

1. 实验名称及目的

1.1. 实验名称

轨迹跟踪控制器设计实验

1.2. 实验目的

对上述轨迹跟踪控制器进行改进，加入偏航角的跟踪，使得控制器可以在偏航角偏转的情况下稳定跟踪。更具体地，设计加性分解控制器，控制多旋翼跟踪圆轨迹。已知所跟踪的圆轨迹圆心位于(0,0)，半径为 1m，多旋翼初始位置随机，可定为(0,0)，且在绕圆飞行过程中，机头始终指向圆心。所设计的控制器有如下性能要求：

- (1) 要求设计一个完整的控制器，使得控制器可以稳定控制多旋翼模型。
- (2) 要求所设计的控制器，在偏航通道稳定的情况下，水平方向仍能稳定飞行同时与期望的幅值偏差不大于 20%。
- (3) 将设计好的加性分解控制器应用到非线性模型，即仿真 2.0，并进行硬件在环仿真。

1.3. 关键知识点

关键知识点 1：实验整体流程

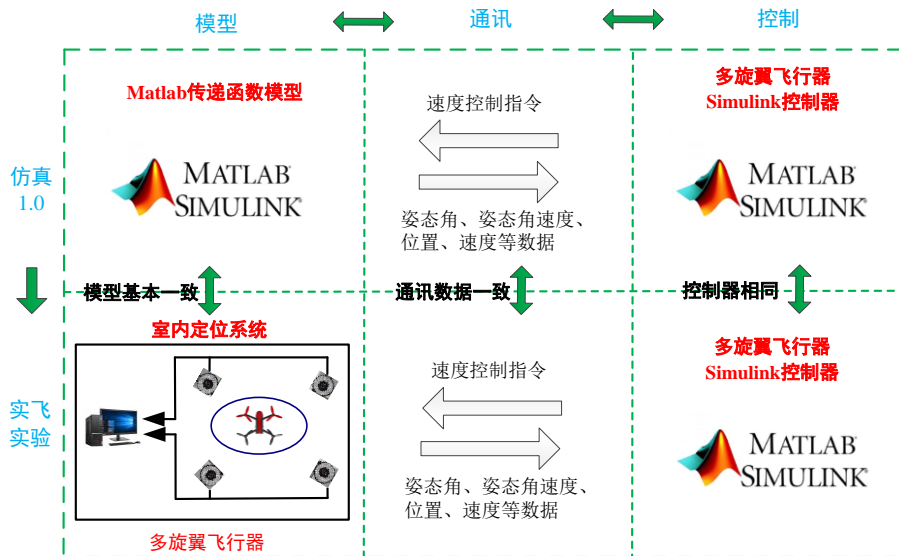


图 实飞阶段流程

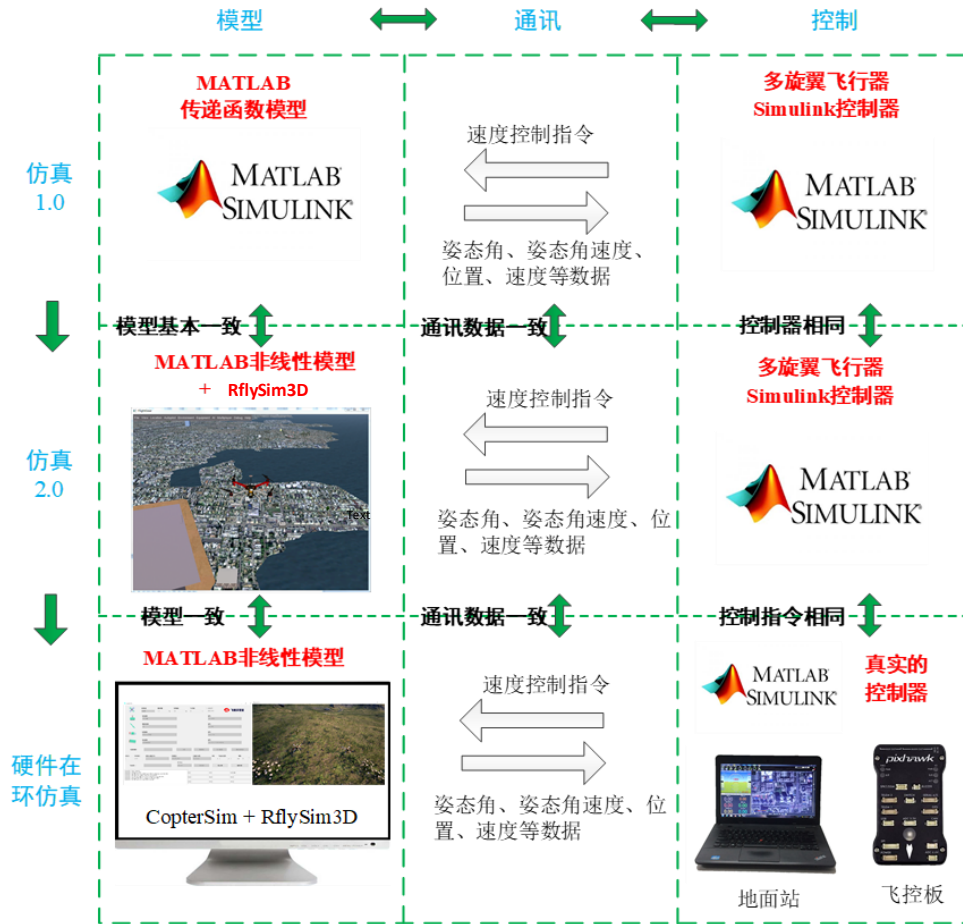
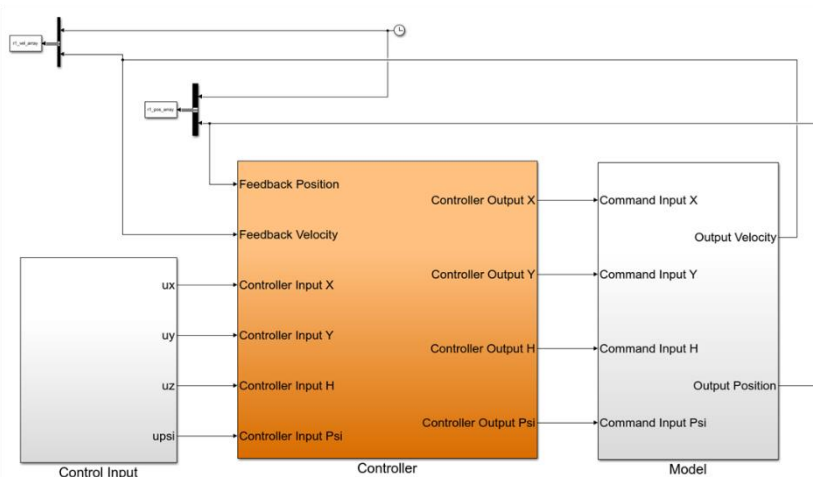


