

轨迹跟踪控制分析实验

1. 实验目的

给定幅值为1、响应时间为仿真第5s的阶跃信号，观察仿真模型的各通道稳态误差、超调量和调节时间。根据所获得的结果，使用频率域方法设计控制器，满足如下性能要求。

- 1) 速度控制环，相位裕度 $>65^\circ$ ，截止频率 $> 3\text{rad/s}$;
- 2) 位置控制环，相位裕度 $>60^\circ$ ，截止频率 $> 2\text{rad/s}$ 。

系统具有较好的系统带宽，分析系统带宽对跟踪效果的影响，并使用设计好的校正控制器进行仿真对照。

2. 实验要求

- 软件要求：Windows 10及以上版本；RflySim工具链^[1]；MATLAB 2022b及以上版本。
若使用Pixhawk 6X飞控，平台安装时的编译命令为：px4_fmu-v6x_default，推荐PX4固件版本为：1.12.3。其他配套飞控及编译命令请见：<https://rflysim.com/doc/zh/1/Hardware.html>
- 硬件要求：笔记本/台式电脑1台^[2]。

3. 实验地址

例程目录：

[安装目录]\RflySimAPIs\6.RflySimExtCtrl\1.BasicExps\e9_TrajectoireFollowing_Segment\e3.2

sim1.0/Save_Cercle.m：保存数据文件

sim1.0/N4_Step1_cercle_comparaison_T10tf.mat：圆轨迹输入输出融合数据文件

sim1.0/e3_2_TF_TrajectoireFollowing_fc.slx：多旋翼轨迹控制器仿真1.0文件

sim1.0/startSimulation.m：初始化参数文件

sim1.0/readme.m：sim1.0文件夹说明文档

sim2.0/N4_Step1_cercle_comparaison_T10.mat：圆轨迹输入输出融合数据文件

sim2.0/N4_Step1_cercle_comparaison_T10tf.mat：圆轨迹输入输出融合数据文件

sim2.0/Plot_Compare_cercle.m：绘制圆形轨迹对比图

sim2.0/Save_Cercle.m：保存数据文件

sim2.0/e3_2_Model_TrajectoireFollowing_FC.slx：多旋翼轨迹控制器仿真2.0文件

sim2.0/startSimulation.m：初始化参数文件

sim2.0/readme.m：sim2.0文件夹说明文档

4. 实验内容或步骤

此处编写实验步骤，并添加相关图片。以及每一步设置结束后，具体的变化等等

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sim1.0文件夹，打开sim1.0/startSimulation.m文件，并单击"运行"按钮，初始化参数。

打开Simulink文件sim1.0/e3_2_TF_TrajectoireFollowing_fc_2017b.slx，整体模块如图所示，按如下步骤进行校正。（sim1.0/e3_2_TF_TrajectoireFollowing_fc_2017b.slx是一个校正完成的例子）

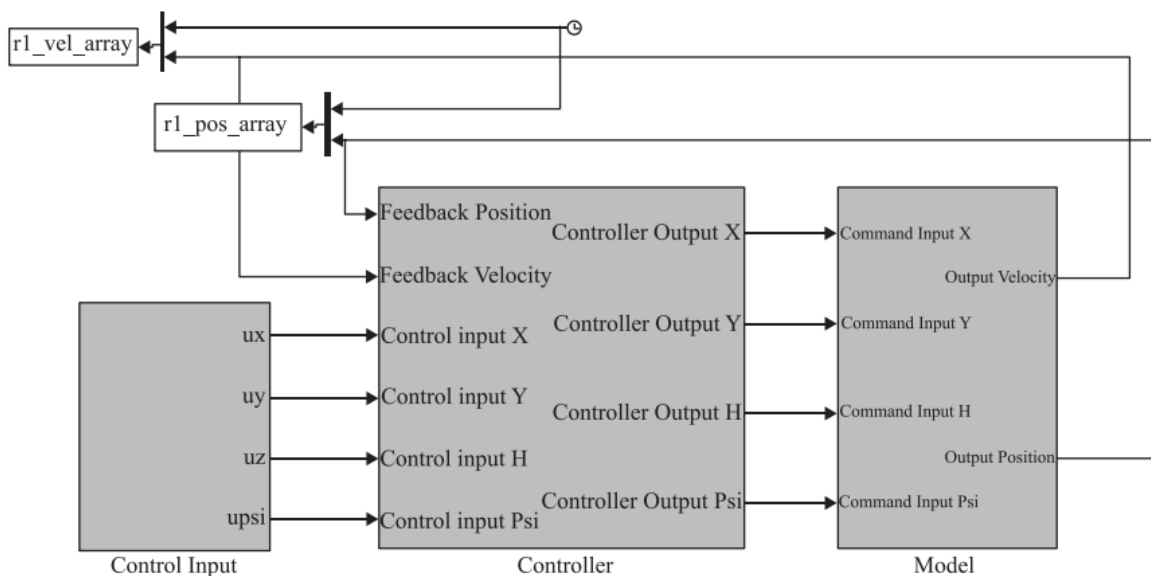


图 6.18 模块整体示意图，Simulink 模型详见“e3_2_TF_TrajectoryFollowing_Segment.slx”

步骤1.2：水平前向通道速度环校正

步骤1.2.1：开环线性分析

(1) 输入点设置:

选中信号线,单击鼠标右键,选择"Linear Analysis Points"- "Open-loop Input"。

(2) 输出点设置:

与输入点设置步骤相同，最后选择"Open-loop Output"。在Controller PIDadjusting中DIC+P or PD Controller设置好的输入点和输出点如图所示。

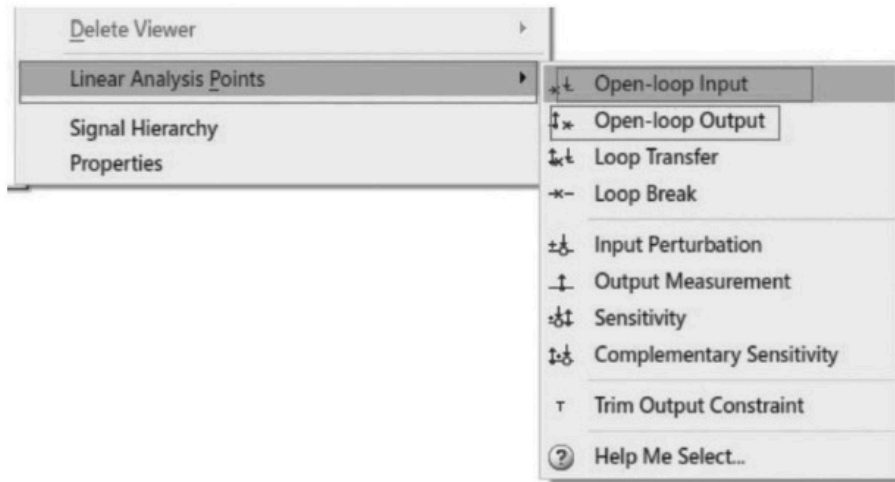


图 6.19 设计扫频输入点和输出点示意图

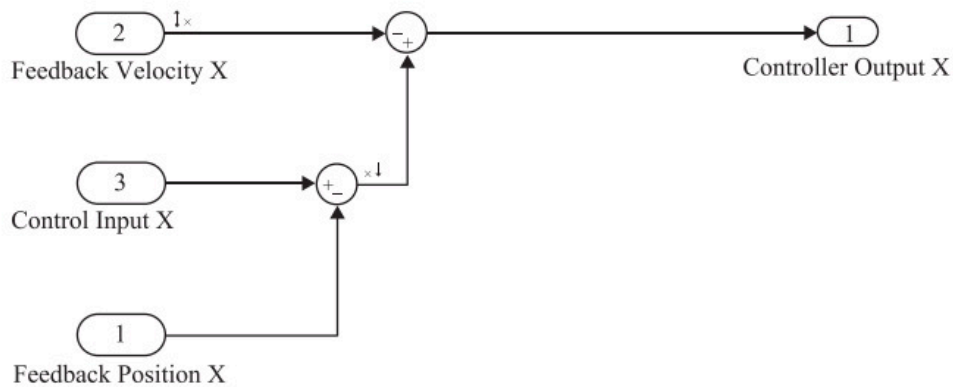


图 6.20 设置好的输入点和输出点

(3) 得到Bode图:

