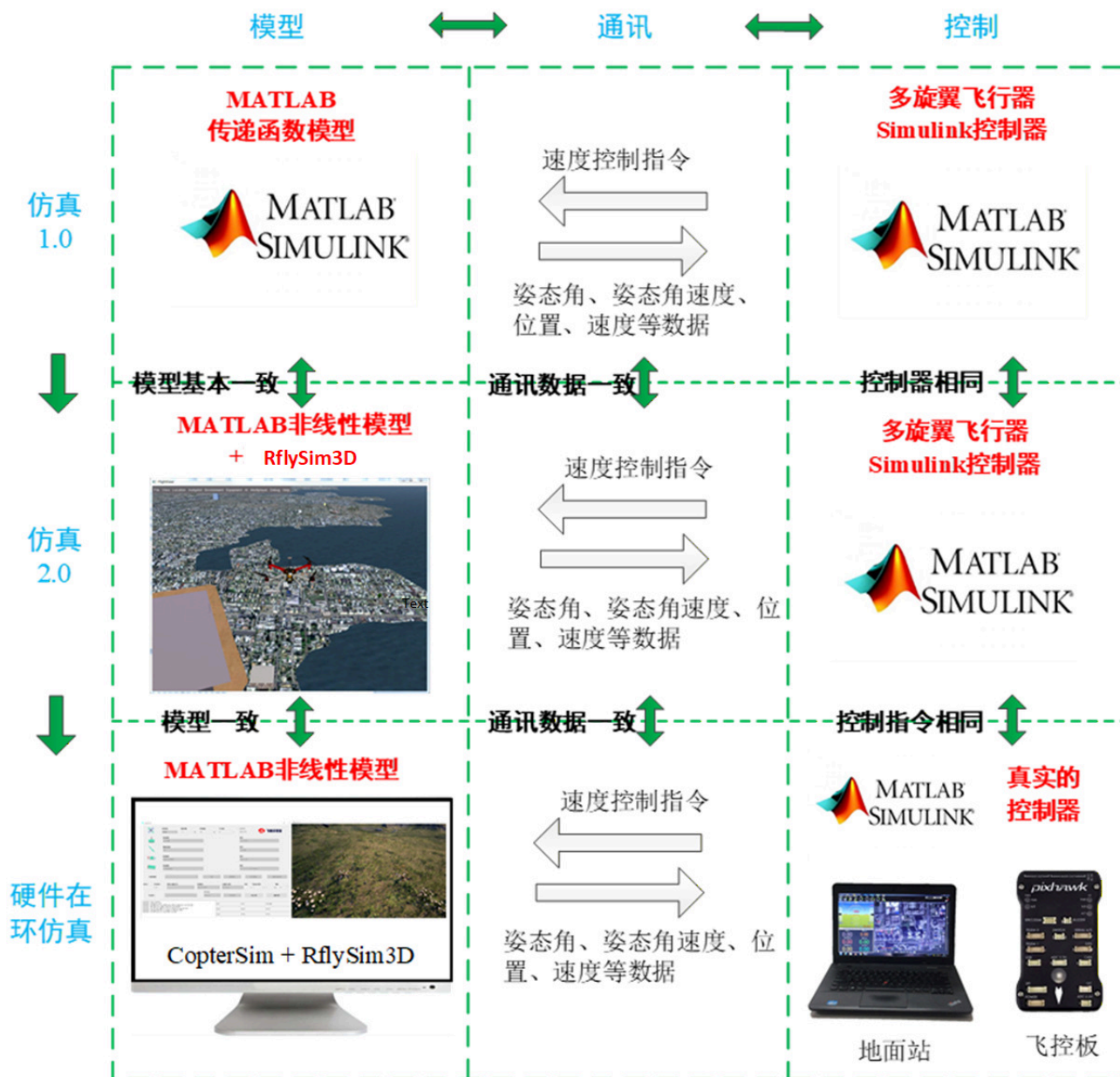
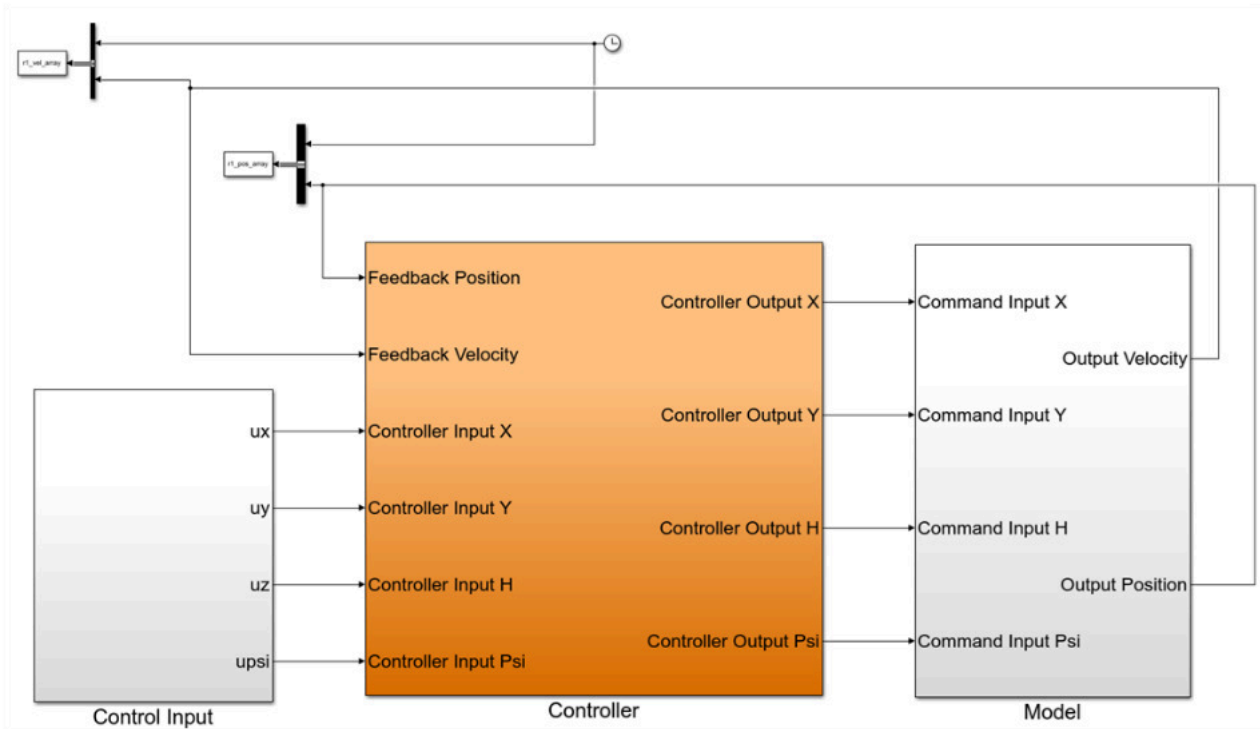


图 仿真阶段流程

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: 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。

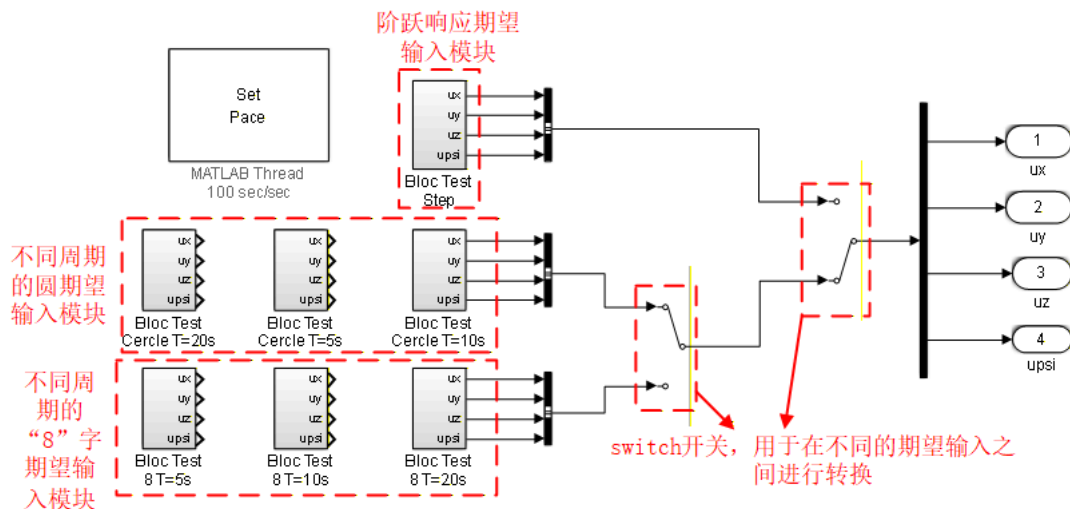


图 1期望输入模块

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

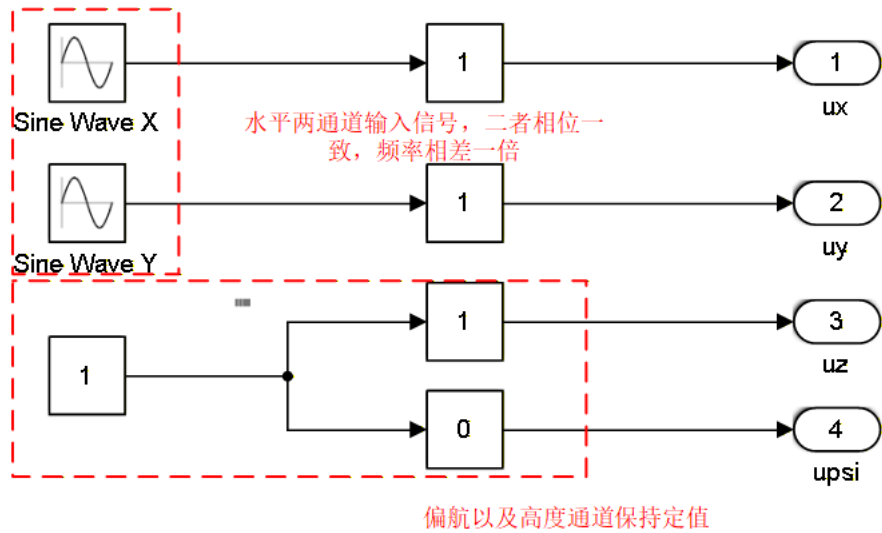


图 2"8"字轨迹期望输入

"Controller"控制器模块

整体结构

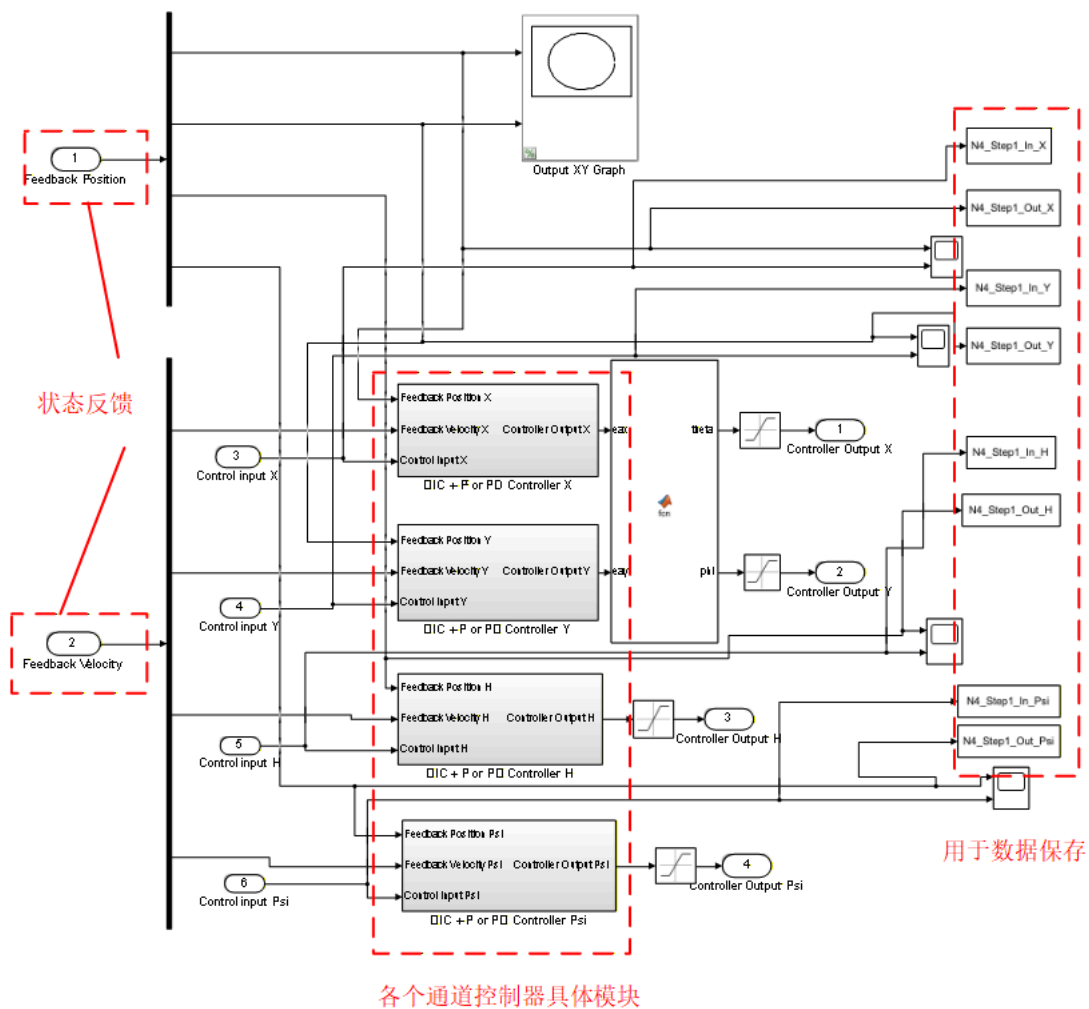


图 3控制器模块

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

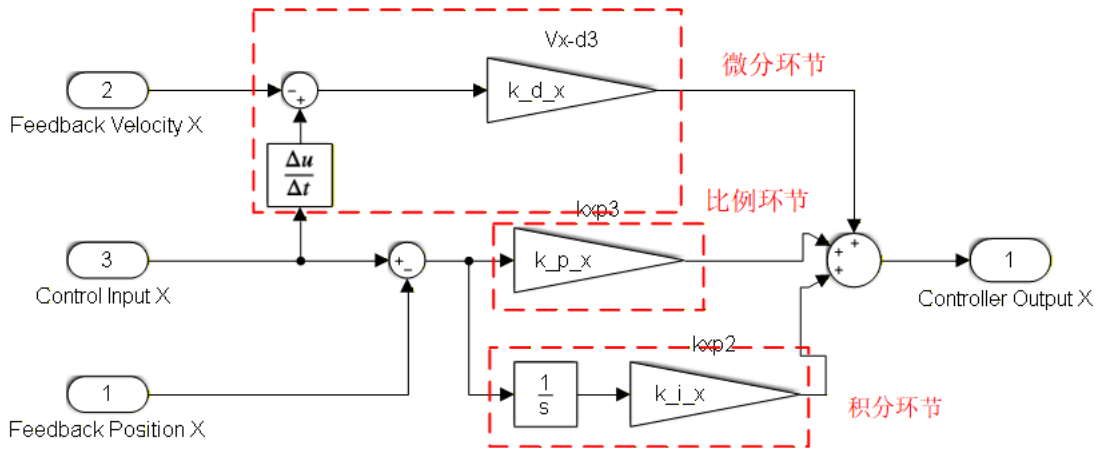


图 4水平前向通道的PID 控制器具体结构

三通道模型

	控制器	期望
高度通道	$u_T(t) = -k_{p_z} (p_{z_e}(t) - p_{z_e d}(t)) - k_{p_z d} (\dot{p}_{z_e}(t) - \dot{p}_{z_e d}(t)) - k_{p_z d d} \int_0^t (p_{z_e}(\tau) - p_{z_e d}(\tau)) d\tau$	