图 仿真阶段流程

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

期望输入模块

本实验不仅需要轨迹跟踪，而且需要机头时刻指向圆心。对于水平方向，可以将前向通道的期望输入定为 $\sin(\omega t)$ ，则水平侧向通道的期望输入为 $\cos(\omega t)$ ，二者的角速率 ω 是相同的，这样就可以实现水平平面内的圆周运动。高度保持不变，机头始终指向圆心，则需要偏航通道的期望输入与水平方向的角速率变化同步。在初始 0 时刻，多旋翼在水平平面需要飞到(0,1)，以水平前向通道正方向为零偏航角，此时机头若指向圆心则需要偏航角保持为 $-\sin(\omega t)$ 。

控制器模块

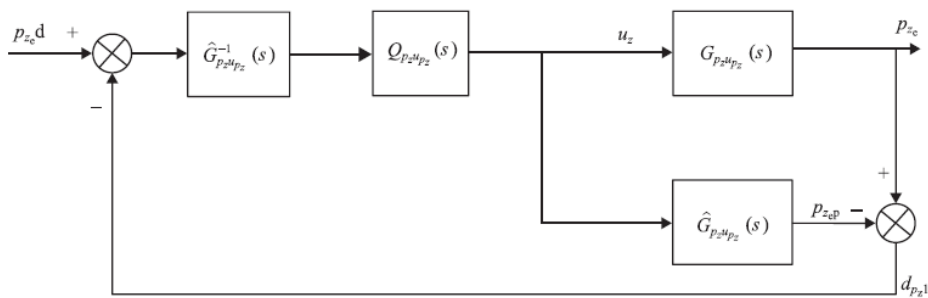


图 6.32 基于加性输出分解方法的控制器设计

三通道模型

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

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

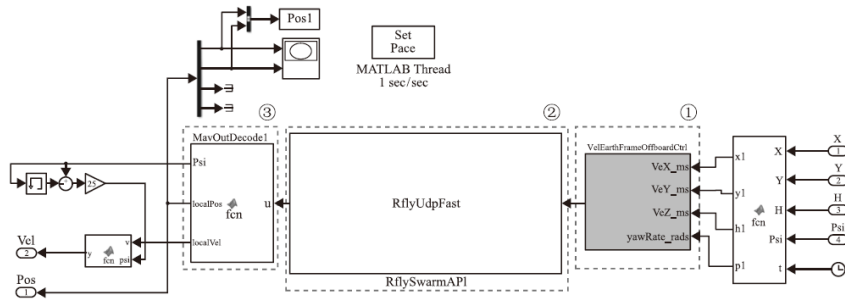
多旋翼模型模块

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)

HIL (硬件在环通信模块——相当于使用 CopterSim 内的 dll 模型)

硬件在环模块内部如图所示，该模块是 Simulink 模型和 RflySim 平台 (CopterSim) 之间的通信桥梁，接收 Simulink 模型中发出的控制指令并返回硬件在环仿真过程中多旋翼的位置、速度和欧拉角等数据。图中虚线框①用于将需要发送的数据打包成统一的数据结构作为虚线框②的输入。虚线框②是 simulink 和 CopterSim 通信的核心模块，包含两个作用，一是将硬件在环仿真需要的数据发送给 CopterSim，一是从 CopterSim 中接收反馈信息。硬件在环仿真的 Simulink 文件保存在每一章的设计实验文件夹中，与软件仿真使用的文件区别在于用硬件在环模块替换了多旋翼模型模块。这是因为硬件在环仿真的多旋翼模型在 CopterSim 中。



2. 实验效果

输出各个方向上在一段时间内的输入/输出对比图，以及在不同周期下圆轨迹曲线的对比图。

3. 文件目录

例程目录: [\[安装目录\]\RflySimAPIs\6.RflySimExtCtrl\1.BasicExps\e7_MutUAVRemoteCtrl\3.TrajectoryFollowing_Segment\e3.3\](#)

文件夹/文件名称	说明	
Sim1.0	Plot_Compare_cercle.m	比较圆不同周期文件

	Compare_Cercle.m	保存 PID 调整前后的圆轨迹比较数据
	N4_Step3_cercle_comparaison_T5.mat	圆轨迹输入输出融合数据文件
	N4_Step3_cercle_comparaison_T10.mat	圆轨迹输入输出融合数据文件
	N4_Step3_cercle_comparaison_T10tf.mat	圆轨迹输入输出融合数据文件
	N4_Step3_cercle_comparaison_T20.mat	圆轨迹输入输出融合数据文件
	e3_3_TF_TrajectoireFollowing_DIC_2017 b.slx	多旋翼轨迹控制器仿真 1.0 文件
	startSimulation.m	初始化参数文件
sim2.0	N4_Step3_cercle_comparaison_T5.mat	圆轨迹输入输出融合数据文件
	N4_Step3_cercle_comparaison_T10.mat	圆轨迹输入输出融合数据文件
	N4_Step3_cercle_comparaison_T10tf.mat	圆轨迹输入输出融合数据文件
	N4_Step3_cercle_comparaison_T20.mat	圆轨迹输入输出融合数据文件
	Plot_Compare_cercle.m	比较圆不同周期文件
	Compare_Cercle.m	保存 PID 调整前后的圆轨迹比较数据
	e3_3_Model_TrajectoireFollowing_DIC_20 17b.slx	多旋翼轨迹控制器仿真 2.0 文件
startSimulation.m	初始化参数文件	
HIL	e3_Model_TrajectoireFollowing_HTIL_201 7b.slx	多旋翼轨迹控制器硬件在环仿真文件
	DrawFigure.asv	图形绘制数据文件
	DrawFigure.m	图形绘制执行文件
	RflyUdpFast.mexw64	动态链接库
	StartSimulation.m	初始化参数文件
	T20s_result.fig	结果输出文件