选择Simulink 上面菜单中的"Analysis"- "Control Design"- "Linear Analysis", 在弹出的窗口中选择"LINEAR ANALYSIS", 单击"Bode", 即可得到Bode图。右键单击曲线,选择"Characteristic"- "All Stability Margins", 可以看到截止频率、幅值裕度和相角裕度等。得到的速度环Bode图如图所示, 可以看到相角裕度为 81.6° , 截止频率为 1.33rad , 不符合要求。

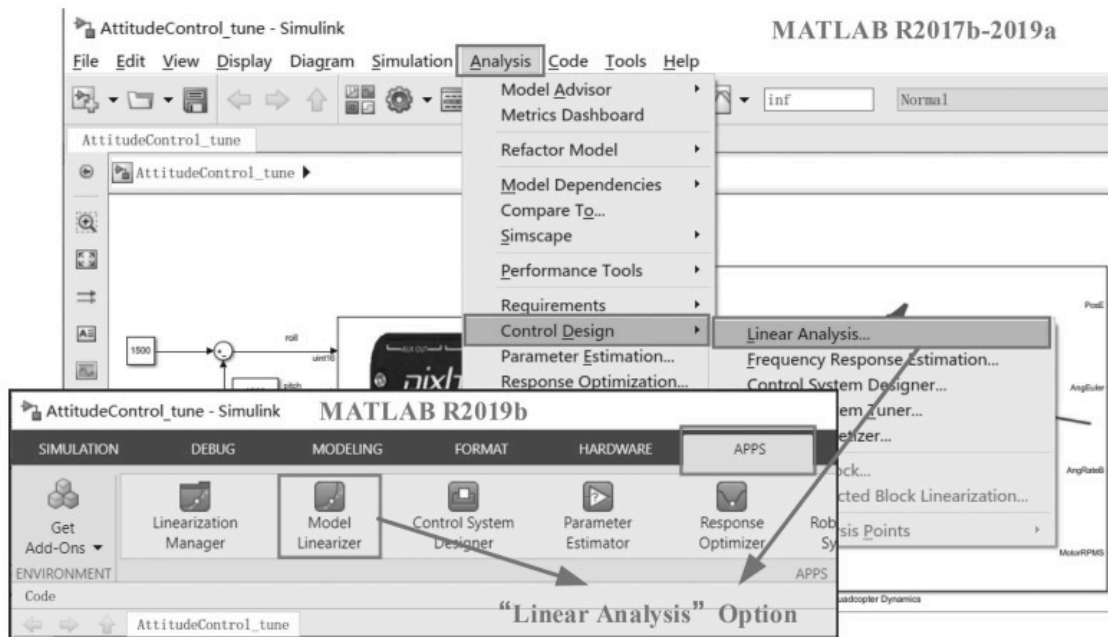


图 6.21 “Linear Analysis” 菜单路径

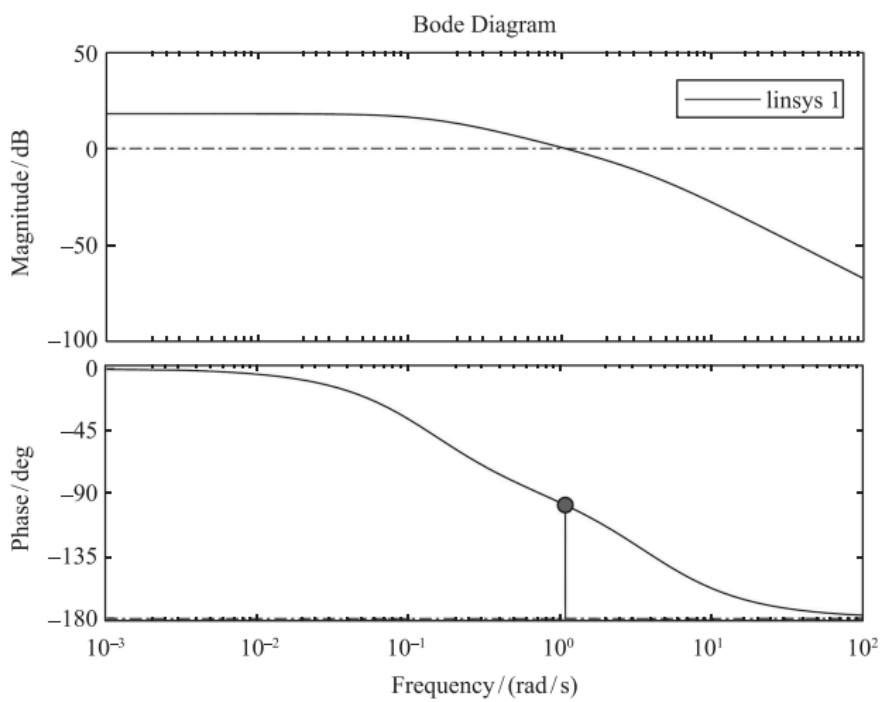


图 6.22 未加校正水平前向通道 Bode 图

新版本的MATLAB点击"APP"下的"模型线性化器":



步骤1.2.2: 根据bode图获取速度环的传递函数

生成 Bode 图后, 在左侧“Linear Analysis Workspace”中会出现“Linsys1”变量。如图 6.23所示, 进行操作后即可得到传递函数模型:

$$G(s) = \frac{4.2003}{(s + 3.75)(s + 0.1412)}$$

整理得

$$G(s) = \frac{7.554}{(0.267s + 1)(7.082s + 1)}$$

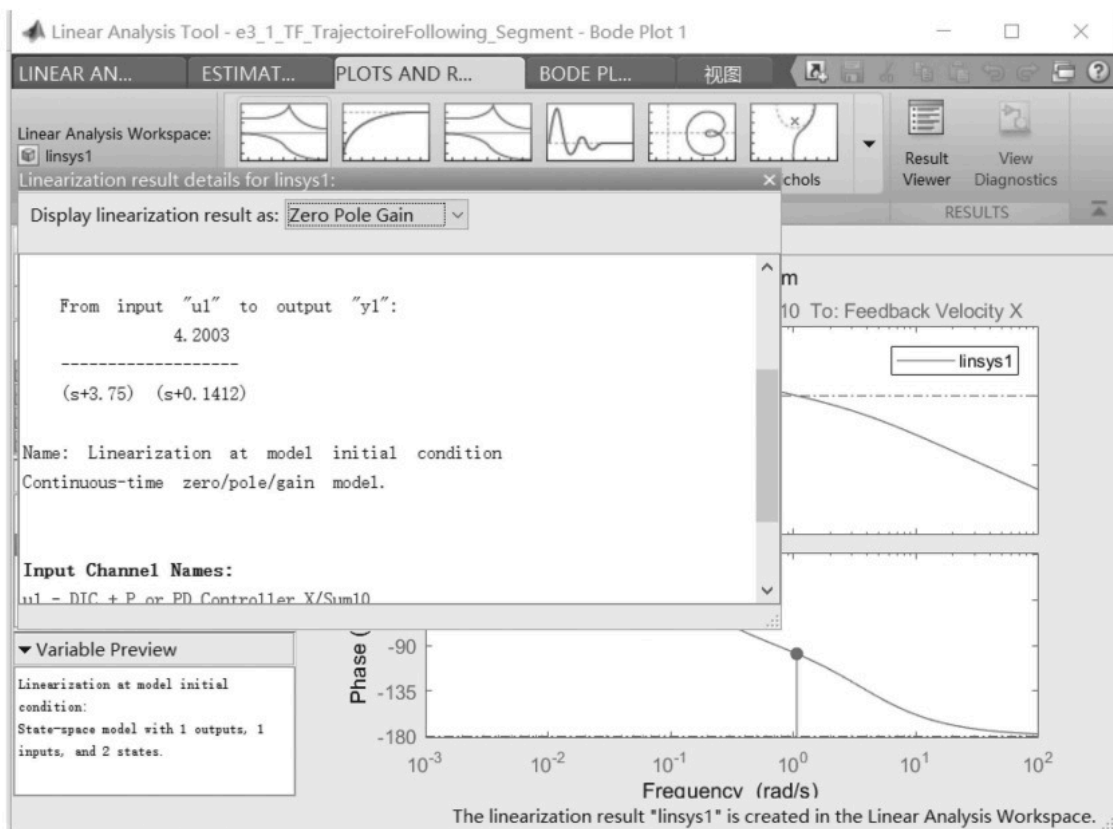
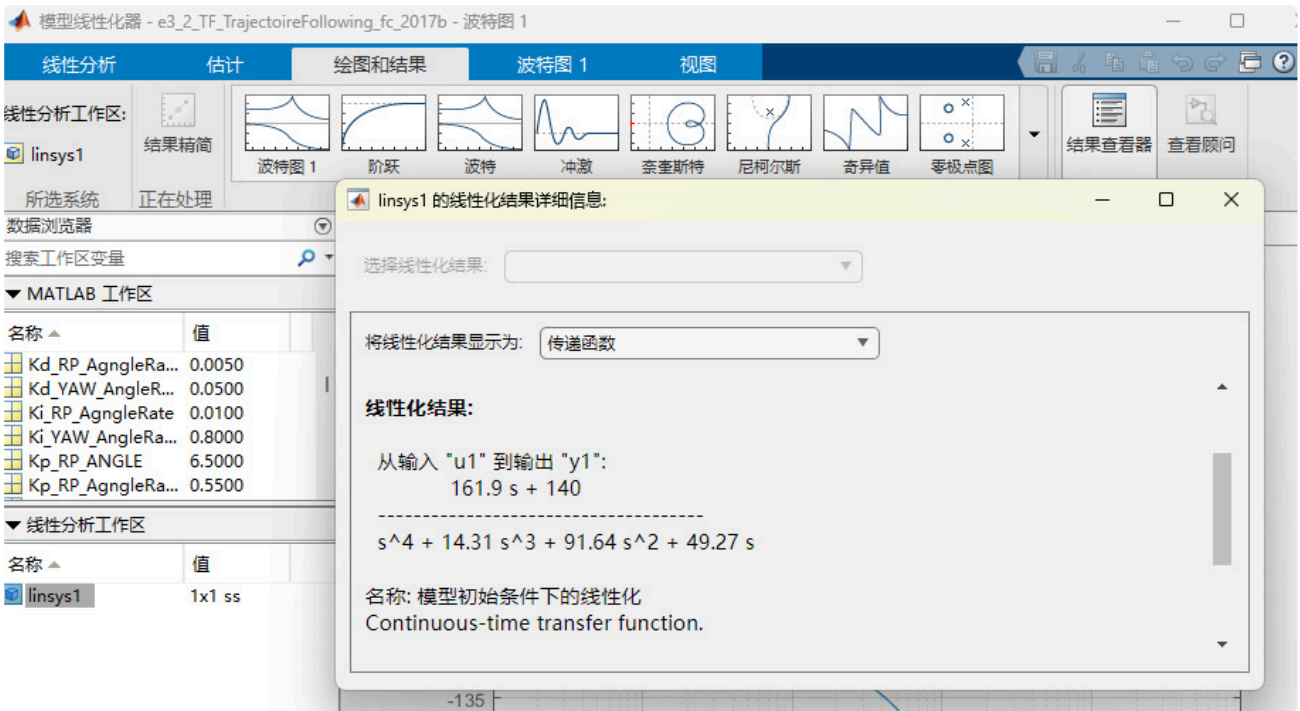


图 6.23 传递函数界面

新版本的MATLAB点击“绘图与结果”, 点击“结果查看器”, 在“将线性化结果显示为”一栏选择“传递函数”:



步骤1.2.3：调整开环增益

首先根据稳态误差调整开环增益。由传递函数可知，未加校正时，系统在阶跃作用下无稳态误差，无须更改开环增益。

步骤1.2.4：增大截止频率并进行相角校正

由前述可知，系统的相角裕度是符合要求的，但截止频率太小，动态响应的快速性不够。增大开环增益K，截止频率增加但相角裕度又不符合要求。考虑使用超前校正，在增加截止频率的同时使系统具有足够的相角裕度。选取截止频率= 3rad/s，由图可知，此处幅值响应为-10.8dB，根据超前校正环节幅频特性得到