偏航通道	$u_{\omega_z}(t) = -k_{\psi p} (\psi(t) - \psi_d(t)) - k_{\psi d} (\omega_z(t) - \dot{\psi}_d(t)) - k_{\psi d d} \int_0^t (\psi(\tau) - \psi_d(\tau)) d\tau$	
水平通道	$u_h(t) = -K_{hp} R_\psi^T (p_h(t) - p_{hd}(t)) - K_{hd} R_\psi^T (\dot{p}_h(t) - \dot{p}_{hd}(t)) - K_{hi} \int_0^t R_\psi^T (p_h(\tau) - p_{hd}(\tau)) d\tau$	

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)

6.参考资料

1. RflySim官方文档
2. 多旋翼飞行器轨迹跟踪控制技术
3. PID控制器在多旋翼飞行器中的应用
4. 系统辨识理论与实践
5. 非线性控制系统理论

7.常见问题

Q1: 在Simulink仿真过程中, 为什么会出现积分饱和现象, 如何解决?

A1: 积分饱和是指当系统存在较大偏差或偏差持续时间较长时, 积分项会不断累积, 导致控制器输出超出执行机构的物理限制范围。这可能导致系统响应变慢或不稳定。解决方法包括: 采用积分分离PID控制、变速积分、抗积分饱和和算法等。在本实验中, 可以通过调整积分增益参数来缓解这一问题。

Q2: 仿真1.0和仿真2.0的结果存在差异是什么原因?

A2: 仿真1.0使用的是通过系统辨识得到的线性化传递函数模型, 而仿真2.0使用的是非线性机理模型。线性模型只是在特定工作点附近的近似, 无法完全反映系统的非线性特性。因此在某些情况下, 仿真结果会有差异。但通过合理设计控制器参数, 可以使线性模型的仿真结果与

非线性模型相近，这也验证了控制器的鲁棒性。

Q3: 如何评价轨迹跟踪控制器的性能?

A3: 轨迹跟踪控制器的性能可以从多个方面来评价:

- 跟踪精度: 实际轨迹与期望轨迹之间的误差大小
- 响应速度: 系统达到期望轨迹的速度
- 稳态误差: 系统稳定后仍存在的误差
- 稳定性: 系统在长时间运行中的稳定性
- 鲁棒性: 面对模型不确定性或外部干扰时的表现

在本实验中, 可以通过比较不同周期下圆轨迹和"8"字轨迹的跟踪效果来评估控制器性能。

1. <https://rflysim.com/> ↩

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