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 6X 或其它 飞控 ^②	1
3	MATLAB 2022b b 及以上	遥控器 ^③	1
		遥控器接收器	1
		数据线、杜邦线等	若干

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

②：若使用 Pixhawk6X 飞控，须保证平台安装时的编译命令为：px4_fmU-v6x_default，固件版本推荐：1.13.3。其他配套飞控请见：<https://rflysim.com/doc/zh/B/2.3Pixhawk6X.html>

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: <https://rflysim.com/doc/zh/B/3.1ET10.html>

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“e3\3.3\sim1.0”文件夹, 打开“e3\3.3\sim1.0\startSimulation.m”文件, 并单击“运行”按钮, 初始化参数。

打开 Simulink 文件“e3\3.3\sim1.0\3_3_TF_TrajectoryFollowing_DIC_2017b.slx”, 整体模块如图 6.33 所示。

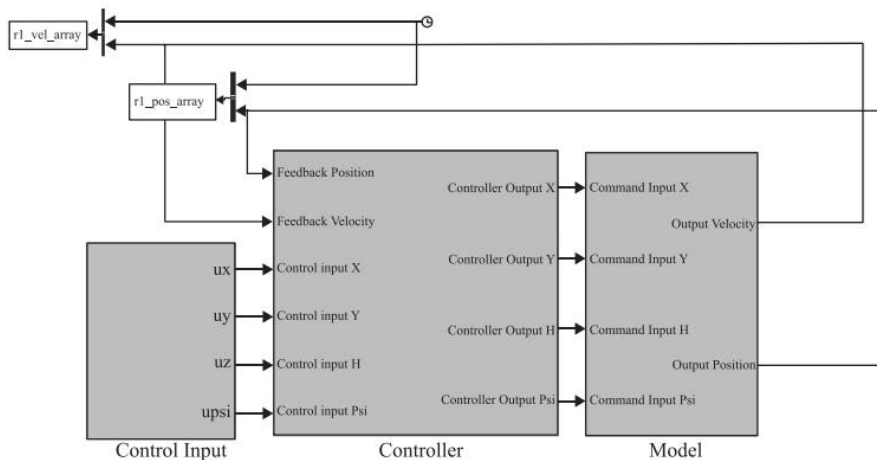


图 6.33 模块整体示意图, Simulink 模型详见“e3_3_TF_TrajectoryFollowing_Segment.slx”

Step 2: 设置期望输入

根据期望输入模块中设计的圆跟踪轨迹, 将控制输入模块做相应修改, 具体如图 6.34 所示, 以周期 $T = 10s$ 的期望输入为例, 其内部如图 6.35 所示。

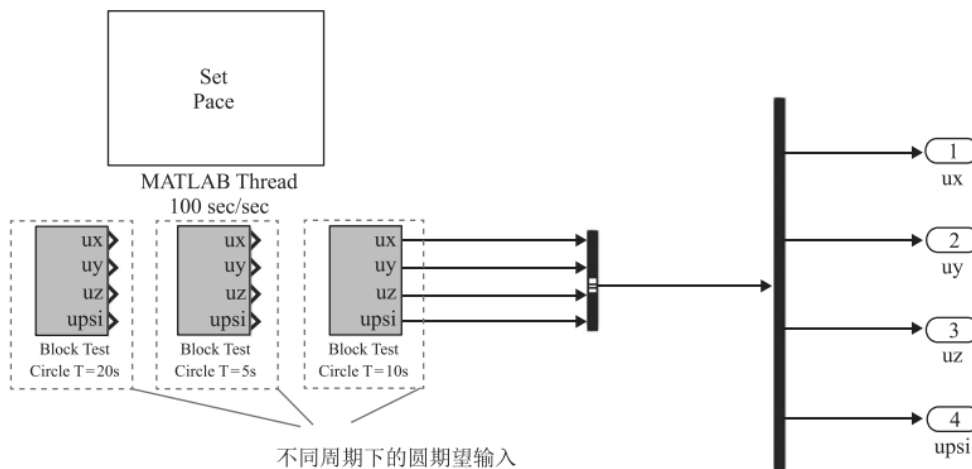


图 6.34 期望输入模块

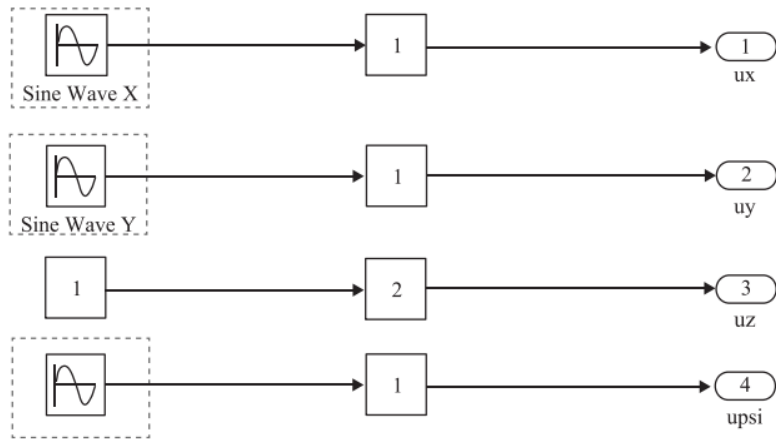


图 6.35 $T = 10s$ 圆轨迹期望输入

Step 3: 控制器设计

高度通道和偏航通道是独立的，并不会与其他通道产生耦合。因此，不需要再设计新的基于加性分解的动态逆控制器。只需针对水平方向的两个通道进行控制器设计。加入加性分解控制器的控制器模块整体如图 6.36 所示。