$$10\lg a + 20\lg |G(j\omega_c)| = 0$$

进而得到 $a=12.023$ 。为了使最大超前相角落在截止频率=3rad/s处，令 $\omega_m = \omega_c$ ，这时有

$$T = \frac{1}{\sqrt{a}\omega_m}$$

即 $T=0.096s$ 。从而我们得到超前校正环节为

$$G_c(s) = \frac{1 + aTs}{1 + Ts} = \frac{1.156s + 1}{0.096s + 1}$$

步骤1.2.5：应用校正器并验证效果

将设计好的校正器加入模型中，如图所示。可以得到加入校正后的Bode图，如图所示。从图中可以看到，截止频率为3.06rad/s，相角裕度为111°，符合要求。

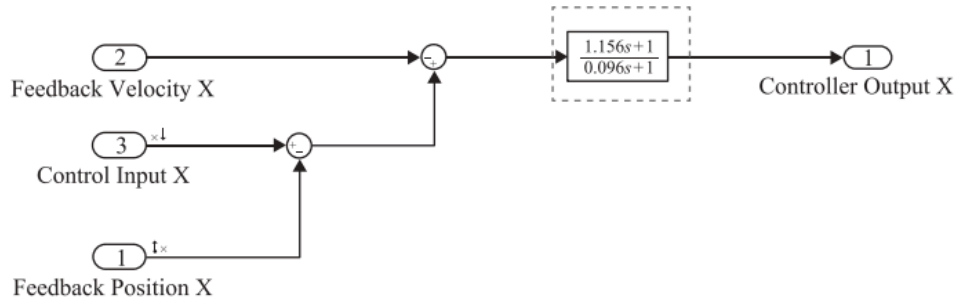


图 6.24 加入校正器

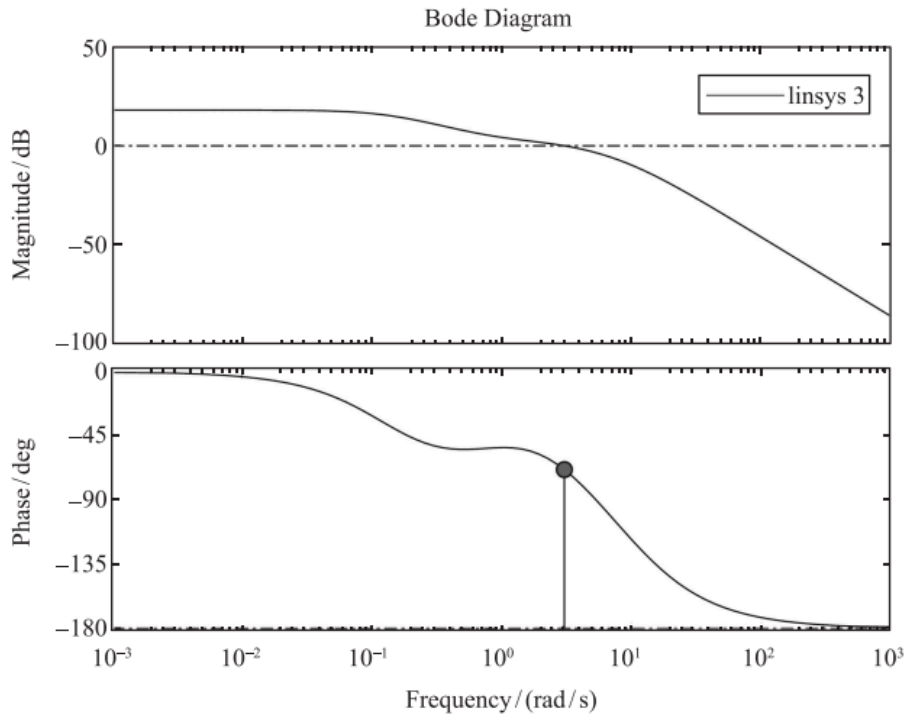


图 6.25 加入校正后的速度控制环 Bode 图

步骤1.3：水平前向通道位置环校正

与速度控制环相似，首先得到未校正的位置控制环开环Bode图，设置输入/输出点如图所示，得到Bode图如图所示。从图中可以看到，截止频率为0.73rad/s，相角裕度为72.2°。根据位置控制环相位裕度 $>60^\circ$ ，截止频率 $>2\text{rad/s}$ 的要求，相角裕度无须调整，只需提高截止频率。为此考虑提高开环增益。从图可以看出， $\omega = 2\text{rad/s}$ 时，幅频特性曲线的值为-10dB，要求加入校正环节后幅频特性曲线值为0dB，则有

$$20\lg K = 10$$

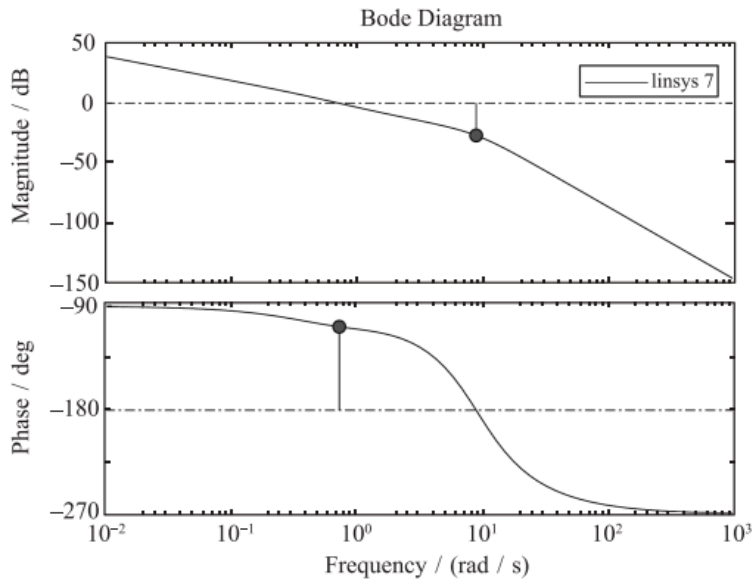


图 6.27 未校正位置控制环开环 Bode 图

可得 $K = 3.2$ 。加入校正环节后的Bode图如图所示，截止频率为 2.01rad/s ，相角裕度为 63.9° ，符合要求。校正前后的阶跃响应对比如图所示，从图中可以看到性能有明显改善。针对其他通道校正器设计，读者可留做练习。

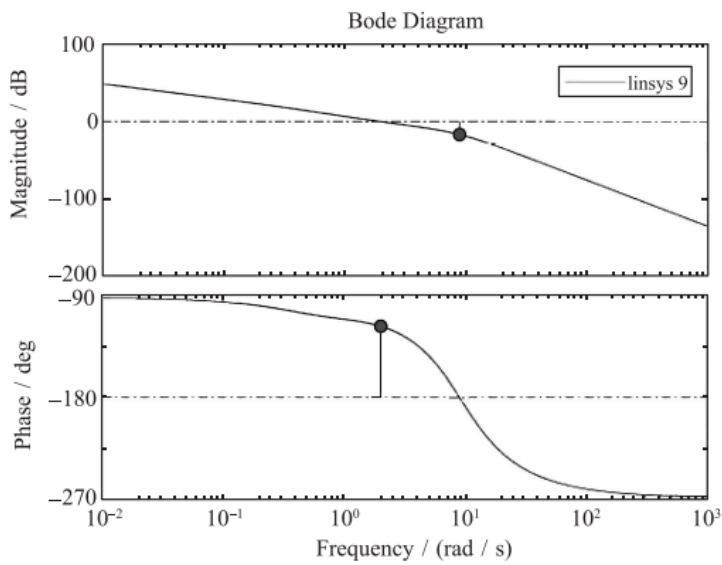


图 6.28 校正后位置控制环开环 Bode 图

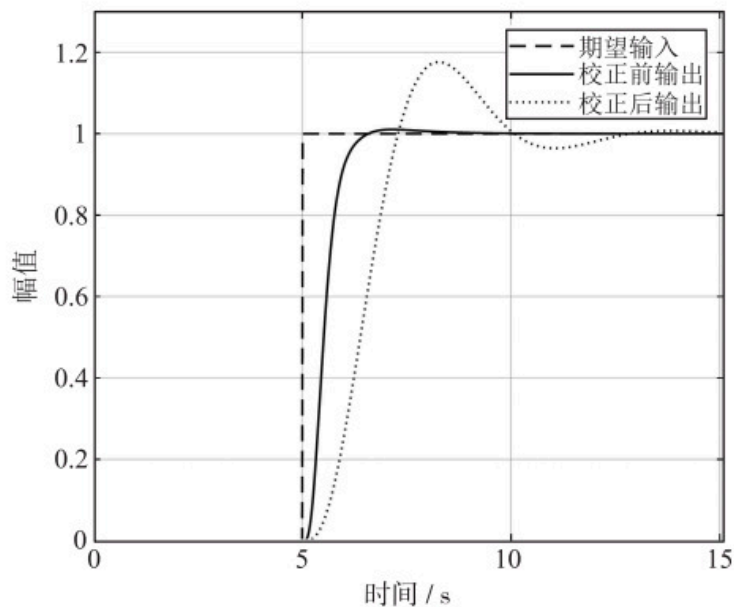


图 6.29 校正前后阶跃响应对比图

步骤1.4：对比校正前后的轨迹跟踪效果

对于实验目标中给定周期 $T = 5\text{s}$ 的圆轨迹信号，加入控制器后的响应结果如图所示。从图中可以看到，此时的跟踪效果在加入校正环节后有了较大的改善。具体步骤可参考基础实验步骤。