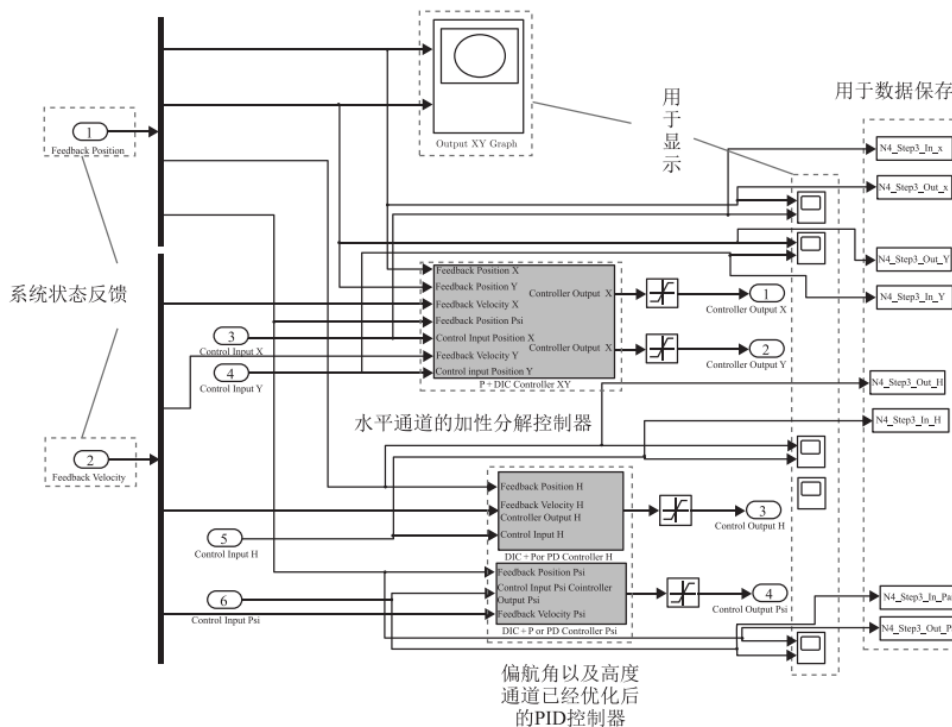


图 6.36 加入加性分解控制器的控制器模块示意图

除水平通道的控制器外，其他模块与分析实验均相同，不再赘述。对于水平通道的控制器设计，根据 6.4.2.2 节中的原理，首先需要将原来的位置期望转化为速度期望，之后再根据速度期望设计相应的加性分解控制器。所设计加性分解控制器模块示意图如图 6.37 所示。

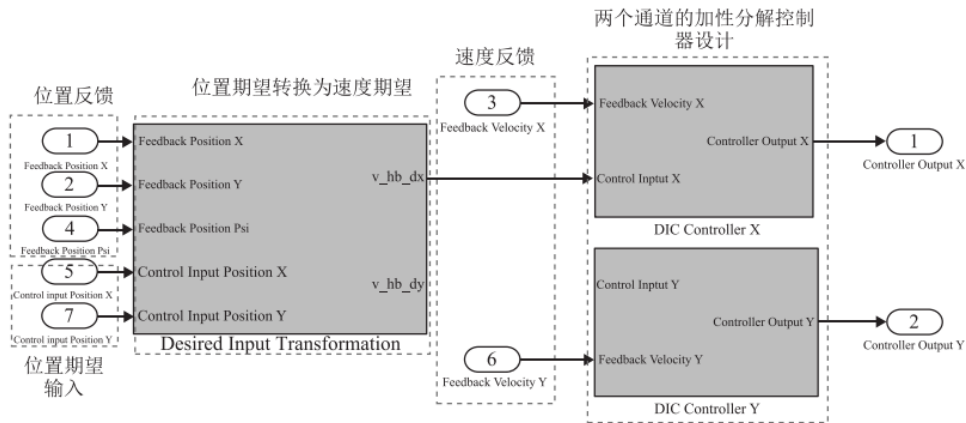


图 6.37 加性分解控制器模块示意图

对于将位置期望转换为速度期望的模块，其内部如图 6.38 所示。

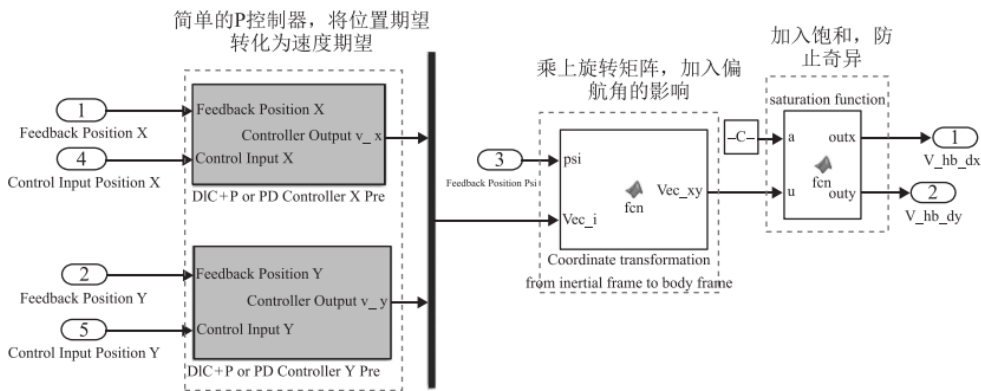


图 6.38 位置期望转换为速度期望

以水平前向通道为例，引入位置反馈，并加入一个比例控制器，其内部如图 6.39 所示。

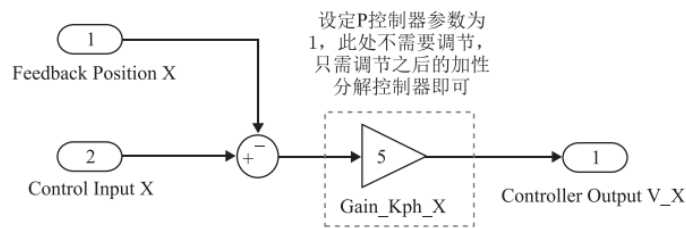


图 6.39 水平前向通道比例控制器示意图

之后引入偏航角的影响，这里使用的是多旋翼实时的偏航角反馈。其内部代码如表 6.1 所示。

表 6.1 偏航反馈代码

```
function Vec_xy = fcn( psi, Vec_i )
T = [cos(psi) -sin(psi) ; sin(psi) cos(psi)] ;
Kph=eye(2);
Vec_xy = inv(T)*Vec_i(1:2,1);
```

此外，为防止瞬时值过大，加入饱和上限。得到速度期望后，加性分解控制器的设计如图 6.40 所示。

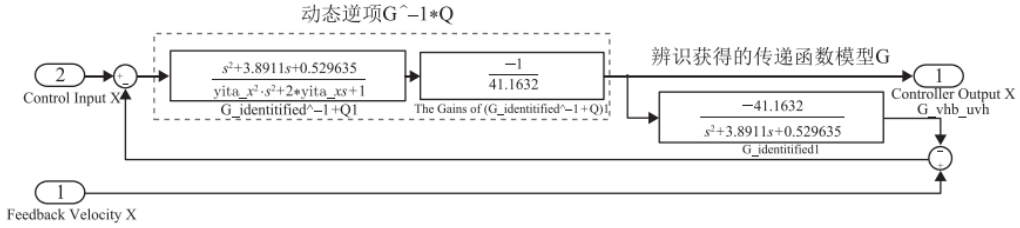


图 6.40 加性分解控制器的设计

Step 4: 滤波器设计

对于加性分解控制器所需要的滤波器，考虑二阶低通滤波器即可，即

$$Q_{v_{hb}u_{vh}}(s) = \begin{bmatrix} \frac{1}{(\eta_x s + 1)^2} & 0 \\ 0 & \frac{1}{(\eta_y s + 1)^2} \end{bmatrix}$$

其中，两个参数 η_x 和 η_y 可在“e3\3.3\sim1.0\startSimulation.m”中修改，具体代码如表 6.2 所示。运行模型文件之前，首先设定合适的滤波器系数，之后再运行模型文件观察控制效果，找到一个合适的滤波器系数。

表 6.2 滤波器系数修改

```
1 yita_x=0;
2 yita_y=0;
```

Step 5: 观测控制效果

运行“e3\3.3\sim1.0\startSimulation.m”进行参数初始化。在设计的控制器下，水平方向两通道以及偏航通道在不同周期下的响应结果如图 6.41 至图 6.49 所示，将每次的实验数据保存，可以看到，设计的控制器具有较好的控制效果。