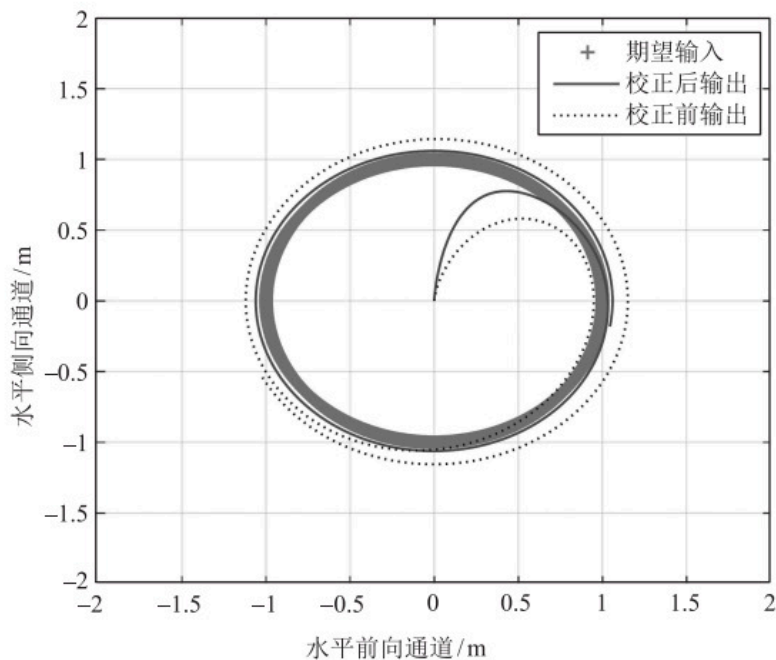


图 6.30 圆轨迹校正前后对比

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 sim2.0 中的 Simulink 文件，其中的闭环系统结构与 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。

步骤2.1：参数初始化

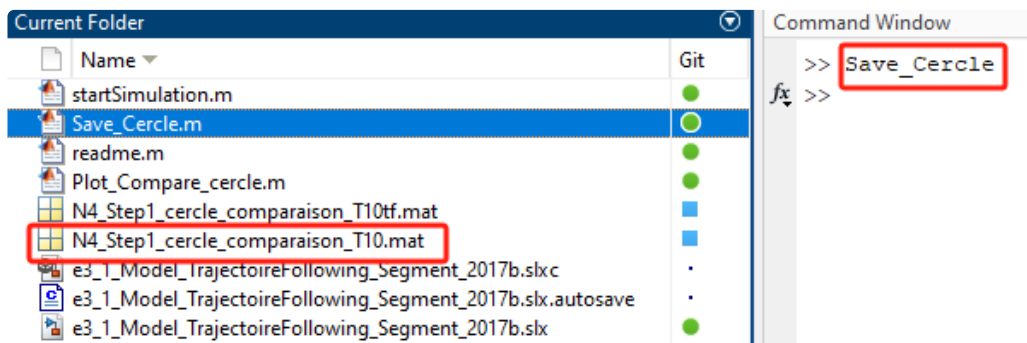
打开本章所附带文件夹中的 `sim2.0` 文件夹，打开 `sim2.0/startSimulation.m` 文件，并单击"运行"按钮，初始化参数。

步骤2.2：仿照sim1.0设计校正器

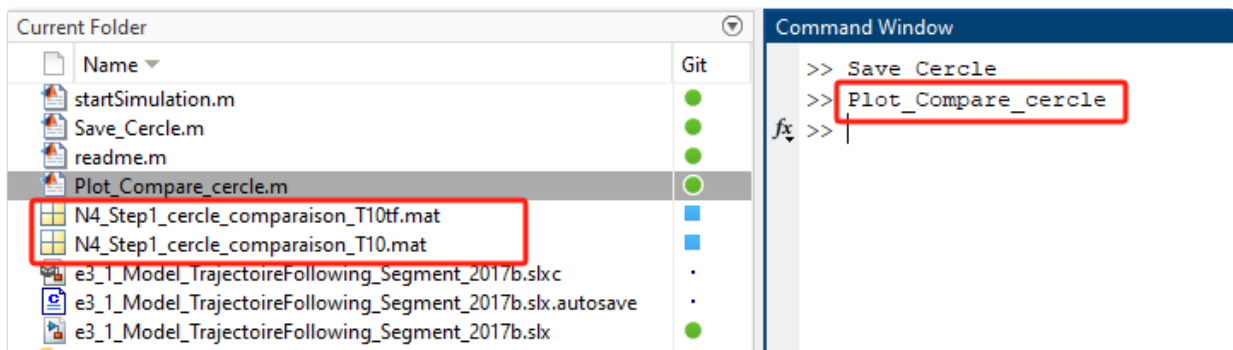
按照仿真1.0中的步骤完成实验。

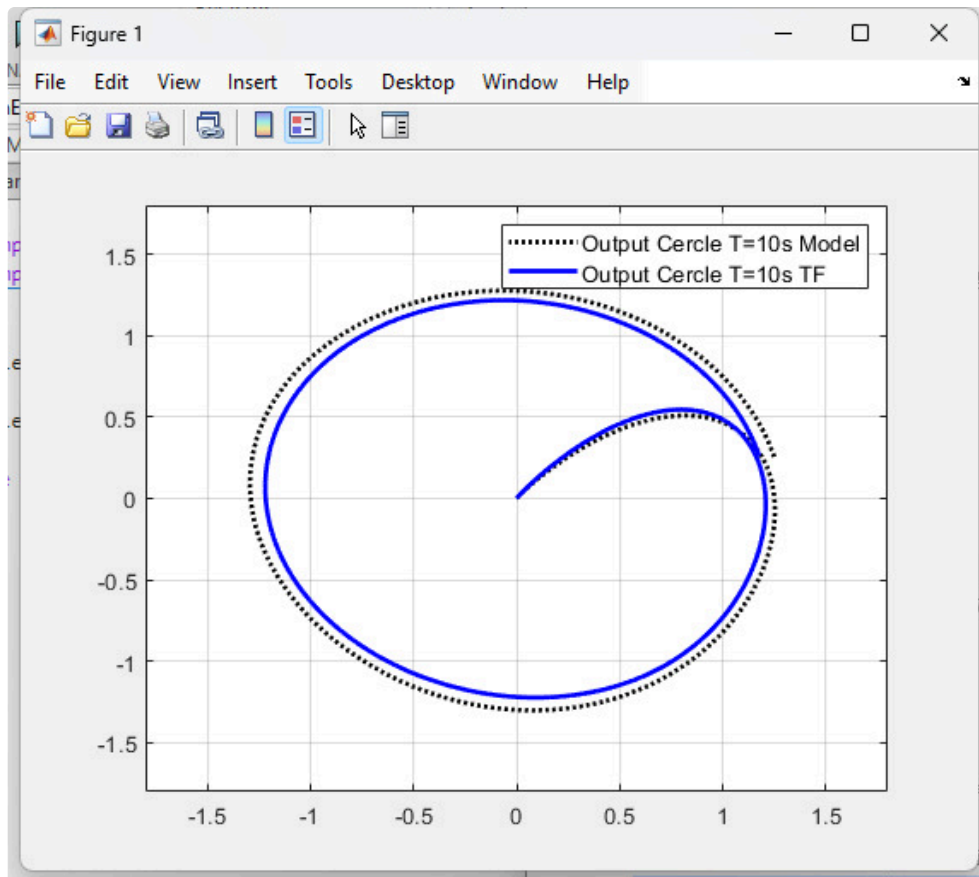
步骤2.3：与仿真1.0对比轨迹跟踪效果

以 $T = 10\text{s}$ 圆轨迹期望输入，运行 `sim2.0/e3_2_Model_TrajectoireFollowing_FC_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

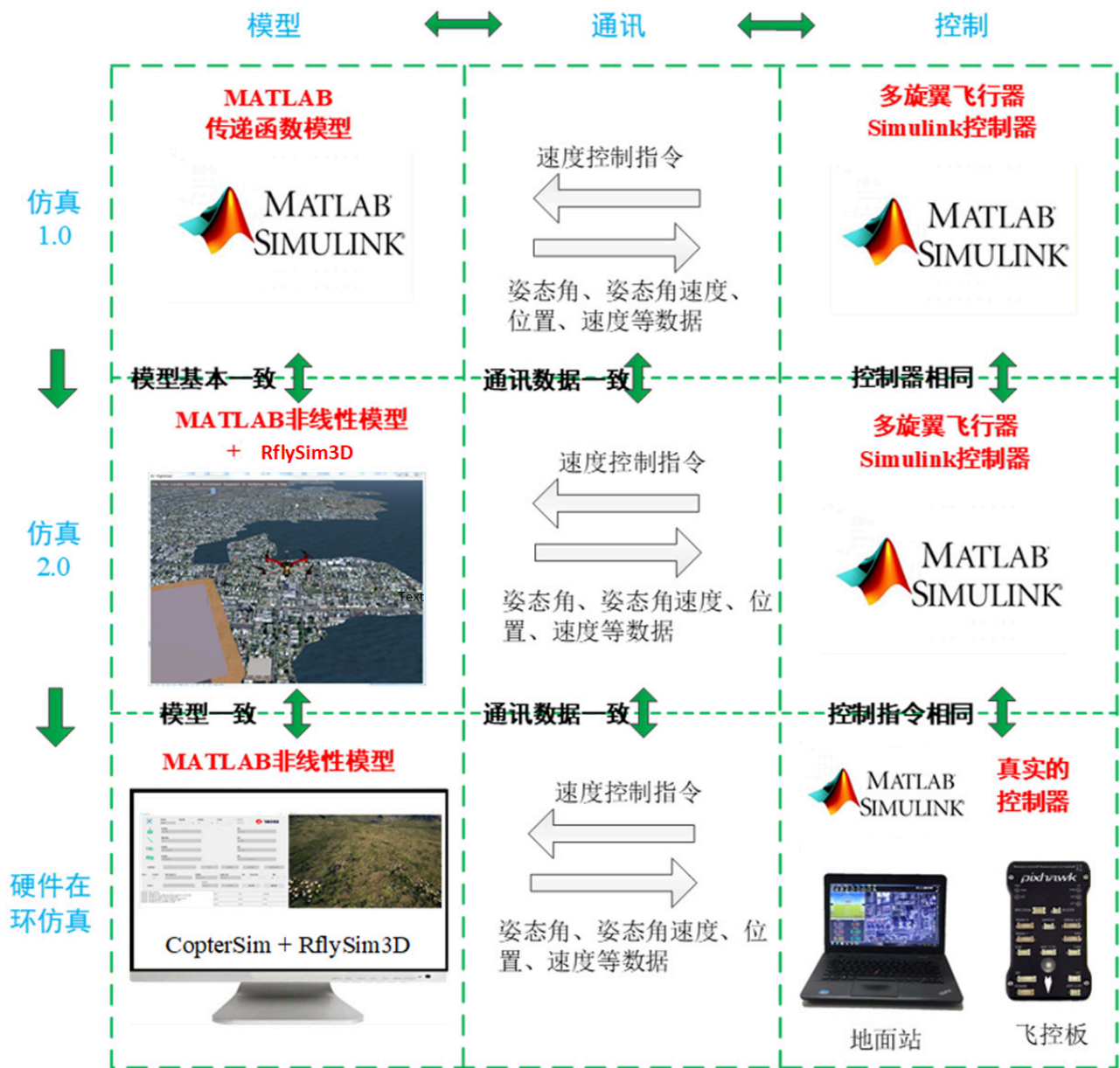
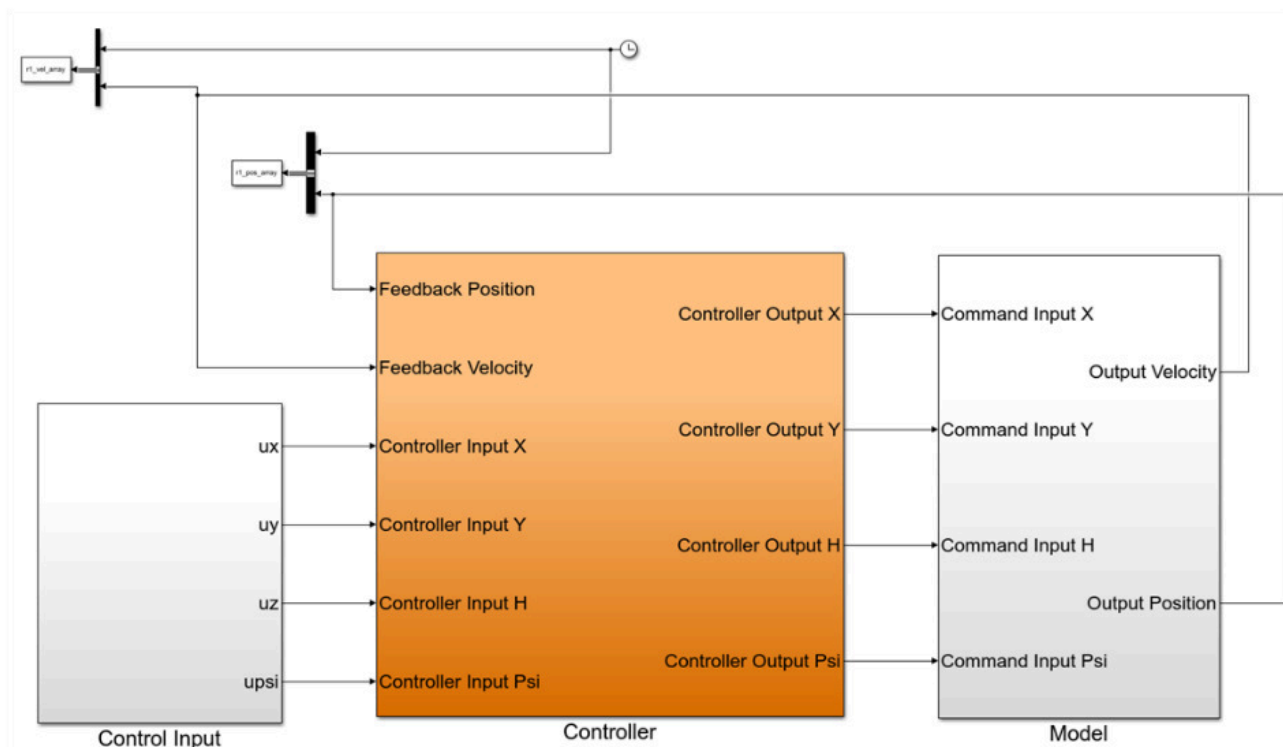


图 仿真阶段流程

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: 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。

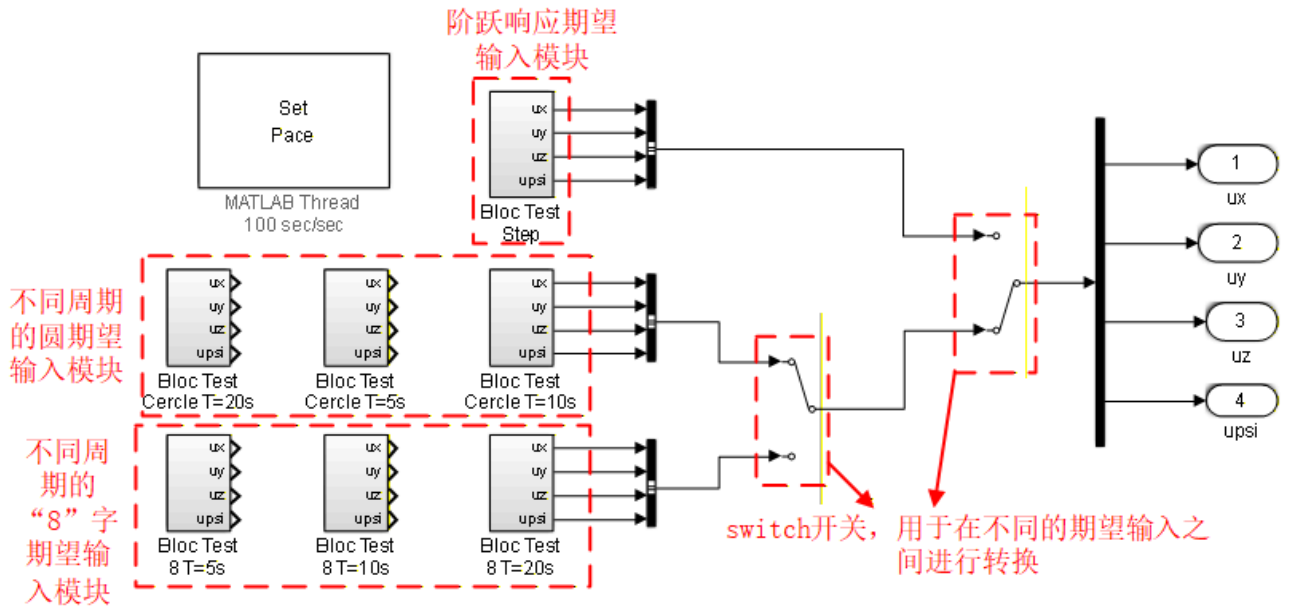


图 1期望输入模块

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

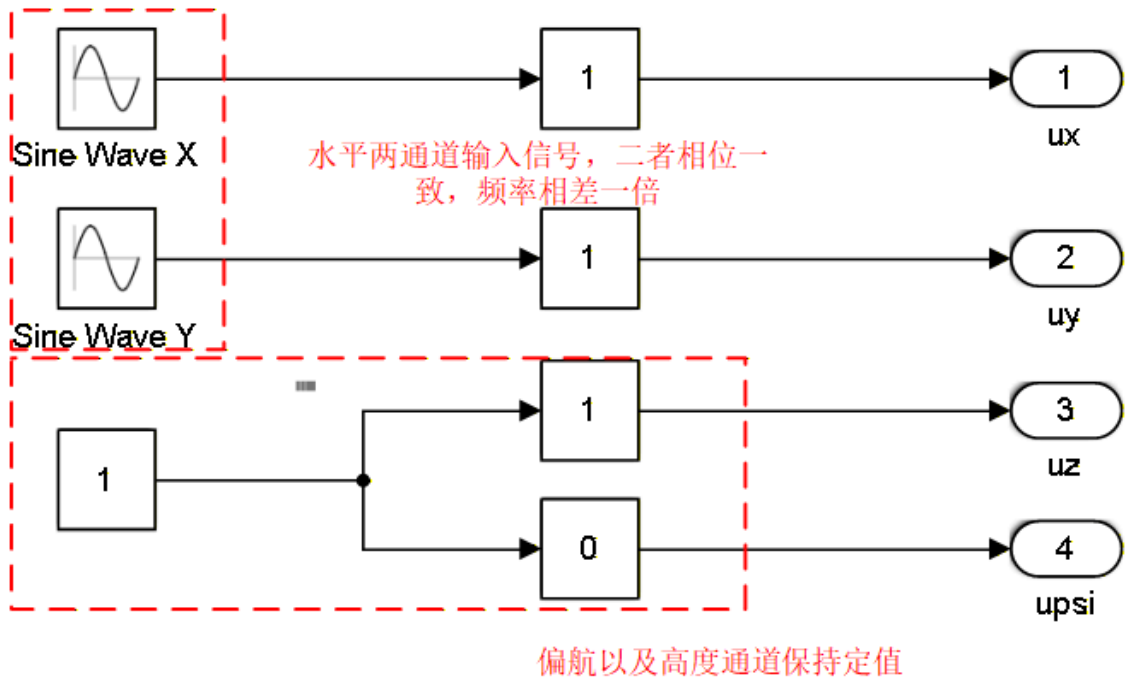


图 2“8”字轨迹期望输入

“Controller”控制器模块

整体结构

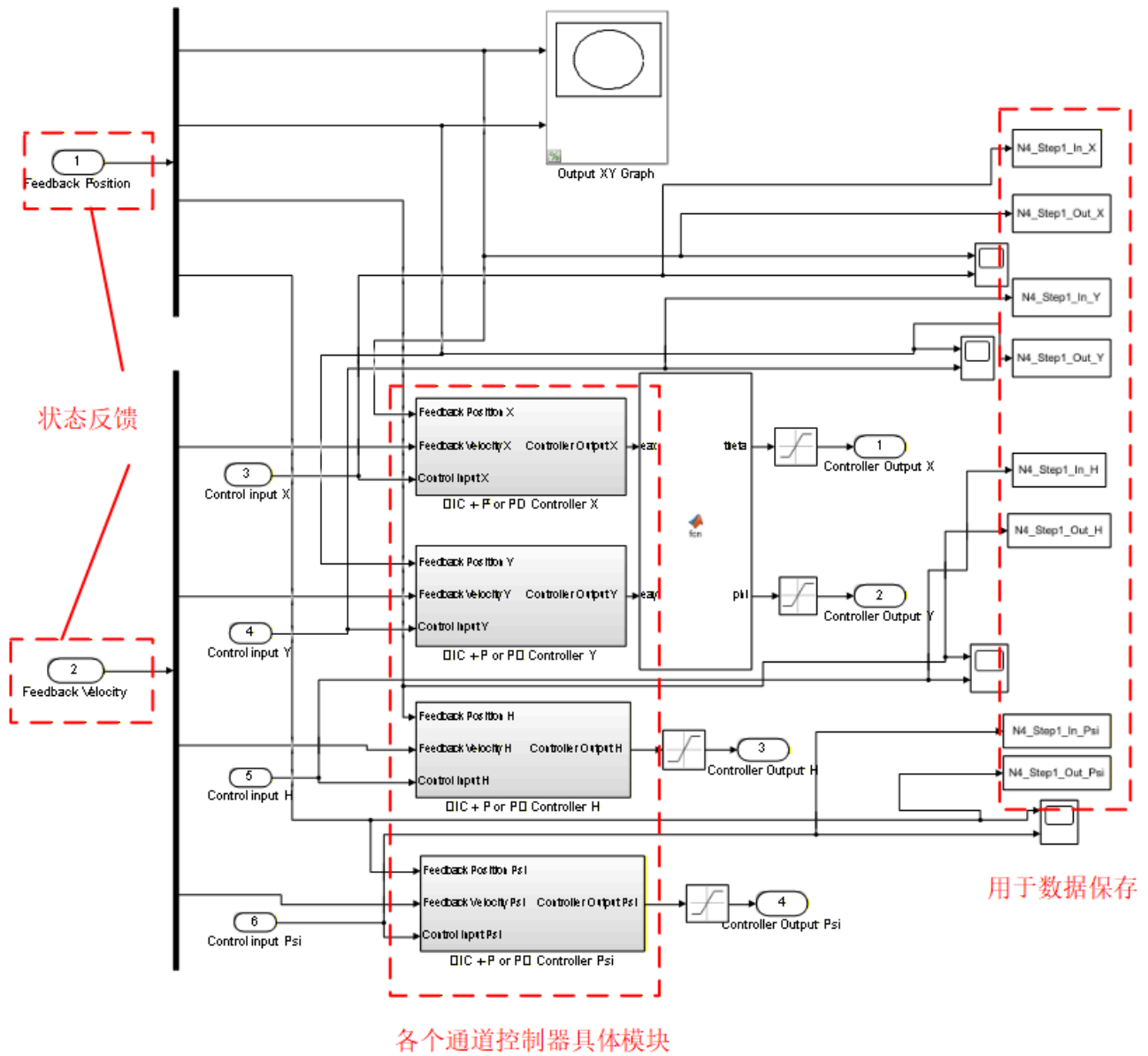


图 3 控制器模块

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

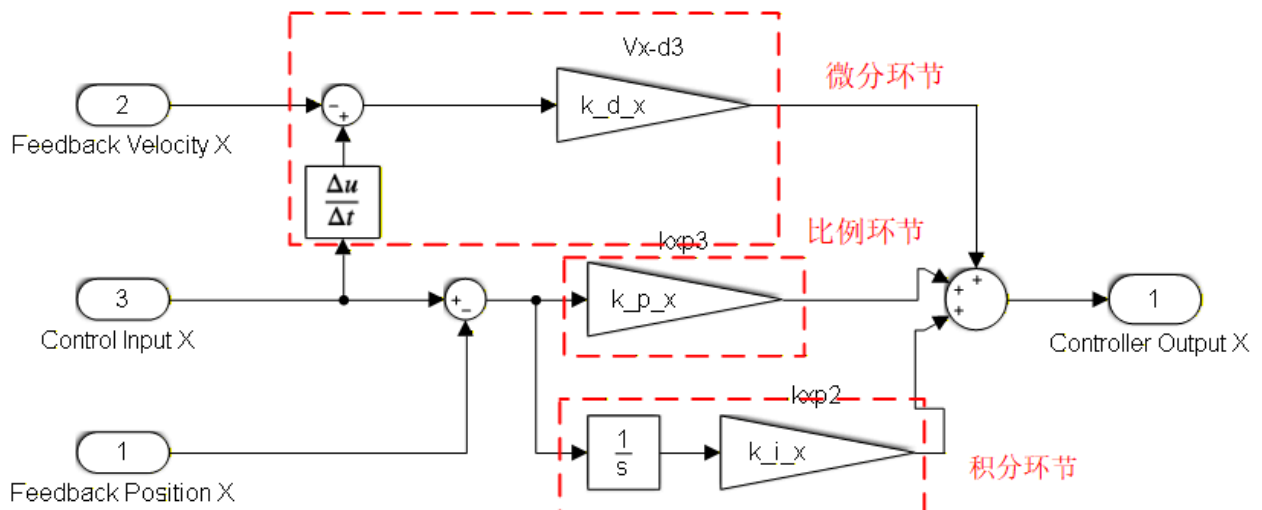


图 4水平前向通道的PID 控制器具体结构

三通道模型

控制器：

$$u_T(t) = -k_{p_z p} (p_{z_e}(t) - p_{z_e d}(t)) - k_{p_z d} (\dot{p}_{z_e}(t) - \dot{p}_{z_e d}(t)) - k_{p_z i} \int_0^t (p_{z_e}(\tau) - p_{z_e d}(\tau)) d\tau$$

期望：

$$p_{z_e d} \in \mathbb{R}$$

控制器：

$$u_{\omega_z}(t) = -k_{\psi p} (\psi(t) - \psi_d(t)) - k_{\psi d} (\omega_z(t) - \dot{\psi}_d(t)) - k_{\psi i} \int_0^t (\psi(\tau) - \psi_d(\tau)) d\tau$$

期望：

$$\psi_d \in \mathbb{R}$$

控制器：

$$\mathbf{u}_h(t) = -\mathbf{K}_{hp} \mathbf{R}_{\psi}^T (\mathbf{p}_h(t) - \mathbf{p}_{hd}(t)) - \mathbf{K}_{hd} \mathbf{R}_{\psi}^T (\dot{\mathbf{p}}_h(t) - \dot{\mathbf{p}}_{hd}(t)) - \mathbf{K}_{hi} \int_0^t \mathbf{R}_{\psi}^T (\mathbf{p}_h(\tau) - \mathbf{p}_{hd}(\tau)) d\tau$$

期望：

$$\mathbf{p}_{hd} \in \mathbb{R}^{2 \times 2}$$

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)

6. 参考资料

1. RflySim官方文档
2. Quan Quan. Introduction to Multicopter Design and Control. Springer, Singapore, 2017
3. 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.
4. 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.
5. 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: 如何初始化参数?

A1: 打开本章所附带文件夹中的 `sim1.0` 或 `sim2.0` 文件夹, 打开对应的 `startSimulation.m` 文件, 并单击"运行"按钮, 初始化参数。

Q2: 如何进行控制器校正?

A2: 首先通过Simulink的"Analysis"->"Control Design"->"Linear Analysis"功能得到Bode图, 然后根据性能要求 (速度控制环相位裕度 $>65^\circ$, 截止频率 $>3\text{rad/s}$; 位置控制环相位裕度 $>60^\circ$, 截止频率 $>2\text{rad/s}$) 设计合适的校正器 (如超前校正、增益调整等), 最后验证校正后的系统性能。

Q3: 如何对比仿真1.0和仿真2.0的效果?

A3: 运行相应仿真文件 (`sim1.0/e3_2_TF_TrajectoireFollowing_fc_2017b.slx` 或 `sim2.0/e3_2_Model_TrajectoireFollowing_FC_2017b.slx`) 后, 执行对应的 `Save_Cercle.m` 保存数据, 然后运行 `Plot_Compare_cercle.m` 加载两个仿真保存的数据进行对比。

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