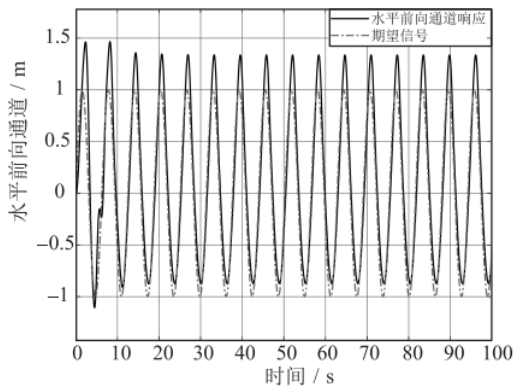


图 6.41 水平前向通道 $T = 5s$ 输入/输出对比图

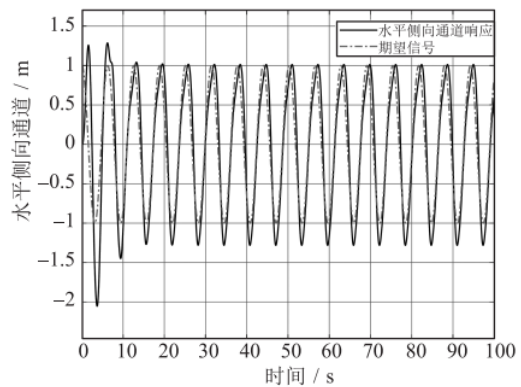


图 6.42 水平侧向通道 $T = 5s$ 输入/输出对比图

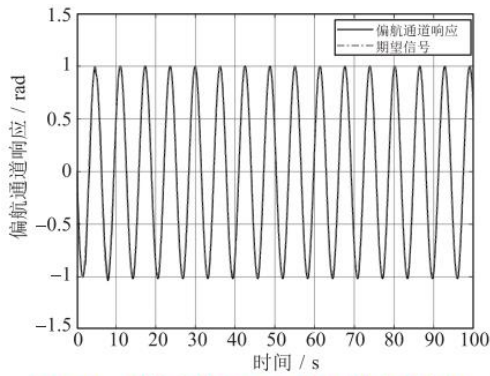


图 6.43 偏航通道 $T = 5s$ 输入/输出对比图

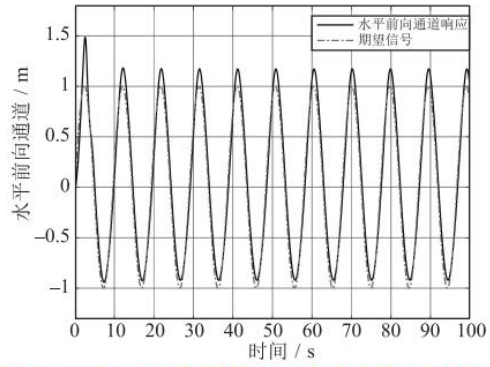


图 6.44 水平前向通道 $T = 10s$ 输入/输出对比图

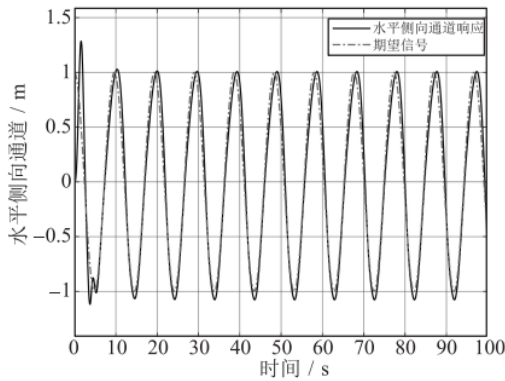


图 6.45 水平侧向通道 $T = 10s$ 输入/输出对比图

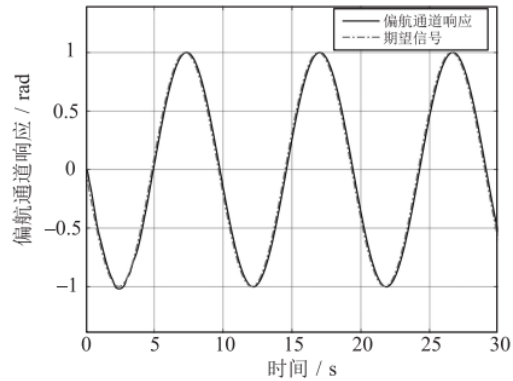


图 6.46 偏航通道 $T = 10s$ 输入/输出对比图

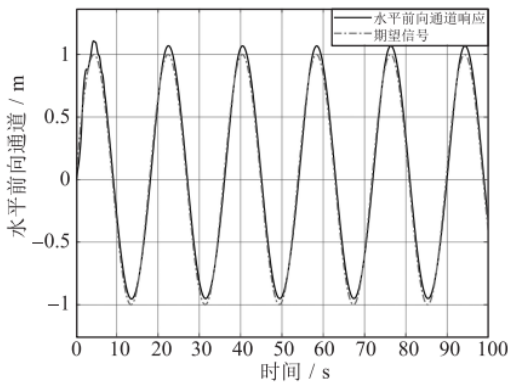


图 6.47 水平前向通道 $T = 20s$ 输入/输出对比图

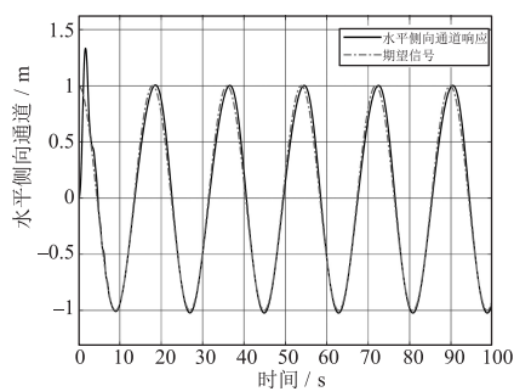


图 6.48 水平侧向通道 $T = 20s$ 输入/输出对比图

利用保存的输入/输出数据，对三种不同周期下的圆轨迹进行综合对比，轨迹响应如图 6.50 所示。从该图可以更直观地看出，当周期 $T = 20s$ 时，控制效果接近理想。随着周期的缩短，控制效果是逐渐变差的，这符合理论：一般而言，根据伯德积分原理，反馈系统不可能兼顾所有频段，只能满足带宽内的频率信号跟踪。在这里，频率越高，控制效果越差。

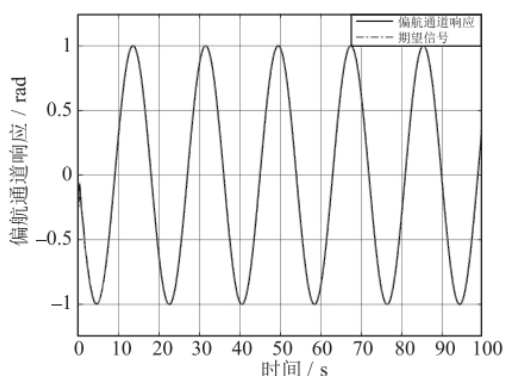


图 6.49 偏航通道 $T = 20s$ 输入/输出对比图

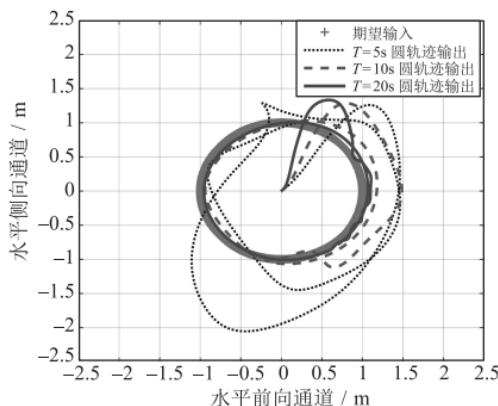


图 6.50 不同周期圆轨迹曲线对比图

(具体操作步骤可以观看每章相对应的 PPT，其中有各个实验相对应的演示视频。)

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。

打开 Simulink 文件“e3\3.3\sim2.0”，其中的闭环系统结构与“e3\3.3\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。

Step 1: 参数初始化

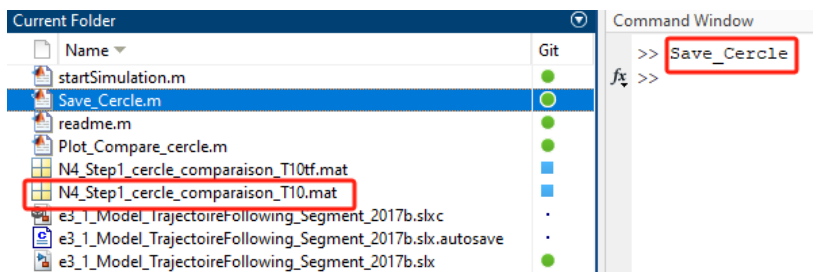
打开本章所附带文件夹中的“e3\3.3\sim2.0”文件夹，打开“e3\3.3\sim2.0\startSimulation.m”文件，并单击“运行”按钮，初始化参数。

Step 2: 仿照 sim1.0 设计校正器

按照[仿真 1.0 中的步骤 step2](#) 完成实验。

Step 3: 与仿真 1.0 对比轨迹跟踪效果

以 $T = 10s$ 圆轨迹期望输入，运行 [sim2.0\3_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

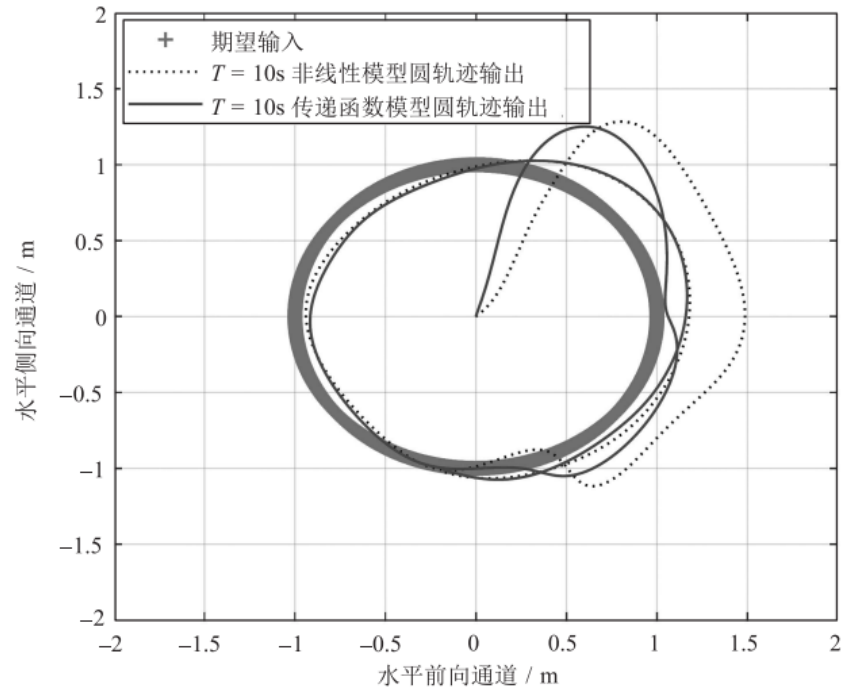
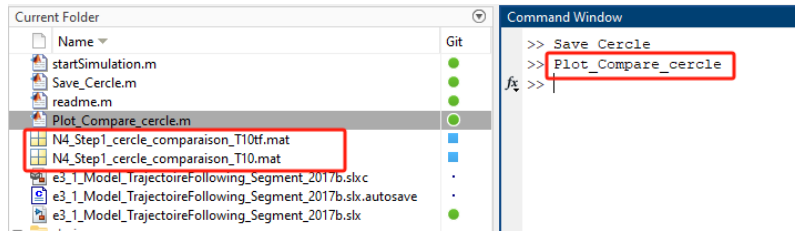


图 6.51 $T = 10s$ 不同模型圆轨迹跟踪对比图

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。

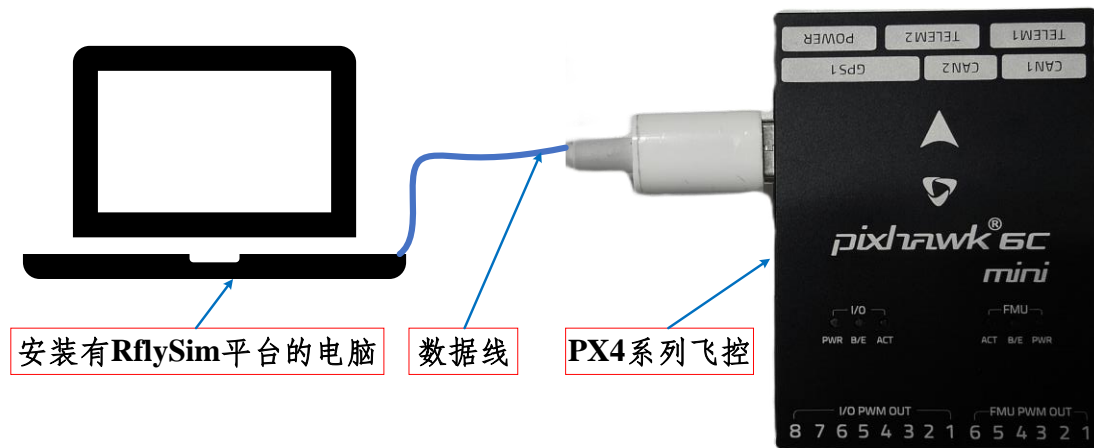
(具体操作步骤可以观看每章相对应的 PPT，其中由各个实验相对应的演示视频。)

5.3. 硬件在环仿真(HIL)实验步骤（选做）

硬件在环仿真实验的内容与设计实验相同,只是把设计实验中的多旋翼模型模块替换成了硬件在环模块，硬件在环仿真所需设备如图所示。

Step 1: 连接硬件

将 Pixhawk 自驾仪与计算机通过 USB 数据线连接。



Step 2: 启动硬件在环仿真

打开桌面“*\桌面\RflyTools\HITLRun.lnk”一键启动硬件在环仿真脚本，在弹出的命令行中。输入 PX4 系列自驾仪显示的串口号，如这里是“3”，只需把该串口号输入下方即可。**注：本脚本也可支持多机仿真，多机硬件在环仿真时，需要插入多个飞控到电脑中，双击“*\桌面\RflyTools\HITLRun.lnk”后弹出的命令行中提示的串口号个数应与连接的自驾仪个数一致，需要在下方输入所有给出的串口号，每个串口号用逗号分隔开。**

```
HITLRun.bat - 快速方式

-----
Please input the Pixhawk COM port list for HIL
Use ',' as the separator if more than one Pixhawk
E.g., input 3 for COM3 of Pixhawk on the computer
Input 3,6,7 for COM3, COM6 and COM7 of Pixhawks
Available COM list on this computer is: 3
-----

My COM list for HITL simulation is: 3
```

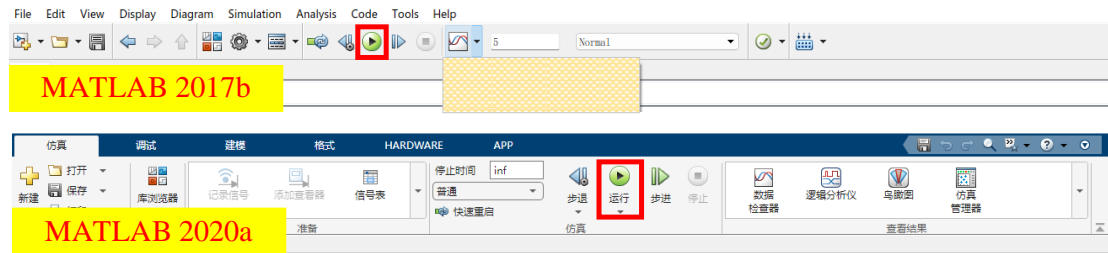
在填写完串口号后，按下回车键，系统会自动打开所有与硬件在环仿真相关的 RflySim 3D、CopterSim 和 QGC 地面站。等待 CopterSim 的左下角状态框中显示：PX4: GPS 3D fixed & EKF initialization finished。

Step 3: 参数初始化

打开本章所附带文件夹中的“e3\3.3\HIL”文件夹，打开“e3\3.3\HIL\startSimulation.m”文件，并单击“运行”按钮，初始化参数。

Step 4: 运行控制程序

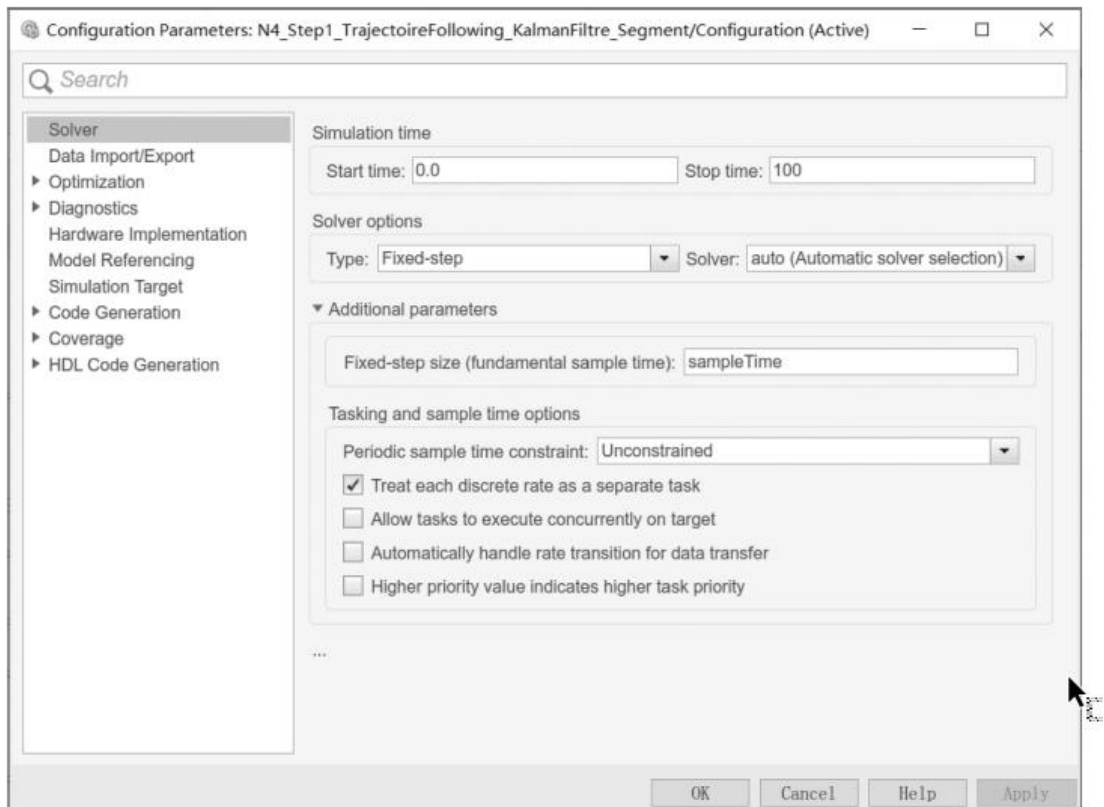
在 Simulink 中打开 [HIL\2_3_TF_KalmanFiltre_Delay_HITL_2017b.slx](#) 程序，单机“运行”。便可以在 RflySim3D 软件中看到硬件在环仿真的效果。



可以实现自驾仪硬件在环仿真，仿真结果通过 RflySim3D 软件实时显示。

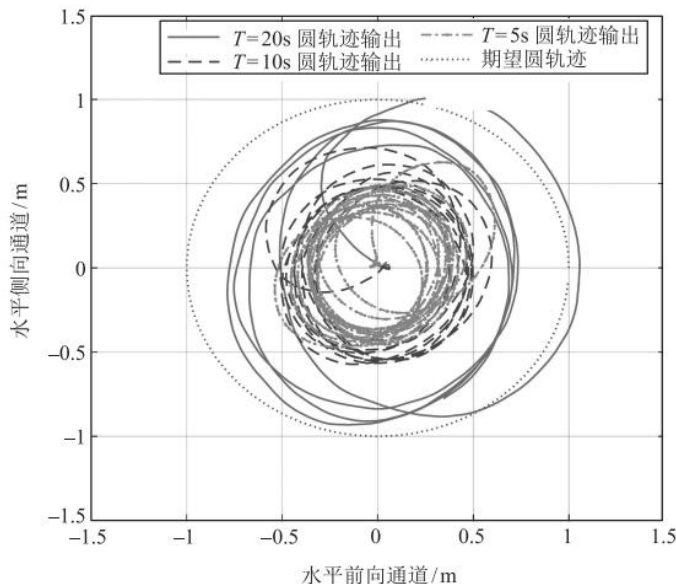


注意：1) 本实验中的仿真模型均在 R2017b 版本中运行，所以需要计算机中的版本为 R2017b 及以上。2) 所有的仿真模型均需要设定固定的仿真步长。在 Simulink 界面的上方菜单栏中单击“Model Configuration Parameters”进入如下图所示的界面，在其中单击“Solver”“Slover options”“Type”，在下拉菜单中选择“Fixed-Step”选项。单击其下方的“Additional Parameters”，在“Fixed-Stepsize”一栏中填入变量名“sample-Time”。这样就规定好了仿真模型的固定仿真步长，而变量“sampleTime”值设置在“startSimulation.m”文件中。



Step 5: 观察结果

实验数据记录在“Pos”文件中，结果如图所示。



硬件在环仿真实验分别实现了周期 20s、10s 和 5s 的飞行实验。从实验结果可以看出，随着周期的延长，多旋翼对水平两通道的期望曲线的跟踪效果随着周期的变长而愈加理想，当 $T = 20s$ 时结果相比于 10s 和 5s 更加理想。说明随着周期的缩短，控制效果越来越差。

注意:每运行完一次 Simulink 模型，都要将自驱仪重新插拔，并重新启动 RflySim 工具链，避免出现 RflySim 接收不到指令的问题。

(具体操作步骤可以观看每章相对应的 PPT，其中由各个实验相对应的演示视频。)

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: