

微小型固定翼无人机飞行控制 设计与实践

第5章 系统辨识实验

全 权

qq_buaa@buaa.edu.cn

自动化科学与电气工程学院



北京航空航天大学
BEIHANG UNIVERSITY



可靠飞行控制研究组

RELIABLE FLIGHT CONTROL GROUP

ni 智能工业化·智能制造·无人系统实训·RflySim实训

微小型固定翼无人机
飞行控制设计与实践

全权 主编
高文瀚 刘国清 陈鑫豪 戴训华
吕书礼 徐琳 李锐 副主编



FIXED-WING UNMANNED
AERIAL VEHICLE

中国工业出版社 电子工业出版社

大纲



- 1. 实验原理**
- 2. 基础实验**
- 3. 分析实验**
- 4. 设计实验**
- 5. 硬件在环仿真实验**
- 6. 本章小结**

- 1、先验知识
- 2、实验设计
- 3、数据采集
- 4、模型选择
- 5、模型计算
- 6、模型检验

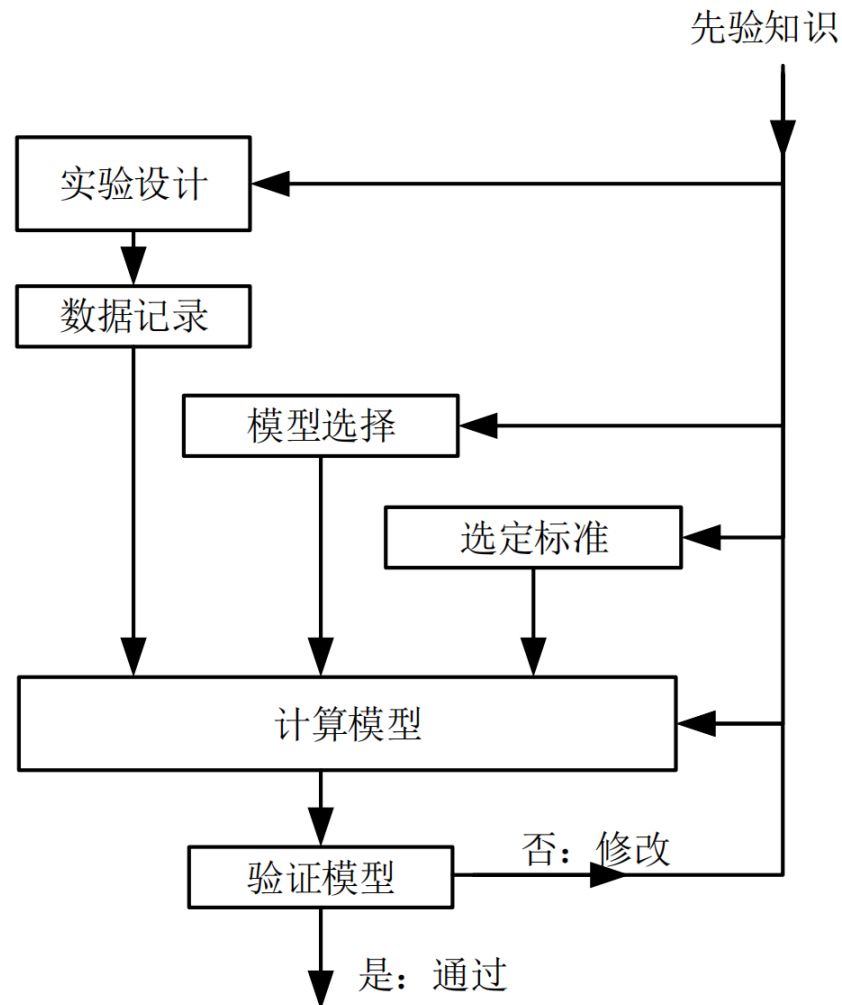


图. 系统辨识步骤

假定自驾仪控制空速、高度和航迹偏角

$$\dot{p}_{x_e} = V_a \cos \psi + v_{wx}$$

$$\dot{p}_{y_e} = V_a \sin \psi + v_{wy}$$

$$\psi = \chi + f(v_{wx}, v_{wy})$$

速度通道 $V_a(s) = G_{V_a}(s) V_{ad}(s)$ 一阶

航向通道 $\chi(s) = G_{\chi}(s) \chi_d(s)$ 二阶

高度通道 $H(s) = G_H(s) H_d(s)$ 二阶

→使用 **MATLAB 系统辨识工具箱**:

提供 MATLAB 函数、Simulink 函数模块以及一个可利用测得的输入输出数据构造动态系统的应用程序。

→使用**CIFER**

(Comprehensive Identification

From Frequency Response) 工具箱:

基于综合频域响应方法进行系统辨识的一款集成工具箱，适用于解决复杂的系统辨识问题。

气动参数辨识——固定翼无人机动力学方程

- 力学方程
$$f_{ax} = \frac{1}{2} \rho V_a^2 S (C_{D0} + C_{D\alpha} \alpha + C_{D\dot{\alpha}} \frac{c}{2V_a} \dot{\alpha} + C_{Dq} \frac{c}{2V_a} \omega_{y_b} + C_{D\delta_e} \delta_e)$$
$$f_{ay} = \frac{1}{2} \rho V_a^2 S (C_{Y0} + C_{Y\beta} \beta + C_{Yp} \frac{b}{2V_a} \omega_{x_b} + C_{Yr} \frac{b}{2V_a} \omega_{z_b} + C_{Y\delta_a} \delta_a + C_{Y\delta_r} \delta_r)$$
$$f_{az} = \frac{1}{2} \rho V_a^2 S (C_{L0} + C_{L\alpha} \alpha + C_{L\dot{\alpha}} \frac{c}{2V_a} \dot{\alpha} + C_{Lq} \frac{c}{2V_a} \omega_{y_b} + C_{L\delta_e} \delta_e).$$
- 力矩方程
$$m_{ax} = \frac{1}{2} \rho V_a^2 S b (C_{l0} + C_{l\beta} \beta + C_{lp} \frac{b}{2V_a} \omega_{x_b} + C_{lr} \frac{b}{2V_a} \omega_{z_b} + C_{l\delta_a} \delta_a + C_{l\delta_r} \delta_r)$$
$$m_{ay} = \frac{1}{2} \rho V_a^2 S c (C_{m0} + C_{m\alpha} \alpha + C_{m\dot{\alpha}} \frac{c}{2V_a} \dot{\alpha} + C_{mq} \frac{c}{2V_a} \omega_{y_b} + C_{m\delta_e} \delta_e)$$
$$m_{az} = \frac{1}{2} \rho V_a^2 S b (C_{n0} + C_{n\beta} \beta + C_{np} \frac{b}{2V_a} \omega_{x_b} + C_{nr} \frac{b}{2V_a} \omega_{z_b} + C_{n\delta_a} \delta_a + C_{n\delta_r} \delta_r)$$
- 方程中的**气动参数**即是本章需要进行辨识的参数

气动参数获取——方法分类

(1) 基于**风洞实验**获取气动参数。风洞的原理是使用动力装置在一个专门设计的管道内驱动一股可控气流，使其流过安置在实验段的静止模型，模拟实物在静止空气中的运动。



图. 风洞

(2) 基于**计算流体力学**获取气动参数 (Computational Fluid Dynamics, CFD)。是通过计算机和数值方法来求解流体力学的控制方程，对流体力学问题进行模拟和分析。

→成本低、速度快、资料完备且可模拟各种不同的工况

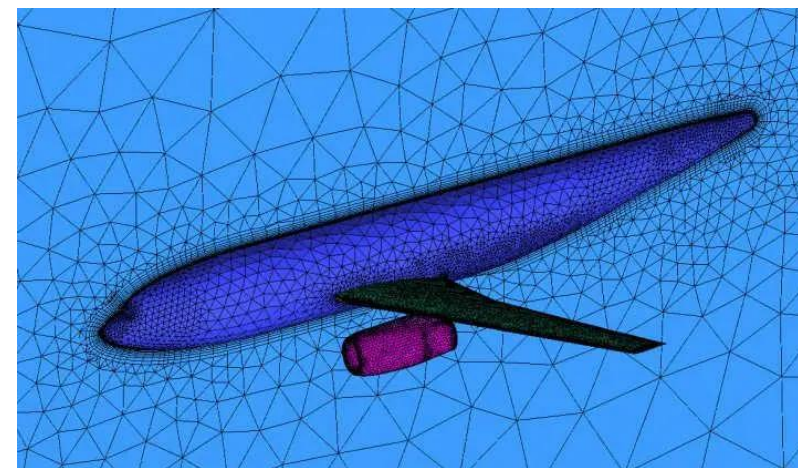


图. 采用 CFD 软件对飞行器模型环境进行网格剖分

(3) 基于**参数辨识**获取气动参数。

气动参数辨识——激励信号

选取偶极方波信号比较合适作为输入信号，信号频谱密度较宽，在高频区域也含有能量：

$$\delta(t) = \begin{cases} 0 & t \in [0, t_1), t \in (t_1 + 2\Delta t, +\infty), \\ a & t \in [t_1, t_1 + \Delta t), \\ -a & t \in [t_1 + \Delta t, t_1 + 2\Delta t) \end{cases}$$

完成飞行实验后导出数据，进行数据预处理后，可以使用[最小二乘法及逐步回归法](#)对气动参数进行辨识。

最小二乘估计算法的数学形式：

假设已知因变量 y 与自变量 $x_1, x_2, \dots, x_n$ 间呈线性关系，即

$$\hat{y} = \lambda_0 + \lambda_1 x_1 + \lambda_2 x_2 + \dots + \lambda_l x_l$$

式中 $\lambda_1, \dots, \lambda_n$ 为未知系数。

气动参数辨识——最小二乘法

为确定这些系数，通常需要采集 m 组观测数据 $(x_1^{(k)}, \dots, x_n^{(k)}, y), k = 1, \dots, m$ 。定义

$$\mathbf{A} = \begin{bmatrix} x_1^{(1)} & x_2^{(1)} & \cdots & x_n^{(1)} \\ x_1^{(2)} & x_2^{(2)} & & \\ \vdots & & \ddots & \\ x_1^{(m)} & & & x_n^{(m)} \end{bmatrix}, \mathbf{b} = \begin{bmatrix} y^{(1)} \\ y^{(2)} \\ \vdots \\ y^{(m)} \end{bmatrix}, \boldsymbol{\lambda} = \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_n \end{bmatrix}, \hat{\boldsymbol{\lambda}} = \begin{bmatrix} \hat{\lambda}_1 \\ \hat{\lambda}_2 \\ \vdots \\ \hat{\lambda}_n \end{bmatrix}$$

式中 $\hat{\boldsymbol{\lambda}}$ 为 $\boldsymbol{\lambda}$ 的最优估计，满足

$$\hat{\boldsymbol{\lambda}} = \arg \min_{\boldsymbol{\lambda}} \left\| \mathbf{A} \hat{\boldsymbol{\lambda}} - \mathbf{b} \right\|^2$$

当 $\mathbf{A}^T \mathbf{A}$ 非奇异时， $\boldsymbol{\lambda}$ 的最优估计为

$$\hat{\boldsymbol{\lambda}} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{b}.$$

逐步回归法的数学思想为：将变量逐一引入回归方程中，在引入时首先判断其偏回归平方和的检验结果是否显著；如果显著则引入，不显著则剔除。

设自变量 x_j ($j = 1, 2, \dots, n$) 共有 n 个，因变量为 y ，共有 m 个观测点。使用最小二乘法进行估计并遍历所有变量，则 y 的估计值为：

$$\hat{y} = \lambda_0 + \lambda_1 x_1 + \lambda_2 x_2 + \dots + \lambda_l x_l$$

式中， $l < n$ ， x_i ($i = 1, 2, \dots, l$) 为从 n 个自变量中按选出来的经统计检验后显著的因子。

逐步回归法在 MATLAB 中有相应的函数，本书使用 “**regress**” 函数和 “**stepwises**” 函数进行参数辨识的数据处理。

- 给出辨识的参数
- 给出用于评价模型参数是否准确的参数——**回归优度 (coefficient of determination)**，记为 R^2 。

给定一系列真值 y_i 和对应的预测值 $\hat{y}_i$ ， R^2 的定义为

$$R^2 = 1 - \frac{\sum_i (\hat{y}_i - y_i)^2}{\sum_i (\hat{y}_i - \bar{y})^2}$$

式中 $\bar{y}$ 为所有真值 y_i 的平均值：

$$\bar{y} = \frac{\sum_i y_i}{m}$$

回归优度 R^2 ：预测值解释了 变量的方差的多大比例，衡量的是预测值对于真值的拟合好坏程度。

- **$R^2 = 1$** ：最理想情况，所有的预测值等于真值。
- **$R^2 = 0$** ：一种可能情况是“简单预测所有 y 值等于 y 平均值”，即所有都等于 （即真实 y 值的平均数），但也有其它可能。
- **$R^2 < 0$** ：模型预测能力差，“简单预测所有 y 值等于 y 平均值”的效果还差。这表示可用了错误模型，或者模型假设不合理。指标 R^2 的最小值没有下限，因为预测可以任意程度的差。因此， R^2 的范围是 $(-\infty, 1]$ 。

注意： R^2 并不是某个数的平方，因此可以是负值。

大纲



1. 实验原理

2. 基础实验

3. 分析实验

4. 设计实验

5. 硬件在环仿真实验

6. 本章小结

实验目标

已知

- (1) 软件：MATLAB R2022b 或以上版本。
- (2) 程序：实验指导包 “e3/e3-1”。实验指导包中主要有：固定翼速度通道 “SpeedChannel”、航向通道 “HeadingChannel” 以及高度通道 “HeightChannel” 制导模型系统辨识示例文件夹。其中 “HeightChannel” 文件夹主要包含飞行日志中高度等指令信号和实际信号的记录文件 “log_trajectory_setpoint_0.csv” 和 “log_vehicle_local_position_0.csv”，从日志文件导出的数据 “HeightChannel_Input.mat” 和 “HeightChannel_Output.mat”，数据预处理脚本 “HeightChannel_DataPreprocess.mlx” 和辨识结果 “HeightChannel_Identification.sid”。

目标

- (1) 了解固定翼无人机制导模型，掌握数据时间配准预处理方法（包括起止时间对齐和采样频率统一），学习 MATLAB 系统辨识工具箱用法。
- (2) 使用速度、航向和高度三个通道采集好的输入输出数据，使用 MATLAB 系统辨识工具箱完成固定翼制导模型的辨识。

实验步骤

步骤一：导入固定翼飞行日志文件，并进行预处理。

(1) 使用 MATLAB 自带的“数据导入”功能可以方便地将“log_trajectory_setpoint_0.csv”和“log_vehicle_local_position_0.csv”文件分别导入 MATLAB 工作区并储存为“HeightChannel_Input.mat”和“HeightChannel_Output.mat”文件，导入数据界面如下：

导入 - D:\MyWorkSpace\Programming\MyLatex\6-1高度垂直\log_trajectory_setpoint_0.csv

导入

视图

分瓣文件

列分隔符:

逗号

分隔符选项

分隔符

范围:

D1:D906

输出类型:

数值矩阵

☐ 排除具有以下项的行：无法导入的单元格

导入内容

所选内容

导入

变量名称行:

1

文本选项

无法导入的单元格

log_trajectory_setpoint_0.csv

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	
数值	数值	数值	数值	数值	数值	数值	数值	数值	数值	数值	数值	数值	数值	数值	数值	数值	数值	
1	timestamp	x	y	z	yaw	yawspeed	vx	vy	vz	accelerati...	accelerati...	accelerati...	jerk[0]	jerk[1]	jerk[2]	thrust[0]	thrust[1]	thrust[2]
2	6588000	0.0011952...	-0.005646...	0.0560281...	1.5777836	0.0	-0.000187...	0.0007393...	0.6992003	-0.009995...	0.0579748...	-0.079996...	0.0	0.0	-4.0	nan	nan	nan
3	6608000	0.0011553...	-0.005424...	0.0556089...	1.577726	0.0	0.0001881...	-0.000740...	0.6992003	0.0100383...	-0.058220...	-0.079996...	0.0	0.0	-4.0	nan	nan	nan
4	6788000	-0.000816...	-0.002349...	0.0561871...	1.578058	0.0	-0.000193...	0.0007510...	0.69920015	-0.010369...	0.0602331...	-0.079992...	0.0	0.0	-4.0	nan	nan	nan
5	6988000	-0.003768...	0.0012528...	0.0566704...	1.5778481	0.0	4.957848e...	0.0007619...	0.69920015	2.4795532...	0.06258106	-0.079992...	0.0	0.0	-4.0	nan	nan	nan
6	7188000	-0.005215...	0.0036645...	0.0608831...	1.5778363	0.0	8.389741e...	0.0007719...	0.6992003	4.196167e...	0.0650301	-0.079996...	0.0	0.0	-4.0	nan	nan	nan
7	7388000	-0.006215...	0.0061653...	0.06474302	1.5776709	0.0	1.0486302...	0.0007806...	0.6992001	5.2452087...	0.0675830...	-0.079994...	0.0	0.0	-4.0	nan	nan	nan
8	7588000	-0.006609...	0.0069887...	0.0697711...	1.5777553	0.0	2.8968421...	-4.781644...	0.6672223	0.0002317...	-0.000398...	-0.333333...	0.0	0.0	0.0	nan	nan	nan
9	7788000	-0.006599...	0.0069712...	0.07238165	1.5773376	0.0	7.5315096...	-0.000127...	0.60055846	0.0002317...	-0.000398...	-0.333333...	0.0	0.0	0.0	nan	nan	nan
10	7988000	-0.006579...	0.0069377...	0.07472348	1.5770252	0.0	0.0001216...	-0.000207...	0.5338955	0.0002317...	-0.000398...	-0.333333...	0.0	0.0	0.0	nan	nan	nan
11	8188000	-0.006550...	0.0068883...	0.07492957	1.5770103	0.0	0.0001680...	-0.000286...	0.4672289	0.0002317...	-0.000398...	-0.333333...	0.0	0.0	0.0	nan	nan	nan
12	8388000	-0.006512...	0.0068229...	0.0668696	1.5774044	0.0	0.0002143...	-0.000366...	0.4005623	0.0002317...	-0.000398...	-0.333333...	0.0	0.0	0.0	nan	nan	nan
13	8588000	-0.006464...	0.0067416...	0.0588177...	1.5775279	0.0	0.0002607...	-0.000446...	0.33389568	0.0002317...	-0.000398...	-0.333333...	0.0	0.0	0.0	nan	nan	nan
14	8788000	-0.006408...	0.0066443...	0.0529380...	1.577529	0.0	0.0003070...	-0.000526...	0.26722908	0.0002317...	-0.000398...	-0.333333...	0.0	0.0	0.0	nan	nan	nan
15	8988000	-0.006341...	0.0065311...	0.04914136	1.5776047	0.0	0.0003534...	-0.000605...	0.20056248	0.0002317...	-0.000398...	-0.333333...	0.0	0.0	0.0	nan	nan	nan
16	9188000	-0.006266...	0.0064020...	0.0422293...	1.5779335	0.0	0.0003997...	-0.000685...	0.13389587	0.0002317...	-0.000398...	-0.333333...	0.0	0.0	0.0	nan	nan	nan
17	9388000	-0.006182...	0.0062569...	0.0369466...	1.5780526	0.0	0.0004461...	-0.000765...	0.0672292...	0.0002317...	-0.000398...	-0.333333...	0.0	0.0	0.0	nan	nan	nan
18	9588000	-0.006088...	0.0060958...	0.0337519...	1.5782366	0.0	0.0004924...	-0.000845...	0.0005625...	0.0002317...	-0.000398...	-0.333333...	0.0	0.0	0.0	nan	nan	nan
19	9788000	-0.005985...	0.0059188...	0.03215249	1.5782241	0.0	0.0005387...	-0.000924...	-0.066104...	0.0002317...	-0.000398...	-0.333333...	0.0	0.0	0.0	nan	nan	nan
20	9988000	-0.005872...	0.0057259...	0.0279564...	1.5779749	0.0	0.0005851...	-0.001004...	-0.1327708	0.0002317...	-0.000398...	-0.333333...	0.0	0.0	0.0	nan	nan	nan
21	10188000	-0.005751...	0.0055170...	0.02333974	1.5777618	0.0	0.0006314...	-0.001084...	-0.199437...	0.0002317...	-0.000398...	-0.333333...	0.0	0.0	0.0	nan	nan	nan
22	10388000	-0.005620...	0.0052922...	0.0169208...	1.577472	0.0	0.0006778...	-0.001163...	-0.266104	0.0002317...	-0.000398...	-0.333333...	0.0	0.0	0.0	nan	nan	nan
23	10588000	-0.005481...	0.0050534...	0.0122914...	1.5773691	0.0	0.0007006...	-0.001203...	-0.332770...	-4.768371...	7.1525574...	-0.333333...	0.0	0.0	0.0	nan	nan	nan
24	10788000	-0.005341...	0.00481305	0.00785004	1.5772434	0.0	0.0006989...	-0.001200...	-0.399437...	-1.215934...	1.9311905...	-0.333333...	0.0	0.0	0.0	nan	nan	nan
25	10988000	-0.005201...	0.0045734...	0.0045060	1.5772492	0.0	0.0006956...	-0.001195...	-0.466103...	-1.978874...	3.2901764...	-0.333333...	0.0	0.0	0.0	nan	nan	nan
26	11188000	-0.005062...	0.00433511	0.0033159...	1.5773844	0.0	0.0006907...	-0.001187...	-0.532770...	-2.884864...	4.839897e...	-0.333333...	0.0	0.0	0.0	nan	nan	nan
27	11388000	-0.004925...	0.0040987...	0.0040940...	1.5772805	0.0	0.0006838...	-0.001175...	-0.599437...	-3.910064...	6.604195e...	-0.333333...	0.0	0.0	0.0	nan	nan	nan
28	11588000	-0.004789...	0.0038651...	0.0036906...	1.5771601	0.0	0.0006747...	-0.001160...	-0.666103...	-5.125999...	8.6545944...	-0.333333...	0.0	0.0	0.0	nan	nan	nan

图. 固定翼飞行日志文件

实验步骤

步骤一：导入固定翼飞行日志文件，并进行预处理。

(2) 对输入输出数据进行时间配准处理。

- 起始和终止时间的配准：需要观察输入输出数据，选取适用于辨识的数据段
- 进行采样频率的配准：需要通过重采样的方式使输入输出数据时间戳对准

```
1 clear
2 HeightData1=importdata( "HeightChannel_Input.mat"); %import data
3 HeightData2=importdata( "HeightChannel_Output.mat");
4 HeightDes=table2array(HeightData1(320:671,2)); %transform table to array
5 Height=table2array(HeightData2(800:1501,2));
6 HeightDesResample=[]; %resample data
7 for i=1:2:size(Height)-1
8     HeightDesResample(i,1)=HeightDes((i+1)/2,1);
9     HeightDesResample(i+1,1)=HeightDes((i+1)/2,1);
10 end
```

清空工作区

导入数据

截取数据并转换为矩阵格式

重采样

速度通道数据预处理代码，来源于 “e3/e3-1/SpeedChannel_DatePreprocess.mlx”

步骤一：导入固定翼飞行日志文件，并进行预处理。

(3) 使用“plot”指令绘制出

“HeightDesResample”和“Height”变量，即可得到的用于辨识的输入输出数据曲线。

- 高度期望曲线（与“HeightDesResample”相对应）：在-82.5与-87.5两个值之间阶跃变化的
- 实际高度曲线（与“Height”变量相对应）：跟踪期望高度曲线变化，但是存在一定的过渡过程和跟踪误差。

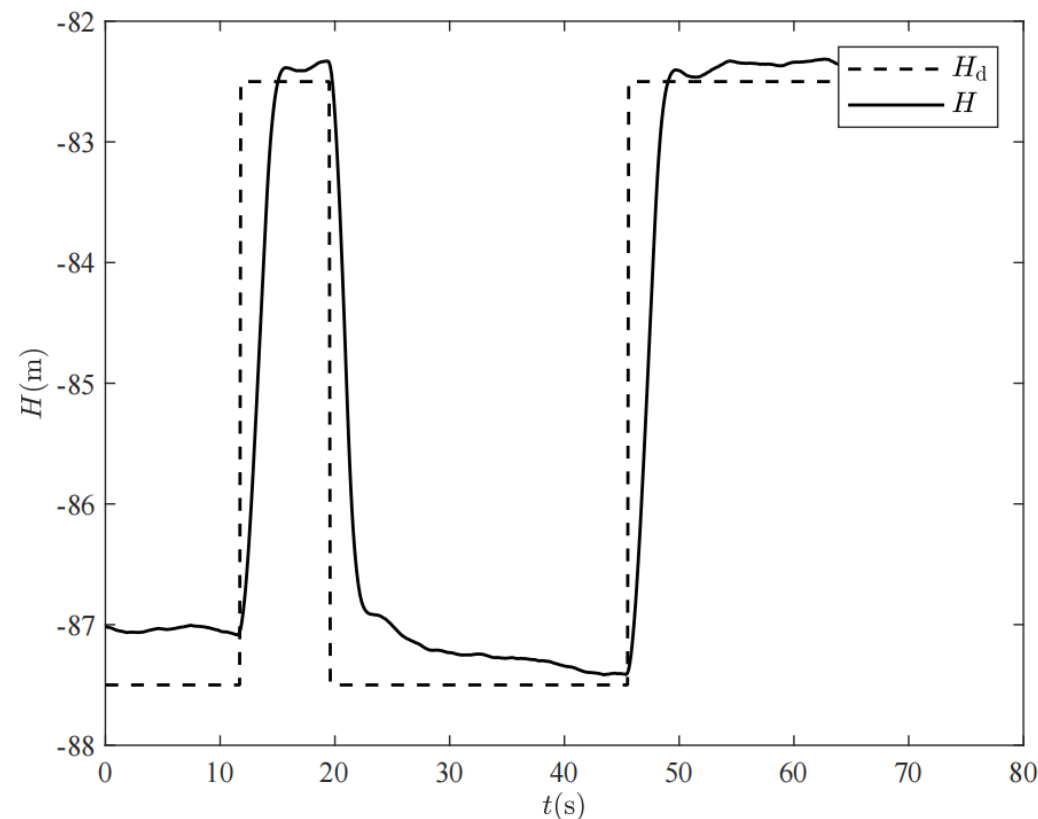


图. 高度通道输入输出数据曲线

步骤二：使用 MATLAB 系统辨识工具箱进行系统辨识

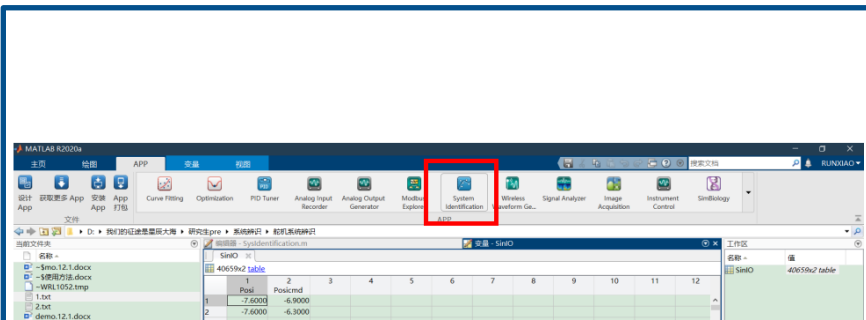


图. 打开工具箱

打开 MATLAB 的系统辨识工具箱
“APP-System Identification”。

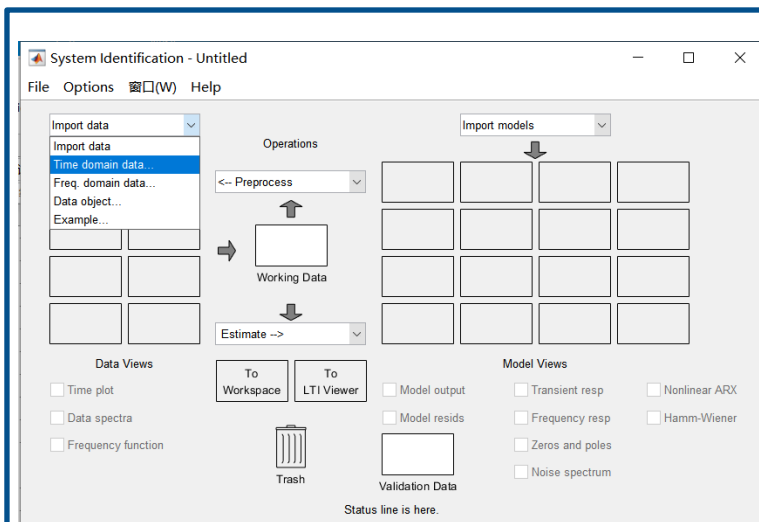


图. 工具箱界面

点击 “Import data”

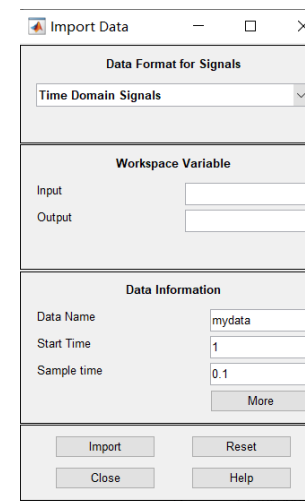


图. 时域设置界面

选择时域数据 “Time domain data”; 设置输入 “input” 为 “HeightDesResample”, 设置输出 “output” 为 “Height”, 设置采样时间 “Sample time” 为 0.1s; 点击 “Import” 即可导入数据。

步骤二：使用 MATLAB 系统辨识工具箱进行系统辨识

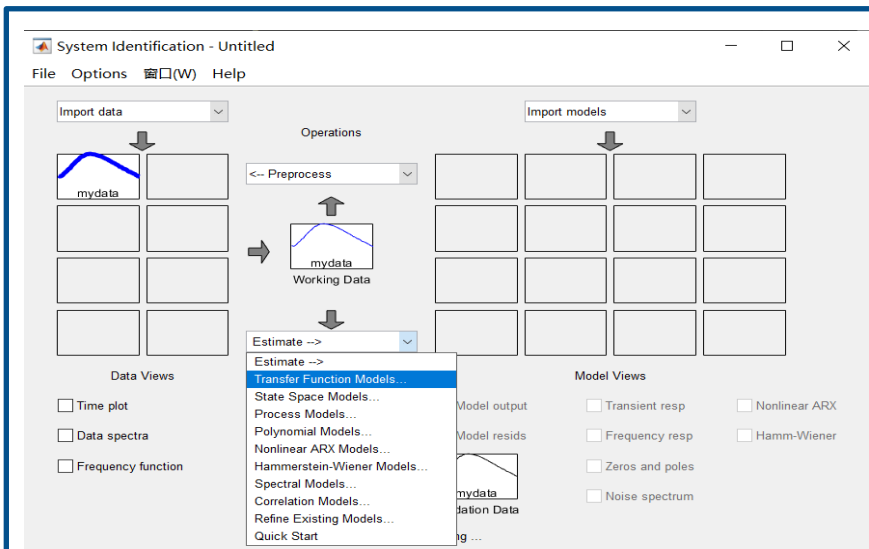


图. 建立传递函数

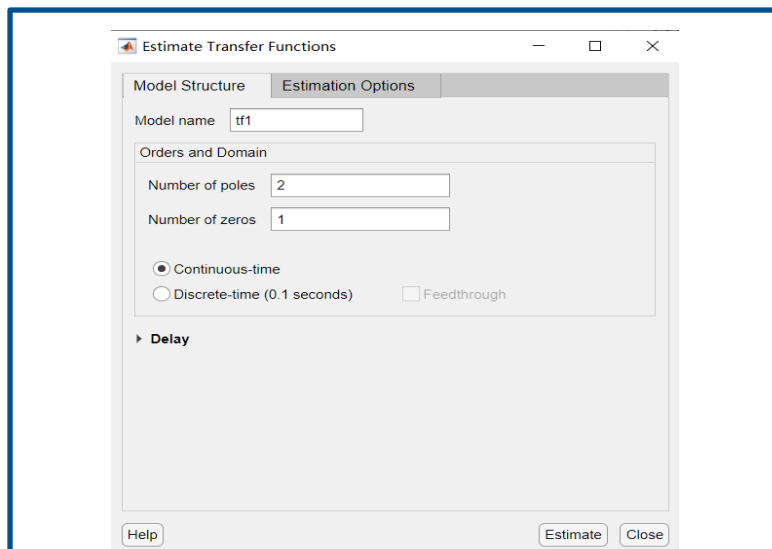


图. 设置传递函数

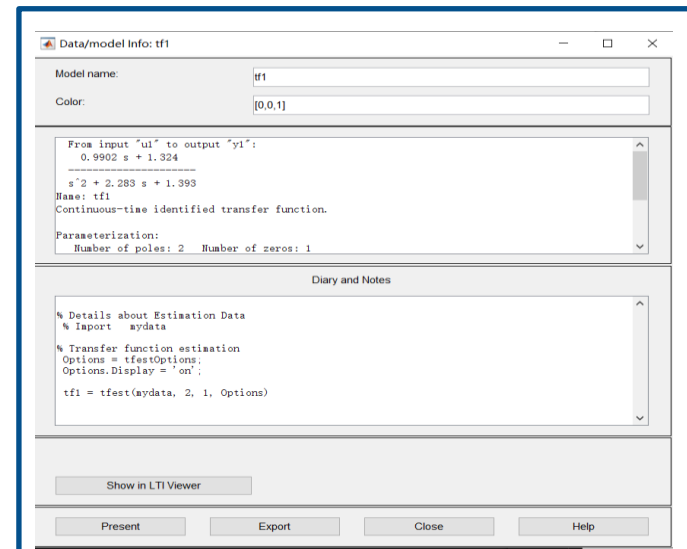


图. 传递函数

点击 “Estimate”，选择传递函数模型 “Transfer Function Models”；

设置模型的零点个数 “Number of zeros” 为 0，极点个数 “Number of poles” 为 1，选择连续传递函数 “Continuous-time”，点击 “Estimate”，辨识传递函数模型。

返回工具箱界面，双击建立好的系统模型就可以显示得到传递函数以及匹配 “Fits”。

步骤三：验证辨识结果

通过步骤二得到传递函数后，选择已建立好的模型，在工具箱界面右下角勾选“Model output”即可得到模型在输入信号作用下的响应曲线和匹配度。通常来说，匹配度大于 70% 则认为辨识效果较好。

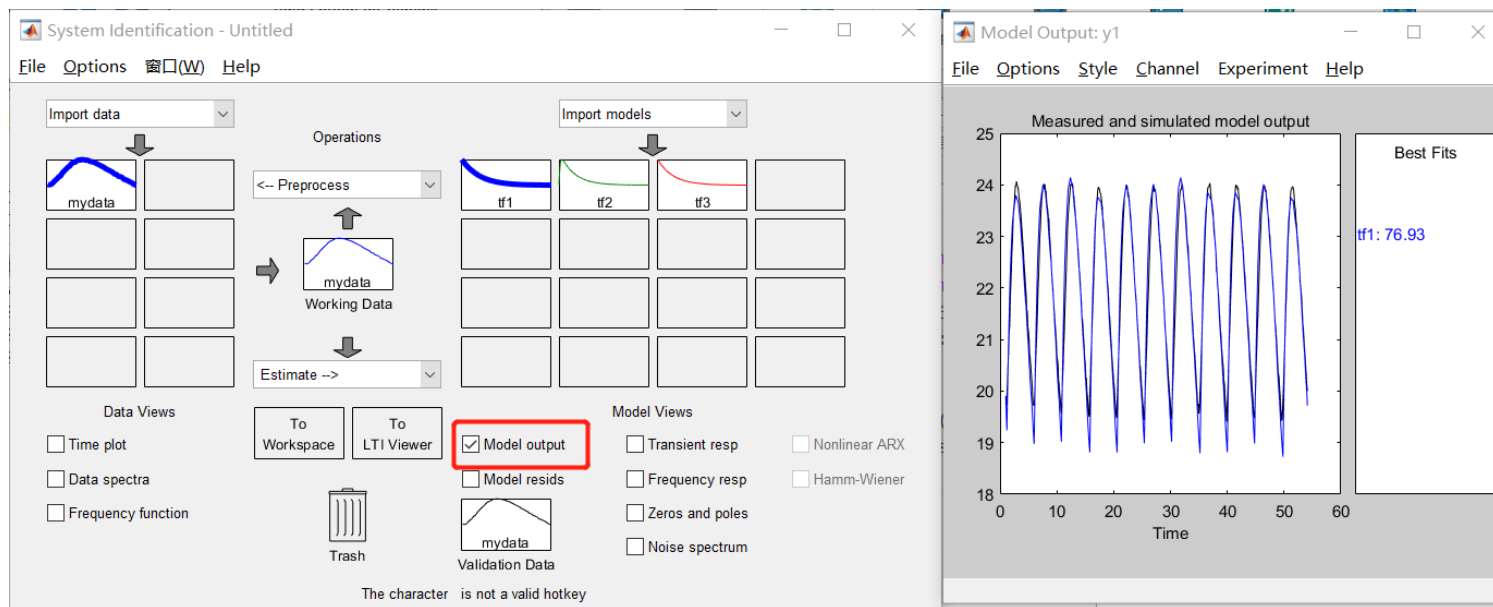


图. 模型输出

步骤三：验证辨识结果

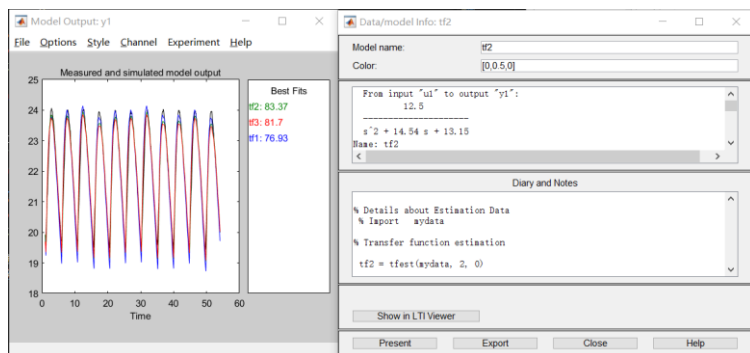


图. 速度通道辨识

速度通道：其中“tf1”、“tf2”、“tf3”曲线分别代表选择不同数量的零极点得到的辨识结果。对比可得，“tf2”即 2 个极点 0 个零点建立的传递函数模型匹配度最高，达到了 83.37%，符合预期。

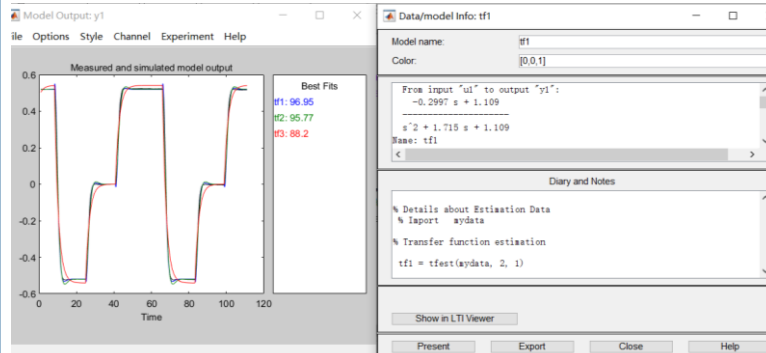


图. 航向通道辨识

航向通道：其中“tf1”、“tf2”、“tf3”曲线分别代表选择不同数量的零极点得到的辨识结果。对比可得，“tf1”即 2 个极点 1 个零点建立的传递函数模型匹配度最高，达到了 96.95%，符合预期。

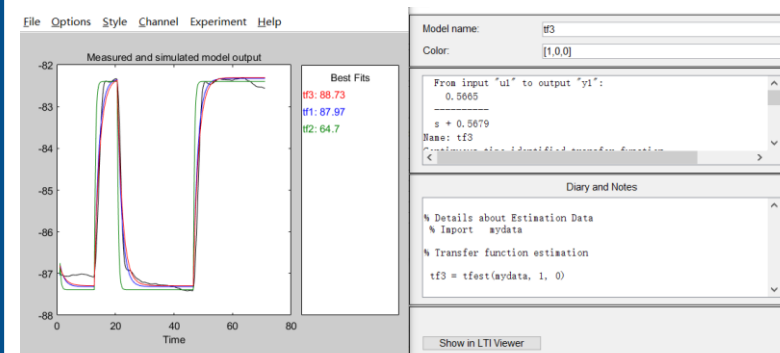


图. 高度通道辨识

高度通道：其中“tf1”、“tf2”、“tf3”曲线分别代表选择不同 Va 数量的零极点得到的辨识结果。对比可得，“tf3”即 1 个极点 0 个零点建立的传递函数模型匹配度最高，达到了 88.73%，符合预期。

实验步骤

步骤三：验证辨识结果

综上所述，可以得到辨识出的制导模型

p_{x_e} : 固定翼无人机在北东地坐标系中的北向位置

p_{y_e} : 固定翼无人机在北东地坐标系中的东向位置

$$\dot{p}_{x_e} = V_a \cos \psi$$

$$\dot{p}_{y_e} = V_a \sin \psi$$

$$\psi = \chi$$

速度通道

$$V_a(s) = \frac{12.5}{s^2 + 14.54s + 13.15} V_{ad}(s)$$

航向通道

$$\chi(s) = \frac{-0.2997s + 1.109}{s^2 + 1.715s + 1.109} \chi_d(s)$$

高度通道

$$H(s) = \frac{0.5665}{s + 0.5679} H_d(s)$$

ψ : 偏航角
 χ : 航迹偏角

MATLAB R2022b

HOME

LOTS

APPS

LIVE EDITOR

INSERT

VIEW

New

Open

Save

Print

Export

Go To

Find

Bookmark

Text

Normal

B

I

U

M

Code

Control

Task

Refactor

Run

Section

Run and Advance

Run to End

Run

Step

Stop

FILE

NAVIGATE

TEXT

CODE

SECTION

RUN

Search Documentation

RUNXIAO

Current Folder

HeightChannel_DataPreprocess.mlx

HeightChannel_Identification.sid

HeightChannel_Input.mat

HeightChannel_Output.mat

idprefs.mat

log_trajectory_setpoint_0.csv

log_vehicle_local_position_0.csv

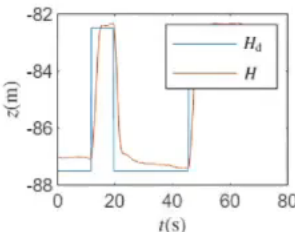
Live Editor - E:\file\Programming\book\fixedwing-book-rfly\fixedwing-book-rfly\ch6\code\e3-1\HeightChannel\HeightChannel_DataPreprocess...

HeightChannel_DataPreprocess.mlx

```
1 clear
2 HeightData1=importdata( "HeightChannel_Input.mat");
3 HeightData2=importdata( "HeightChannel_Output.mat");
4 HeightDes=table2array(HeightData1(320:671,2));
5 Height=table2array(HeightData2(800:1501,2));
6 HeightDesResample=[];
7 for i=1:2:size(Height)-1
8     HeightDesResample(i,1)=HeightDes((i+1)/2,1);
9     HeightDesResample(i+1,1)=HeightDes((i+1)/2,1);
10 end
11
12 % figure (1)
13 % stackedplot(HeightData1);
14 % figure(2)
15 % stackedplot(HeightData2);
```

t = 1x702

0 0.1000 ...



Workspace

Name	Value
------	-------

Command Window

New to MATLAB? See resources for [Getting Started](#).

```
f >>
```

HeightChannel_DataPreprocess.mlx (Liv...

No details available

打开“e3/e3-1/HeightChannel”文件夹

大纲



1. 实验原理
2. 基础实验
3. 分析实验
4. 设计实验
5. 硬件在环仿真实验
6. 本章小结

实验目标

已知

- (1) 软件：MATLAB R2022b 或以上版本。
- (2) 程序：实验指导包“e3/e3-2”。实验指导包主要有：小型固定翼模型“model”文件夹，文件夹中包含小型固定翼模型文件“SmallFixedWingUAVupper.slx”和初始化文件“InitDatactrl.m”；运行模型得到的辨识所需的原始数据“Cm_Data.mat”，数据处理脚本“Cm_DataPreprocess.mlx”，直接用于辨识的数据“Cm_DataFiltered.mat”和俯仰力矩通道气动参数辨识的脚本“Cm_Identification.mlx”。

目标

以俯仰力矩通道为例，辨识如下式中的 $C_{m0}, C_{m\alpha}, C_{m\dot{\alpha}}, C_{mq}, C_{m\delta_e}$ ，了解气动参数辨识的流程和思想。对比分析不同幅值或持续时间的偶极方波或频率变化范围的扫频信号作为激励信号的辨识效果。

$$m_{ay} = \frac{1}{2} \rho V_a^2 S c_A (C_{m0} + C_{m\alpha} \alpha + C_{m\dot{\alpha}} \frac{c}{2V_a} \dot{\alpha} + C_{mq} \frac{c}{2V_a} \omega_{yb} + C_{m\delta_e} \delta_e)$$

步骤一：设置不同幅值和持续时间的偶极方波信号输入

在“SmallFixedWingUAVupper.slx”的“Command”路径下中，可以使用“Signal Builder (PitchChannel)”模块向俯仰通道注入激励。

对于俯仰通道辨识，可以使用不同幅值和持续时间的偶极方波信号作为激励，图中的激励信号为**幅值 20° 持续时间 4s 的偶极方波信号**。

为了防止其它通道对俯仰辨识效果产生影响，需要保持其它通道不变，仅给俯仰通道注入激励信号。

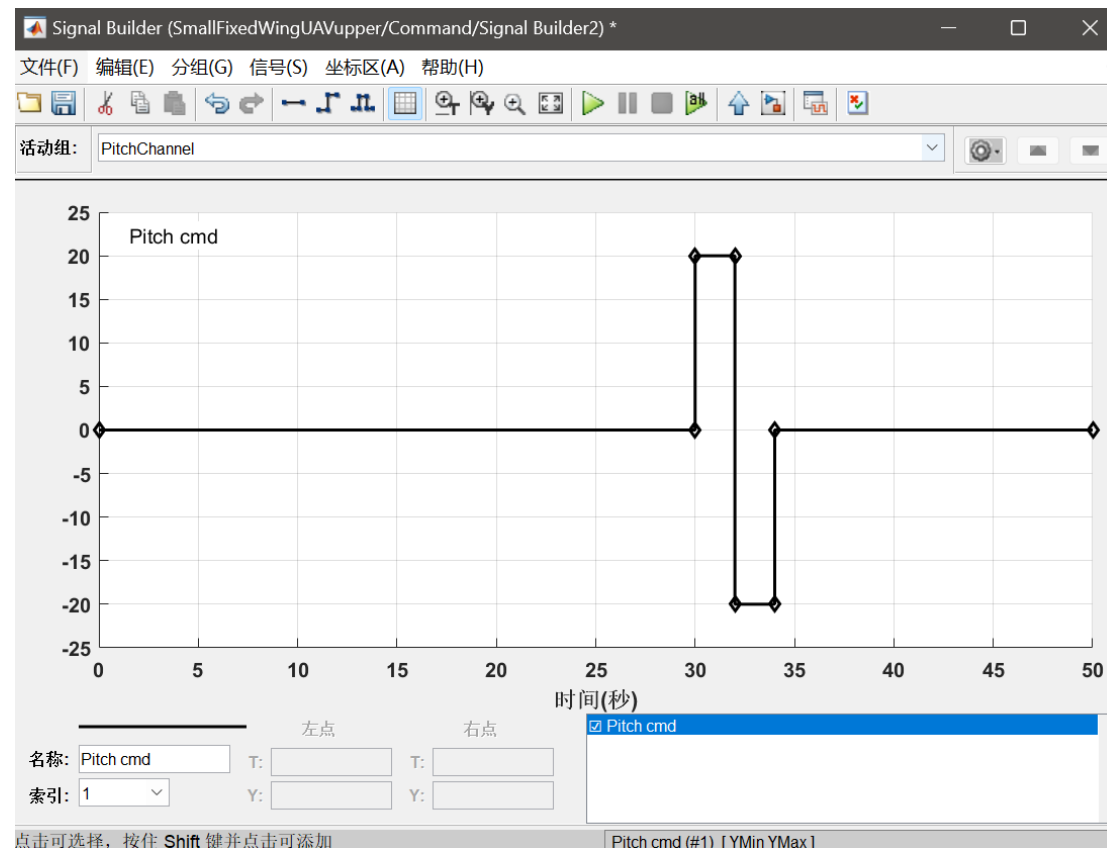


图. 在“Signal Builder”中更改俯仰力矩通道激励信号

步骤二：运行软件仿真并记录所需的各状态量，并进行滤波处理

(1) 打开“model”文件夹，首先运行初始化文件“InitDatactrl.m”，然后运行“SmallFixedWingUAVupper.slx”程序。该程序运行完成后可以自动导出俯仰力矩通道辨识所需的数据，并保存为原始数据文件“Cm_Data.mat”。

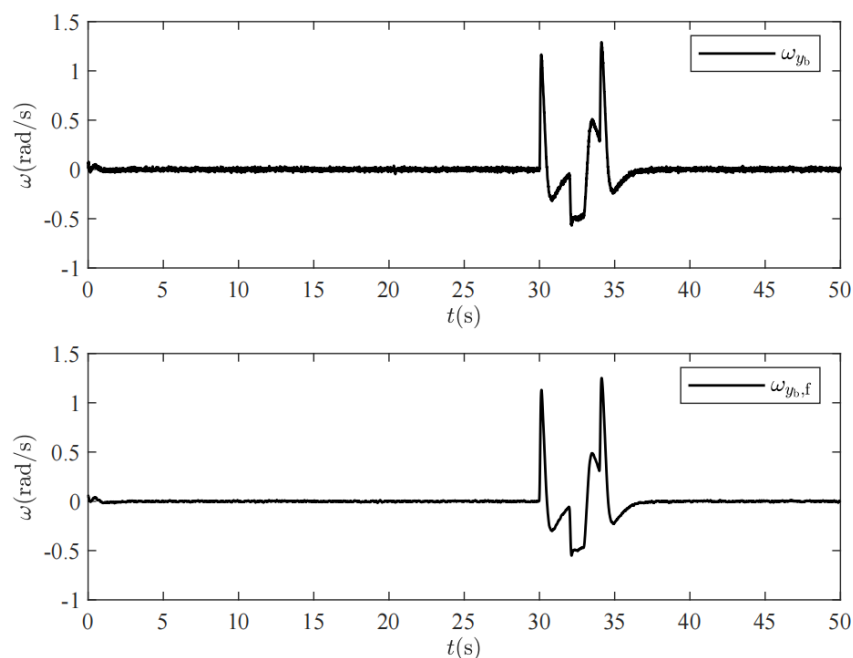


图. 部分数据滤波后效果

(2) 选择 MATLAB 自带的滑动平均滤波器进行处理, 进一步, 可以通过使用 MATLAB 信号分析器观察各信号频谱特征选择滤波器参数, 选择平均滑动滤波窗口长度为15。

- ω_{yb} : 原始数据
 - $\omega_{yb,f}$: 滤波后数据
- ➡ 对比可见滤波效果较为明显。

步骤二：运行软件仿真并记录所需的各状态量，并进行滤波处理

```
1 clear;clc;
2 data = load("Cm_Data.mat");
3 % Split data
4 alpha = data.data(2,:);
5 dE = data.data(3,:);
6 q = data.data(4,:);
7 Airspeed = data.data(5,:);
8 alpha_dot = data.data(6,:);
9 wby_dot = data.data(7,:);
10 % Moving average filter
11 window = 15;
12 data_filtered.alpha=movmean(alpha,window);
13 data_filtered.dE=movmean(dE,window);
14 data_filtered.wby=movmean(q,window);
15 data_filtered.Airspeed=movmean(Airspeed,window);
16 data_filtered.alpha_dot=movmean(alpha_dot,window);
17 data_filtered.wby_dot=movmean(wby_dot,window);
```

俯仰力矩通道数据预处理代码，
来源于“e3/e3-1/Cm_DataPreprocess.mlx”

清空工作区

从“.mat”文件中读取数据并存储为“data”

将“data”拆分为各变量对应数据

对各个状态数据进行窗口长度为 15 的滑动平均滤波处理

“movmean”函数： MATLAB 提供的移动均值函数，其语法为“M = movmean(A,k)”。该函数返回由局部“k”个数据点的均值组成的数组，其中每个均值是基于“A”的相邻元素的长度为“k”的滑动窗计算得出。

- 当“k”为奇数时，窗口以当前位置的元素为中心。
- 当“k”为偶数时，窗口以当前元素及其前一个元素为中心。
- 当没有足够的元素填满窗口时，窗口将自动在端点处截断。
- 当窗口被截断时，只根据窗口内的元素计算平均值。

步骤三：使用最小二乘法和逐步回归法辨识气动参数

(1) 编写代码，选取无人机进行平稳飞行后的数据进行辨识。

- 基于固定翼姿态动力学方程，计算出实际俯仰力矩系数。
- 通过 MATLAB 自带的“regress”和“stepwise”函数进行最小二乘法估计和逐步回归，得到俯仰通道气动参数。

```
1 clear;
2 load('Cm_DataFiltered.mat'); %load data
3 wby=data_filtered.wby(5500:7500)'; %split data
4 wby_dot=data_filtered.wby_dot(5500:7500)';
5 Va=data_filtered.Airspeed(5500:7500)';
6 deltaE=data_filtered.dE(5500:7500)';
7 alpha=data_filtered.alpha(5500:7500)';
8 dalpha=data_filtered.alpha_dot(5500:7500)';
9 % Aircraft geometric parameters
10 Iy=0.144;
11 c=0.3;
12 b=1.2;
13 S=0.32;
14 rho = 1.225;
```

清空工作区

从“.mat”文件中读取数据并存储为“data”

将“data”拆分为俯仰角速率、俯仰角加速度、空速、舵偏角、迎角和迎角变化率等数据

输入了该通道辨识所需的无人机转动惯量以及几何参数等

俯仰力矩通道数据预处理代码，来源于“e3/e3-1/Cm_DataPreprocess.mlx”

步骤三：使用最小二乘法和逐步回归法辨识气动参数

```
15 qbar=zeros(2001,1);
16 Cm=zeros(2001,1);
17 dalpha_after=zeros(2001,1);
18 q_after=zeros(2001,1);
19 for i=1:2001 %calculate qbar Cm dalpha_after q_after
20     qbar(i)=0.5*rho*Va(i)^2;
21     Cm(i)=(Iy*wby_dot(i))/(qbar(i)*c*S);
22     dalpha_after(i)=dalpha(i)*c/(2*Va(i));
23     q_after(i)=wby(i)*c/(2*Va(i));
24 end
25 % regress
26 Cm0=ones(2001,1);
27 XCm=[Cm0,alpha,dalpha_after,q_after,deltaE];
28 [bCm,bintCm,rCm,rintCm,statsCm]=regress(Cm,XCm);
29 % stepwise
30 stepwise(XCm,Cm);
31 Cm_identify=XCm*bCm;
```

计算动压，根据固定翼姿态动力学方程计算俯仰力矩系数，以及对迎角变化率和俯仰角速率进行修正

生成零升力矩项

将用于辨识的状态量数据合并成矩阵

使用“regress”函数进行辨识

使用“stepwise”函数进行辨识

为使用辨识出的气动参数计算俯仰力矩系数

俯仰力矩通道数据预处理代码，来源于“e3/e3-1/Cm_DataPreprocess.mlx”

步骤三：使用最小二乘法和逐步回归法辨识气动参数

“regress”函数：为多元线性回归函数，其语法为“ $[b, bint, r, rint, stats] = regress(y, X)$ ”，返回有

- 向量“b”为“y”中的响应对矩阵“X”中的预测变量的多元线性回归的系数估计值；
- “bint”为系数估计值的 95 % 置信区间的矩阵；
- “r”为残差向量；
- “rint”为可用于诊断离群值的区间，离群值是指与集合中其它观测值差异较大的任何数据点；
- “stats”包含 R2 统计量、以及其它统计量指标（本实验中未用到）。

需要注意的是要计算具有常数项（截距）的模型的系数估计值，在矩阵 X 中要包含一个由 1 构成的列，在此处即为零升力矩项。

“stepwise”函数：本质上与“regress”函数一样，也是多元线性回归，但是进行了可视化显示，可以逐次加入被辨识的数据并进行显著性检测，也可以显示“regress”函数得出的各种结果。

步骤三：使用最小二乘法和逐步回归法辨识气动参数

(2) 辨识结果：以幅值 20 度持续时间 4 秒的偶极方波信号辨识结果为例进行说明。

“stepwise”函数的界面，主要包括三个部分：

- ①中左侧的图为回归系数带有误差条的辨识结果
 - 纵坐标的“X1”至“X4”为参与逐步回归的自变量
 - 横坐标为相应的数值，蓝色的线条表示 90% 的置信区间，粗点表示辨识的结果
- ①中右侧的部分与左侧的各行相对应，共有 3 列：
 - 回归系数“Coeff.”列
 - 其余列为其它的统计量
- ②为最终的回归结果：“截距”、拟合优度 R^2 、“F 统计量”、“RMSE”、“调整 R 方”、“p”。
- ③为依次引入自变量后，“RMSE”的变化情况。

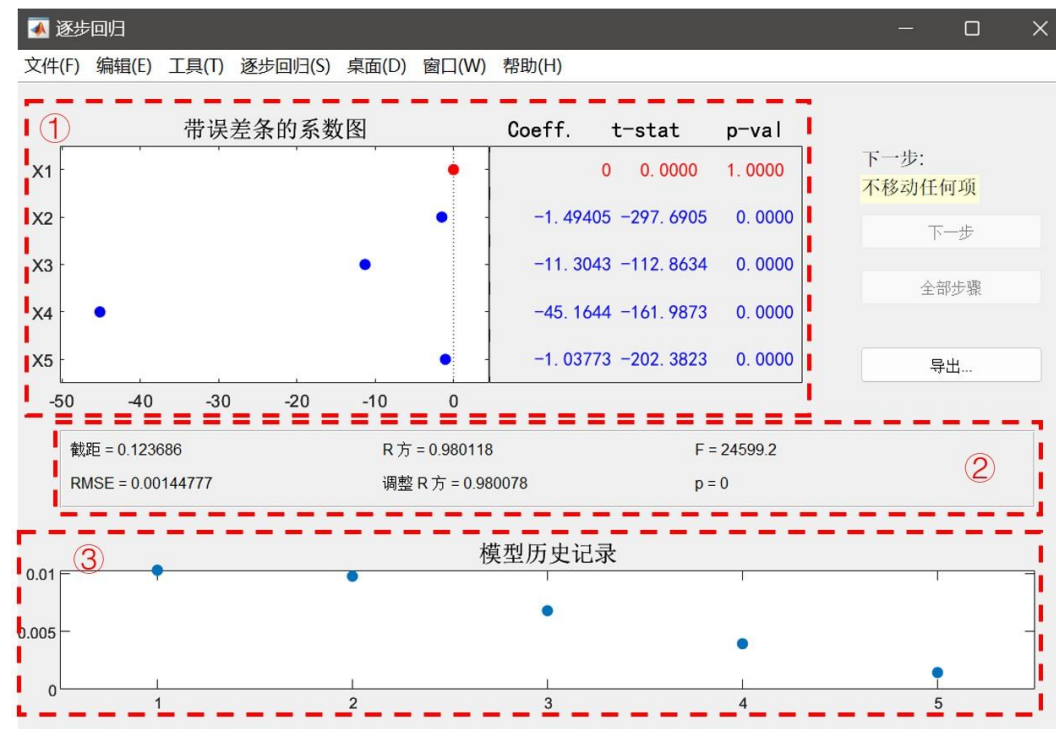


图. 逐步回归

步骤三：使用最小二乘法和逐步回归法辨识气动参数

本例中，“X2”至“X5”为实验中所求的气动参数 $C_{m0}, C_{m\alpha}, C_{m\dot{\alpha}}, C_{mq}, C_{m\delta_e}$ ，而截距则对应 C_{m0} ，拟合优度 R^2 为 0.98，说明回归结果比较精确。在本章的实验中，只需要关注辨识出的**截距、回归系数以及拟合优度 R^2** 就足够了。

- C_m ：俯仰力矩系数真值
- $C_{m,I}$ ：根据辨识出的气动参数拟合出的俯仰力矩系数。

→ 辨识出的气动参数较为准确。

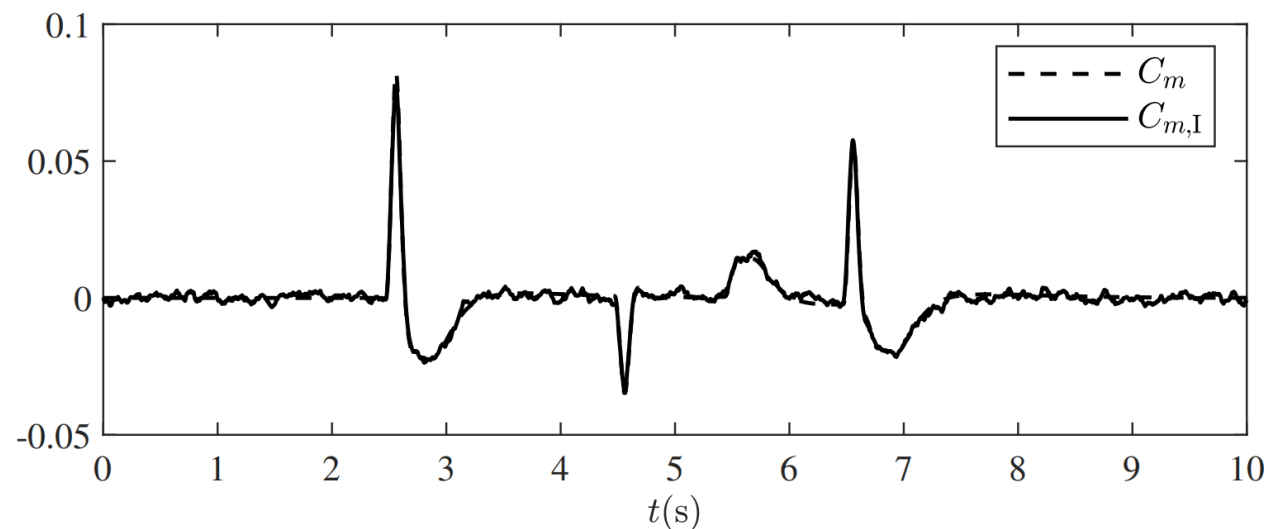


图. 拟合效果

步骤四：验证辨识结果，并比较不同输入信号的辨识精确度。

(1) 辨识结果：

更换不同幅值和持续时间的偶极方波，完成 6 组对比实验，得到表中的数据。

幅值	20°			5°		
长度	4s	6s	8s	4s	6s	8s
R^2	0.980118	0.98354	0.97199	0.956337	0.958191	0.930923

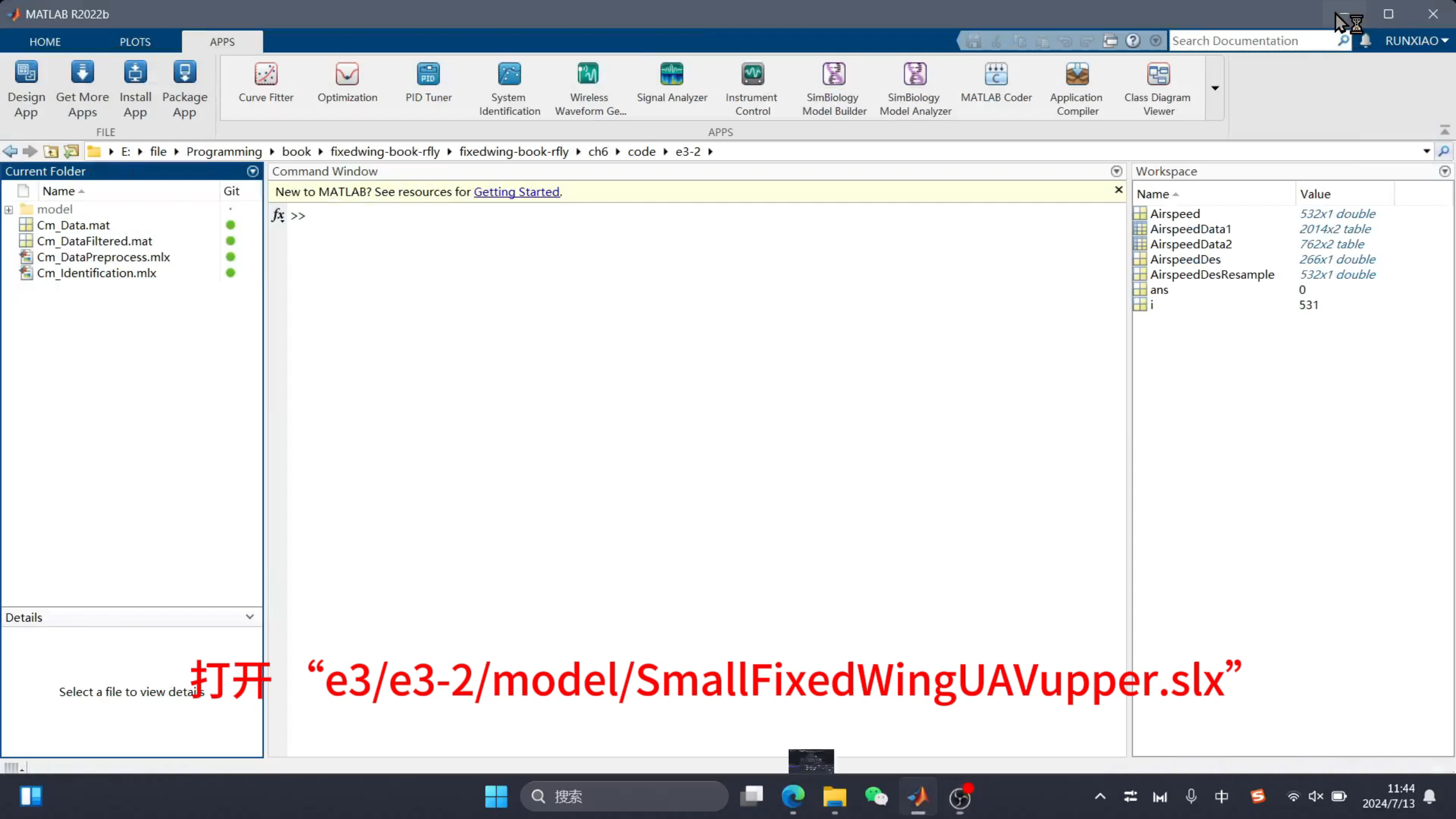
(2) 比较不同输入信号的辨识精确度：

- 采用偶极方波输入的长度为 6s 时的 R^2 比 4s 和 8s 时的 R^2 更大
- 随着偶极方波幅值的增大， R^2 更接近于 1，有较好的拟合效果

(3) 分析：

- 输入信号的幅值对辨识结果的影响。在保证无人机稳定的前提下，输入信号的幅值越大，意味着无人机的作动幅度越大。
 - 更充分地激发无人机相应通道的运动模式，有利于提高辨识精度；
 - 产生的姿态角、姿态角速率等数据的变动更大，有利于将其从噪声数据中分离出来，从而提高辨识的精度。
- 输入偶极方波信号的长度对辨识结果的影响。
 - 偶极方波信号的长度会影响其在频谱上的能量分布，合适的频率能量分布可以更好地激发无人机相应频段的响应，从而提高辨识的精度。

图. 俯仰力矩通道不同偶极方波信号辨识结果



HOME

PLOTS

APPS

Design App

Get More Apps

Install App

Package App

Curve Fitter

Optimization

PID Tuner

System Identification

Wireless Waveform Ge...

Signal Analyzer

Instrument Control

SimBiology Model Builder

SimBiology Model Analyzer

MATLAB Coder

Application Compiler

Class Diagram Viewer

FILE

APPS

Current Folder

Name

Git

model

Cm_Data.mat

Cm_DataFiltered.mat

Cm_DataPreprocess.mlx

Cm_Identification.mlx

Details

Select a file to view details

Command Window

New to MATLAB? See resources for [Getting Started.](#)

fx

>>

Workspace

Name	Value
Airspeed	532x1 double
AirspeedData1	2014x2 table
AirspeedData2	762x2 table
AirspeedDes	266x1 double
AirspeedDesResample	532x1 double
ans	0
i	531

打开“e3/e3-2/model/SmallFixedWingUAVupper.slx”

11:44

2024/7/13

大纲



1. 实验原理
2. 基础实验
3. 分析实验
4. 设计实验
5. 硬件在环仿真实验
6. 本章小结

实验目标

已知

- (1) 软件：MATLAB R2022b 或以上版本。
- (2) 程序：实验指导包“e3/e3-3”。实验指导包主要有：小型固定翼模型“model”文件夹，文件夹中包含小型固定翼模型文件“SmallFixedWingUAVupper.slx”和初始化文件“InitDatactrl.m”；运行模型得到的辨识所需的原始数据“Cl_Data.mat”，数据处理脚本“Cl_DataPreprocess.mlx”，直接用于辨识的数据“Cl_DataFiltered.mat”和俯仰力矩通道气动参数辨识的脚本“Cl_Identification.mlx”。

目标

在固定翼无人机 Simulink 模型中，设计合适的输入信号，激励并采集数据；辨识滚转力矩通道的气动导数，即式中的 $C_{l_0}, C_{l_\beta}, C_{l_p}, C_{l_r}, C_{l_{\delta_a}}, C_{l_{\delta_r}}$ ；最后，验证得到的非线性模型的吻合程度。

$$m_{ax} = \frac{1}{2} \rho V_a^2 S b (C_{l_0} + C_{l_\beta} \beta + C_{l_p} \frac{b}{2V_a} \omega_{x_b} + C_{l_r} \frac{b}{2V_a} \omega_{z_b} + C_{l_{\delta_a}} \delta_a + C_{l_{\delta_r}} \delta_r)$$

步骤一：设置激励信号

从基础实验和分析实验的经验可得：采用持续时间为 4s、幅值为 20° 的偶极方波作为滚转力矩通道的激励信号。

步骤二：分析力矩平衡方程

为了辨识滚转力矩通道，首先需要通过力矩平衡方程计算式中的滚转力矩 m_{ax} 。力矩平衡方程为：

$$m_{ax} \approx J_{xx} \dot{\omega}_{x_b}$$

根据该方程，利用估计的滚转角加速度 $\dot{\omega}_{x_b}$ 和已知的转动惯量 J_{xx} 可以计算出滚转力矩 m_{ax} 。

步骤三：分析力矩平衡方程

根据固定翼无人机气动方程，有

$$m_{ax} = \frac{1}{2} \rho V_a^2 S b (C_{l_0} + C_{l_\beta} \beta + C_{l_p} \frac{b}{2V_a} \omega_{x_b} + C_{l_r} \frac{b}{2V_a} \omega_{z_b} + C_{l_{\delta_a}} \delta_a + C_{l_{\delta_r}} \delta_r)$$

只要获得了状态量 $\rho, V_a, \beta, \omega_{x_b}, \omega_{z_b}, \delta_a, \delta_r$ 以及几何参数 S, b ，即可通过最小二乘法得到需要辨识的参数 $C_{l_0}, C_{l_\beta}, C_{l_p}, C_{l_r}, C_{l_{\delta_a}}, C_{l_{\delta_r}}$ 。

步骤四：气动角获取

在 MATLAB 软件仿真中，假设使用气动角传感器可以测得迎角 α 和侧滑角 β 。对于一些并没有安装气动角传感器的固定翼无人机，则可以采用估计的方式来获得迎角 α 和侧滑角 β 。估计方法如下：一般情况下飞行日志中会提供四元数和地面坐标系下的速度，从四元数可以按照式便地解算出无人机姿态角 (ϕ, θ, ψ) 如下：

$$\phi = \arctan \frac{2(q_0 q_1 + q_2 q_3)}{1 - 2(q_1^2 + q_2^2)}$$

$$\theta = \arcsin(2(q_0 q_2 - q_1 q_3))$$

$$\psi = \arctan \frac{2(q_0 q_3 + q_1 q_2)}{1 - 2(q_2^2 + q_3^2)}$$

结合已知的地面坐标系下的速度 $\mathbf{v}_g$ ，可以得到机体坐标系下的速度：

$${}^b\mathbf{v}_g = \begin{bmatrix} v_{gx_b} \\ v_{gy_b} \\ v_{gz_b} \end{bmatrix} = \begin{bmatrix} \cos \theta \cos \psi & \cos \theta \sin \psi & -\sin \theta \\ \sin \theta \cos \psi \sin \phi - \sin \psi \cos \phi & \sin \theta \sin \psi \sin \phi + \cos \psi \cos \phi & \cos \theta \sin \phi \\ \sin \theta \cos \psi \cos \phi + \sin \psi \sin \phi & \sin \theta \sin \psi \cos \phi - \cos \psi \sin \phi & \cos \theta \cos \phi \end{bmatrix} \mathbf{v}_g$$

步骤四：气动角获取

若忽略风速 v_w 的影响，进一步，由式

$$\alpha = \arctan \left(\frac{v_{azb}}{v_{axb}} \right)$$
$$\beta = \arcsin \left(\frac{v_{ayb}}{\sqrt{v_{axb}^2 + v_{ayb}^2 + v_{azb}^2}} \right)$$

可以得到迎角 α 和侧滑角 β 。

步骤五：参数辨识方法

在获得了辨识式左侧的滚转力矩 l 和右侧的状态量 $\rho, V_a, \beta, \omega_{xb}, \omega_{zb}, \delta_a, \delta_r$ 以及几何参数 S, b 之后，可以使用 MATLAB 中的 “regress” 函数和 “stepwise” 函数进行参数辨识。

最小二乘法

逐步回归法

步骤一：设置激励信号

对于激励信号设计，保证无人机以待辨识通道运动为主，以此来提高辨识精确度。在“SmallFixedWingUAVup_x0002_per.slx”的“Command”路径下中，可以使用“Signal Builder（RollChannel）”模块向滚转通道注入激励。

为了防止其它通道对滚转通道辨识效果产生影响，需要保持其它通道不变，仅给滚转通道注入激励信号，注入的激励信号如图所示。

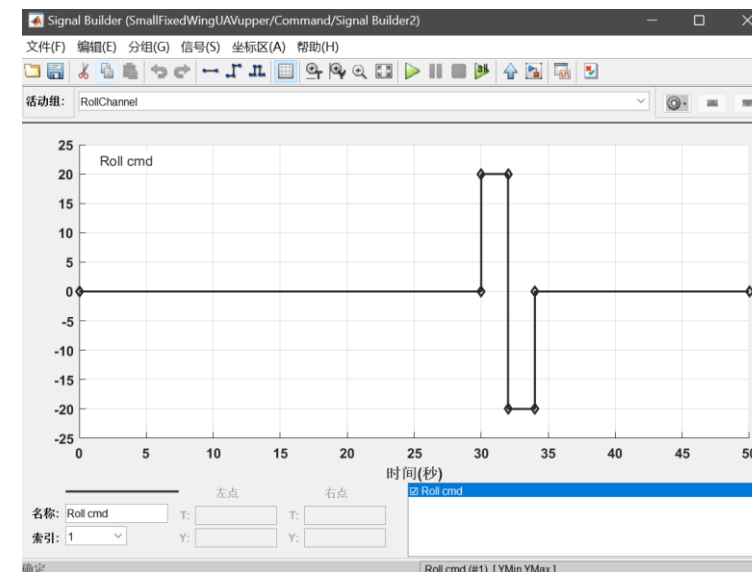


图. 在“Signal Builder”中更改滚转力矩通道激励信号

步骤二：运行 Simulink 软件仿真，完成辨识所需的作动并记录相应数据

打开“model”文件夹，首先运行初始化文件“InitDatactrl.m”，然后运行“SmallFixedWingUAVupper.slx”程序。该程序运行完成后可以自动导出滚转力矩通道辨识所需的数据，并保存为原始数据文件“Cl_Data.mat”。

步骤三：对各个状态量数据进行滤波等预处理

与分析实验相同，选择窗口长度为 15 的滑动平均滤波器对数据进行处理。

```
1 clear;clc;
2 data = load("C1_Data.mat");
3 % split data
4 beta = data.data(2,:);
5 dA = data.data(3,:);
6 dR = data.data(4,:);
7 Airspeed = data.data(5,:);
8 wbx = data.data(6,:);
9 wbx_dot = data.data(7,:);
10 wbx_dot = data.data(8,:);
11 % Average sliding filter
12 window = 15;
13 data_filtered.beta=movmean(beta>window);
14 data_filtered.dA=movmean(dA>window);
15 data_filtered.dR=movmean(dR>window);
16 data_filtered.Airspeed=movmean(Airspeed>window);
17 data_filtered.wbx=movmean(wbx>window);
18 data_filtered.wbx_dot=movmean(wbx_dot>window);
19 data_filtered.wbx_dot=movmean(wbx_dot>window);
```

清空工作区

从“.mat”文件中读取数据并存储为“data”

将“data”拆分为辨识需要的侧滑角、副翼舵偏角、方向舵舵偏角、空速、滚转角速率、偏航角速率以及滚转角加速度数据

对各个状态数据进行窗口长度为 15 的滑动平均滤波处理

滚转力矩通道数据预处理代码，来源于“e3/e3-3/C1_DataPreprocess.mlx”

步骤四：使用最小二乘法和逐步回归法辨识气动参数

选取无人机进行平稳飞行后的数据进行辨识。

- 基于固定翼姿态动力学方程，计算出实际滚转力矩系数力矩系数。
- 通过 MATLAB 自带的“regress”和“stepwise”函数进行最小二乘法估计和逐步回归，得到滚转通道气动参数。

```
1 clear;
2 load('Cl_DataFiltered.mat'); %load data
3 % split data
4 wbx=data_filtered.wbx(5500:7500)';
5 wbx_dot=data_filtered.wbx_dot(5500:7500)';
6 Va=data_filtered.Airspeed(5500:7500)';
7 dA=data_filtered.dA(5500:7500)';
8 dR=data_filtered.dR(5500:7500)';
9 beta=data_filtered.beta(5500:7500)';
10 wbz=data_filtered.wbz(5500:7500)';
11 % Aircraft geometric parameters
12 Ix=0.0894;
13 c=0.3;
14 b=1.2;
15 S=0.32;
16 rho = 1.225;
```

清空工作区

从“.mat”文件中读取数据并存储为“data”

将“data”拆分为滚转角速率、滚转角加速度、空速、副翼舵偏角、方向舵舵偏角、侧滑角、偏航角速率

输入了该通道辨识所需的无人机转动惯量以及几何参数等

滚转力矩通道数据预处理代码，来源于“e3/e3-3/Cl_DataPreprocess.mlx”

步骤四：使用最小二乘法和逐步回归法辨识气动参数

```
17 qbar=zeros(2001,1); %calculate qbar Cl wbx_compensation wbz_compensation
18 Cl=zeros(2001,1);
19 dalpha_after=zeros(2001,1);
20 q_after=zeros(2001,1);
21 wbx_compensation=zeros(2001,1);
22 wbz_compensation=zeros(2001,1);
23 for i=1:2001
24     qbar(i)=0.5*rho*Va(i)^2;
25     Cl(i)=(Ix*wbx_dot(i))/(qbar(i)*b*S);
26     wbx_compensation(i)=0.5*wbx(i)*b/(Va(i));
27     wbz_compensation(i)=0.5*wbz(i)*b/(Va(i));
28 end
29 % regress
30 Cl0=ones(2001,1);
31 XC1=[Cl0,beta,wbx_compensation,wbz_compensation,dA,dR];
32 [bCl,bintCl,rCl,rintCl,statsCl]=regress(Cl,XC1);
33 % stepwise
34 stepwise(XC1,Cl);
35 Cl_identify=XC1*bCl;
```

计算动压，根据固定翼姿态动力学方程计算滚转力矩系数，以及对滚转角速率和偏航角速率进行修正

生成零升力矩项

将用于辨识的状态量数据合并成矩阵

使用“regress”函数进行辨识

使用“stepwise”函数进行辨识

为使用辨识出的气动参数计算俯仰力矩系数

滚转力矩通道数据预处理代码，来源于“e3/e3-3/Cl_DataPreprocess.mlx”

步骤五：验证辨识结果

- 滚转力矩通道辨识结果如右上图： R^2 达到 0.998，回归效果较好。
- 如下表所示，辨识出的气动参数与真值进行对比，辨识结果与真值基本吻合。

图. 滚转力矩通道辨识结果与真值对比

气动参数	C_{l_0}	C_{l_β}	C_{l_p}	C_{l_r}	$C_{l_{\delta a}}$	$C_{l_{\delta r}}$
真值	0	-0.04	-0.414	0.399	0.0677	0.0168
辨识结果	0.00001	-0.0297304	-0.350213	0.561094	0.0675133	0.0262252

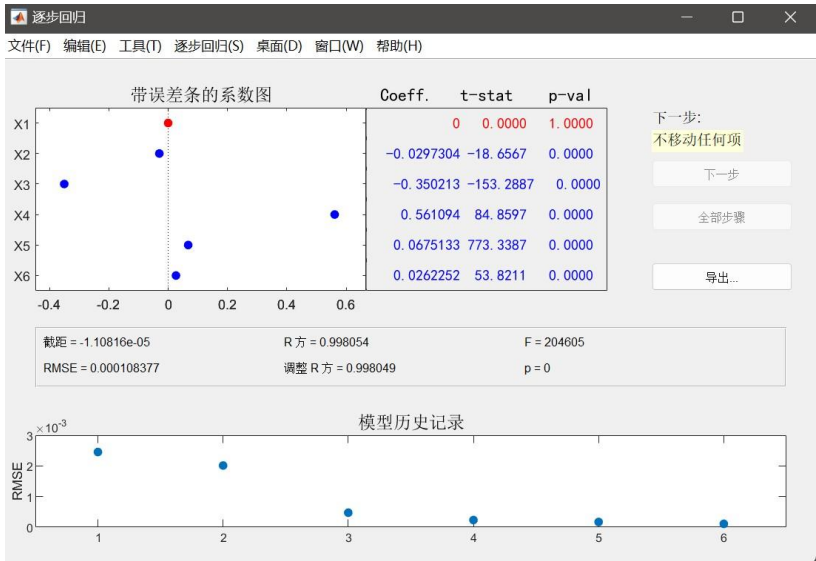


图. 滚转力矩通道逐步回归

- 如右下图所示, 对比实际的滚转力矩导数和辨识结果拟合出的滚转力矩导数, 二者基本重合, 证明辨识精度较高。

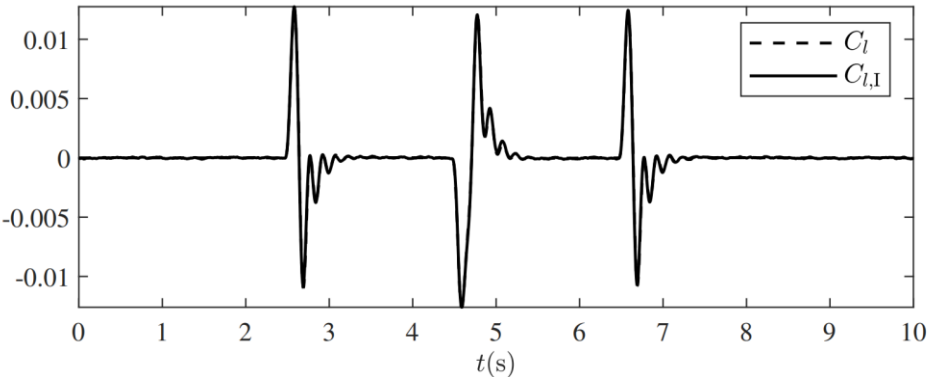
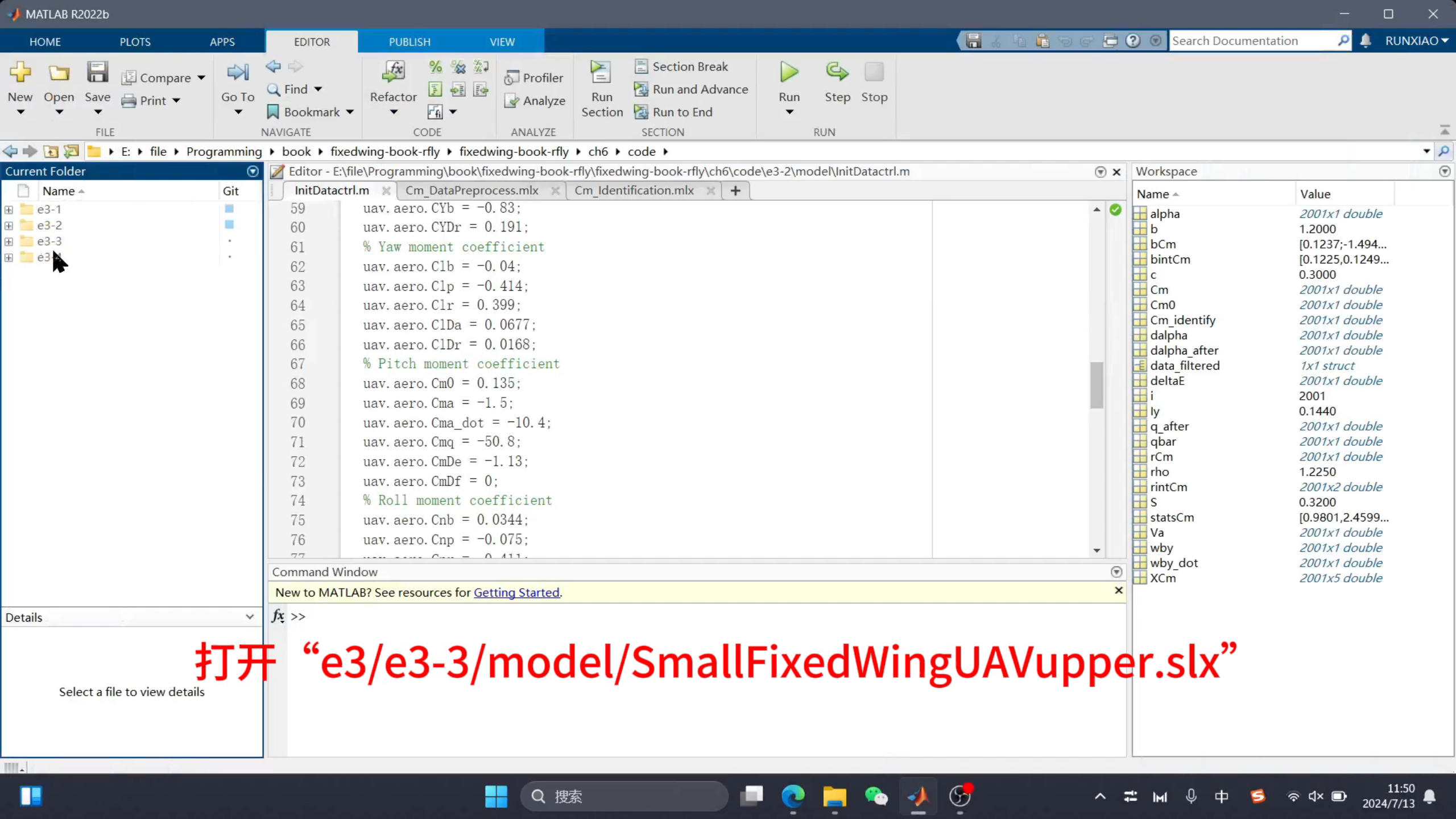


图. 滚转力矩通道拟合效果



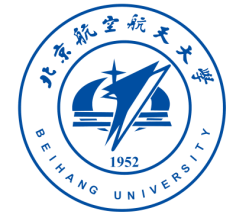
打开“e3/e3-3/model/SmallFixedWingUAVupper.slx”

大纲



1. 实验原理
2. 基础实验
3. 分析实验
4. 设计实验
5. 硬件在环仿真实验
6. 本章小结

硬件在环仿真实验-制导模型辨识目标



实验目标

已知

(1)软件：MATLAB R2022b 及以上版本，RflySim 仿真平台。

(2)硬件：实验指导包“e3/e3-4/GuidanceModel_Identification”。实验指导包中有高度通道实验代码包“HeightChannel”文件夹，该文件夹中主要有：高度通道辨识轨迹控制脚本

“HeightChannel_OffboardController.py”，从自驾仪硬件中导出的“log.ulg”，经过“pyulog”功能包转码得到的“log_vehicle_local_position_0.csv”和“log_vehicle_local_position_setpoint_0.csv”，从“csv”文件导出的“HeightChannel_Input.mat”和“HeightChannel_Output.mat”，对输入输出数据进行预处理的脚本“HeightChannel_DataPreprocess.mlx”以及完成辨识的脚本“HeightChannel_Identification.sid”。

(3)硬件：CubePilot/Pixhawk 自驾仪。

目标

在硬件在环仿真环境下，完成固定翼无人机的制导模型系统辨识，该实验与基础实验类似，区别在于需要使用 CubePilot/Pixhawk 自驾仪进行实验得到所需的数据。

实验步骤

步骤一：搭建“Path Follow”模块

以高度通道硬件在环实验为例进行说明。使用“航点 + 空速”的指令接口对固定翼无人机进行控制，可以设定路径点参考速度为 40m/s，固定翼在横侧向上不作动，保持直线前飞，而在高度通道进行偶极方波式的运动。

```
1  v = 40 # set refrence airspeed for mission points
2  for i in range(41): # mission point set
3      x = -v*i
4      y = 0
5      if i <= 5:
6          z = -100
7      elif 5 < i <= 21:
8          z = -120
9      elif 21 < i <= 41:
10         z = -80
11     else:
12         z = -100
13     missionPoints.append([x, y, z])
```

——> 设定路径点参考速度

——> 设定北向航点

——> 设定东向航点

}——> 设定第 0-5 个航点高度为 100m

}——> 设定第 6-21 个航点高度为 120m

}——> 设定第 22-41 个航点高度为 80m

}——> 设定其余航点高度为 100m

——> 在每次循环中用“append”函数将三维航点行向量按行拼接，得到航点列表

构造循环生成航点队列

高度通道轨迹设计，来源于“e3/e3-4/GuidanceModel_Identification/HeightChannel_OffboardController.py”

硬件在环仿真实验-制导模型辨识目标



实验步骤

步骤二：运行硬件在环仿真，完成辨识所需作动

如左下图，在电脑上接入 Cubepilot/CubePilot/Pixhawk 自驾仪，打开实验包内的“HeightChannel”文件夹，运行“AircraftMathworksMavlinkHITLRun.bat”文件，弹出右下图界面，在终端窗口输入合适的串口号，按下回车键，一键启动仿真平台。

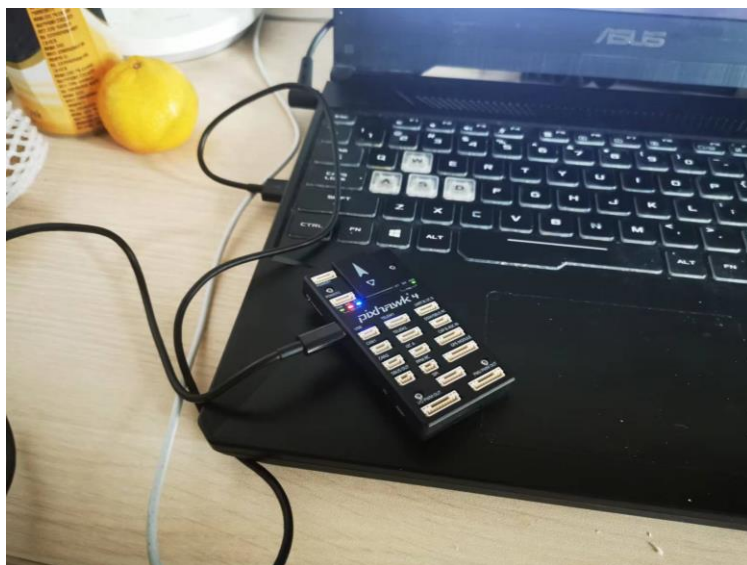


图. 硬件连接

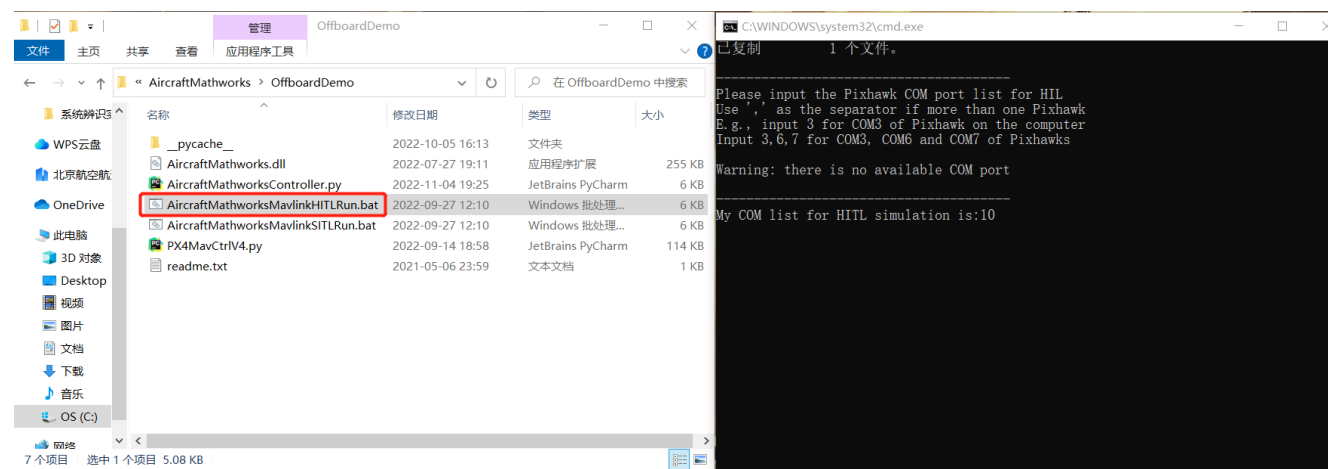


图. 平台设置

硬件在环仿真实验-制导模型辨识目标



实验步骤

步骤二：使用 MATLAB 系统辨识工具箱进行系统辨识

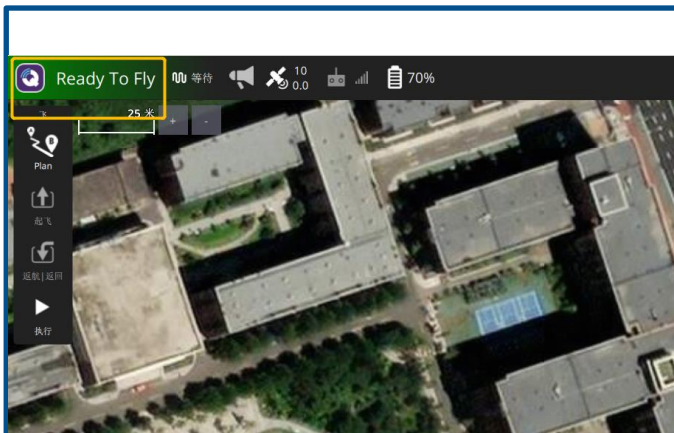


图. 准备起飞

切换到 QGC 控制台界面，当图中左上角无人机处于“Ready to fly”状态后，运行提前设计好的轨迹控制程序“HeightChannel_OffboardController.py”。

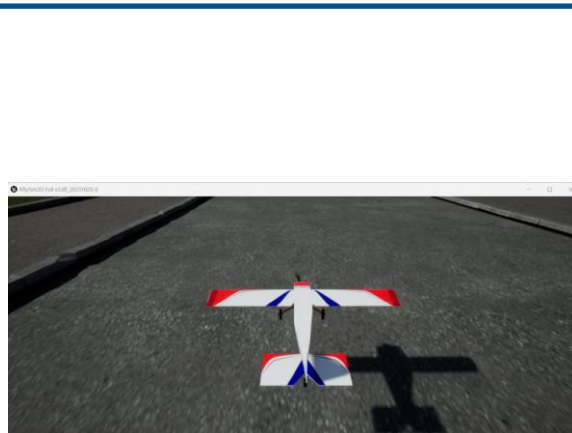


图. 飞行运动

无人机将接收 Python 终端发送的航点指令飞行，可以通过 RflySim3D 观察无人机运动。

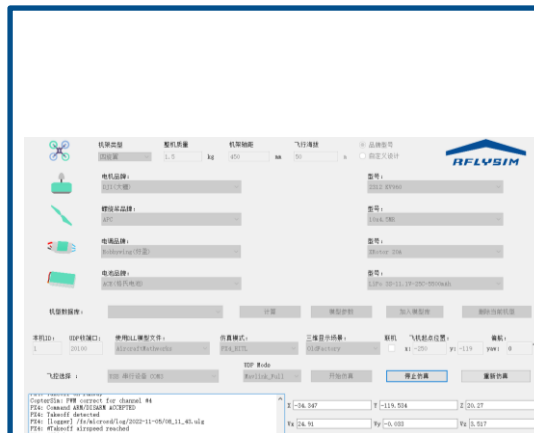


图. 飞行状态

通过 CopterSim 观测无人机部分实时状态。

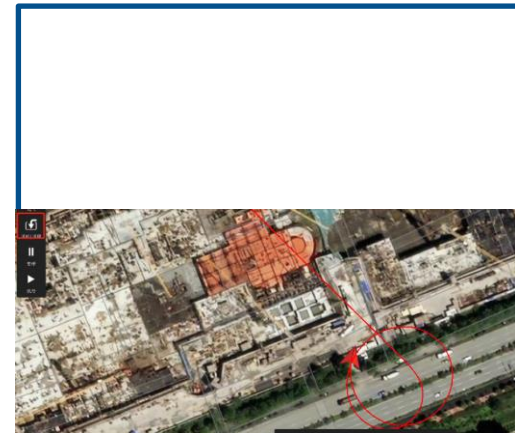


图. 返航

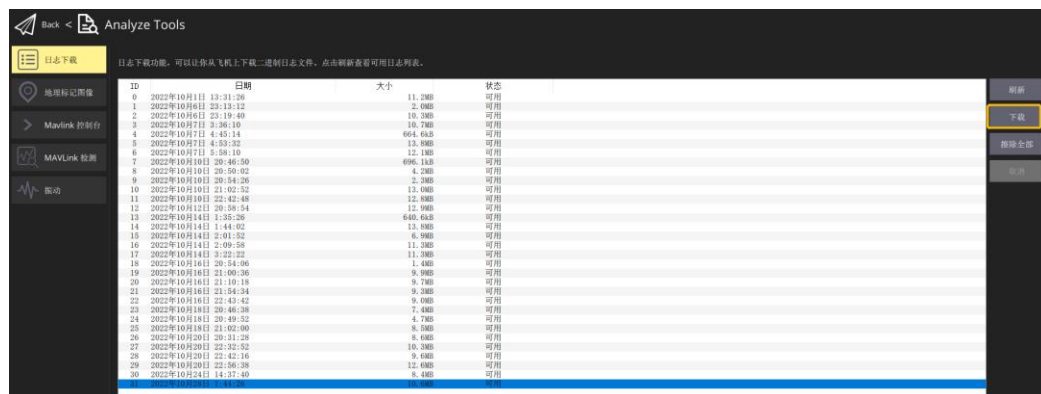
当无人机飞行到最后一个航迹点位置时，将进行原地盘旋运动。此时，运行 QGC 平台的返航程序即可实现无人机返航，结束飞行运动。

硬件在环仿真实验-制导模型辨识目标



实验步骤

步骤三：从自驾仪中导出飞行日志，进行数据预处理

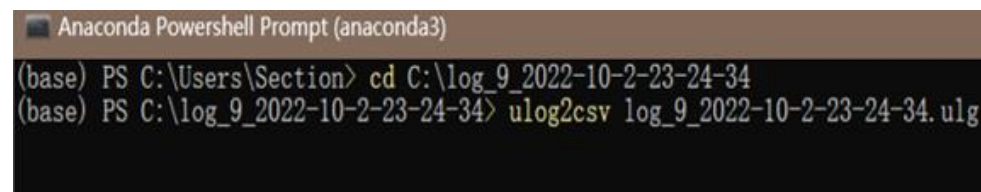


The screenshot shows the 'Log Download' interface in QGC's Analyze Tools. It features a table with columns for ID, Date, Size, and Status. The table lists 30 log files from 2022-10-11 to 2022-10-23. A 'Download' button is highlighted in the top right corner of the table area.

ID	日期	大小	状态
0	2022年10月11日 13:31:26	11.2MB	可用
1	2022年10月11日 23:13:12	2.0MB	可用
2	2022年10月11日 23:19:40	10.3MB	可用
3	2022年10月11日 0:36:10	10.7MB	可用
4	2022年10月11日 4:45:14	664.6KB	可用
5	2022年10月11日 4:53:32	13.8MB	可用
6	2022年10月11日 5:36:10	12.1MB	可用
7	2022年10月10日 20:56:50	696.1KB	可用
8	2022年10月10日 20:50:02	4.2MB	可用
9	2022年10月10日 20:54:06	2.3MB	可用
10	2022年10月10日 21:02:52	13.0MB	可用
11	2022年10月10日 22:42:48	12.9MB	可用
12	2022年10月12日 20:58:54	12.9MB	可用
13	2022年10月14日 1:35:36	640.6KB	可用
14	2022年10月14日 1:44:02	13.8MB	可用
15	2022年10月14日 2:01:52	6.9MB	可用
16	2022年10月14日 2:09:58	11.3MB	可用
17	2022年10月14日 3:22:22	11.3MB	可用
18	2022年10月14日 20:54:06	1.4MB	可用
19	2022年10月16日 21:00:36	9.9MB	可用
20	2022年10月16日 21:10:18	9.7MB	可用
21	2022年10月16日 21:54:54	9.3MB	可用
22	2022年10月16日 22:43:42	9.0MB	可用
23	2022年10月18日 20:46:38	7.4MB	可用
24	2022年10月18日 20:49:32	4.7MB	可用
25	2022年10月18日 21:02:00	8.3MB	可用
26	2022年10月20日 20:51:28	8.6MB	可用
27	2022年10月20日 22:32:52	10.3MB	可用
28	2022年10月20日 22:42:16	9.4MB	可用
29	2022年10月20日 22:56:38	12.0MB	可用
30	2022年10月24日 14:27:40	9.4MB	可用

图. 数据下载

当无人机稳定降落后，点击 QGC 平台左上角图标，选择“Analyze Tools”，在“Log Download”界面可选择最近一次飞行的数据进行下载。



```
Anaconda Powershell Prompt (anaconda3)
(base) PS C:\Users\Section> cd C:\log_9_2022-10-2-23-24-34
(base) PS C:\log_9_2022-10-2-23-24-34> ulog2csv log_9_2022-10-2-23-24-34.ulg
```

图. pyulog使用方法

使用“pip install”指令在 Python 环境下安装“pyulog”功能包，使用“ulog2csv”指令将“ulog”文件转化为可以导入到 MATLAB 中的“csv”格式。

实验步骤

步骤四：使用 MATLAB 系统辨识工具箱完成制导模型辨识

与基础实验相同，使用 MATLAB 系统辨识工具箱完成高度通道制导模型辨识。辨识效果如图1.33，匹配度达到了 84.18%，符合预期。

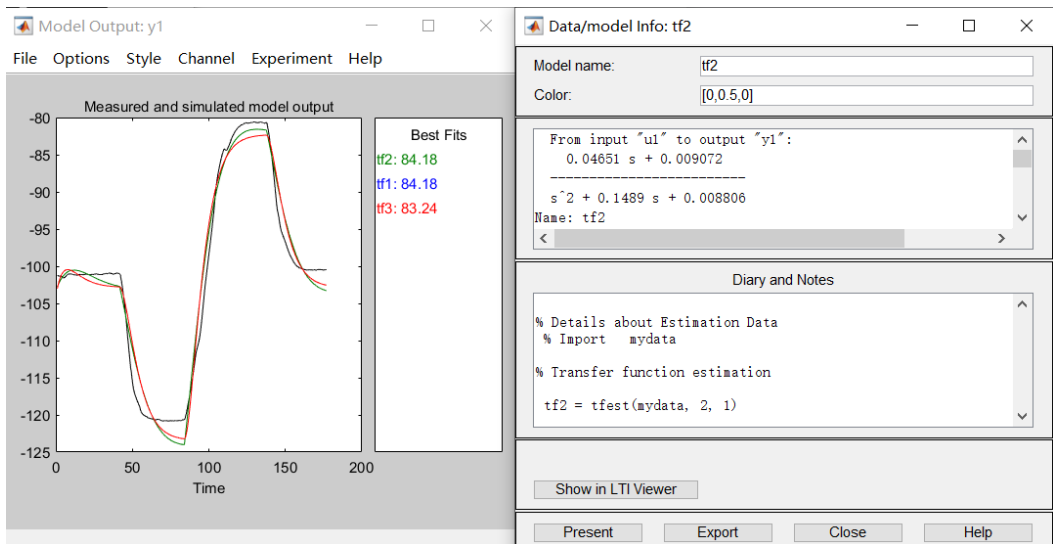


图. 高度通道辨识

高度通道的制导模型为：

$$G_H(s) = \frac{H(s)}{H_d(s)} = \frac{0.04651s + 0.009072}{s^2 + 0.1489s + 0.008806}$$

其余通道辨识的流程与高度通道类似，区别在于设计的航迹不同、选取的输入输出数据不同，可以参考实验包内提供的代码进行实践。

实验目标

已知

(1)软件：MATLAB R2022b 及以上版本，RflySim 仿真平台。

(2)硬件：实验指导包“e3/e3-4/AerodynamicParameter_Identification”实验包。实验指导包中主要有：固定翼无人机硬件在环实验控制脚本“AircraftMathworksAttCtrl.py”，待辨识的固定翼无人机模型 dll 文件“Small_x0002_FixedWingUAVIdentificationHIL.dll”，硬件在环实验一键启动脚本

“UAVIdentification.bat”，飞行日志数据文件夹“log”，其中主要包括从自驾仪硬件中导出的飞行日志文件“log.ulog”和相应的 csv 文件；从 csv 文件中导入数据并配准的自动化脚本

“load_manipulate_data.m”以及该脚本导出的数据文件“Data.mat”，输入输出数据预处理脚本

“Cl_DataPreprocess.mlx”以及该脚本导出的处理后的数据文件“Cm_DataFiltered.mat”，滚转力矩通道气动参数辨识脚本“Cl_Identification.mlx”。

(3)硬件：CubePilot/Pixhawk 自驾仪。

目标

在在硬件在环仿真环境下，完成固定翼无人机的滚转力矩通道的气动参数辨识。与分析实验和设计实验类似，区别在于需要使用 CubePilot/Pixhawk 自驾仪进行实验得到所需的数据。

实验步骤

步骤一：设计固定翼无人机滚转通道激励

与制导模型辨识硬件在环实验不同，使用“姿态 + 油门”的指令接口对固定翼无人机进行控制，可以设定固定翼无人机油门固定为 0.8，在纵向上不作动；而在滚转通道进行偶极方波式的运动，幅值为 30 度，持续时间为 20 秒。

1	<code>mav1.SendAttPX4([30,0,0],0.8)</code>	} → 设定固定翼无人机的滚转角为 30 度，油门为 0.8，并持续 10s
2	<code>time.sleep(10)</code>	
3	<code>mav1.SendAttPX4([-30,0,0],0.8)</code>	} → 设定固定翼无人机的滚转角为-30 度，油门为 0.8，并持续 10s
4	<code>time.sleep(10)</code>	
5	<code>print("arrive")</code>	→ 为打印标志行，说明本周期运动结束

滚转通道轨迹设计，来源于“e3/e3-4/AerodynamicParameter_Identification/AircraftMathworksAttCtrl.py”

实验步骤

步骤二：运行硬件在环仿真，完成辨识所需动作

与制导模型辨识硬件在环实验类似，运行硬件在环仿真，可以得到相应的飞行日志文件。

- (1) 将待辨识的固定翼无人机模型dll文件“SmallFixedWingUAVIdentificationHIL.dll”拷贝到“PX4PSP/CopterSim/external/model”路径下。
- (2) 将 CubePilot/Pixhawk 自驾仪连接到电脑上。
- (3) 运行硬件在环实验一键启动脚本“UAVI_x0002_identification.bat”并选择合适的端口启动仿真。
- (4) 切换到 QGC 界面，无人机处于“Ready to fly”状态后，运行控制脚本“AircraftMathworksAttCtrl.py”。
- (5) 在控制脚本作用下，固定翼无人机会循环执行设定的姿态激励动作。进行 3-5 次激励动作后，手动关闭脚本，无人机会进入盘旋状态。
- (6) 运行 QGC 平台的返航程序即可实现无人机返航（具体操作参考制导模型辨识硬件在环实验）。

实验步骤

步骤三：从自驾仪中导出飞行日志，进行数据预处理

(1) 与制导模型辨识硬件在环实验相同，从 QGC 下载最新的飞行日志，将其存入到“log”文件夹下。在“log”文件夹下使用“pyulog”工具包中的“ulog2csv”命令将日志文件转化为 csv 文件。

- “vehicle_local_position_0.csv”：获取地面坐标系下的无人机的位置、速度和加速度；
- “vehi_x0002_cle_attitude_0.csv”：获取四元数表示的无人机姿态；
- “vehicle_angular_velocity_0.csv”：获取无人机的角速率；
- “airspeed_0.csv”：获取无人机的空速；
- “vehicle_air_data_0.csv”：获取无人机的动压和空气密度；
- “actuator_controls_0_0.csv”：获取无人机作动器的控制信号。

可以使用“load_manipulate_data.m”脚本自动从“log”文件夹下导入 csv 文件，并自动进行时间配准处理

。

实验步骤

步骤三：从自驾仪中导出飞行日志，进行数据预处理

(2) 时间配准处理后，需要根据已知的状态数据（地面坐标系下的速度、姿态），推算出参数辨识需要的迎角和侧滑角数据，推算方法与分析实验的实验设计中的描述一致。

在硬件在环实验中，可以忽略风的影响。估计迎角和侧滑角，可使用式子：

$$V_a = \sqrt{v_{ax_b}^2 + v_{ay_b}^2 + v_{az_b}^2}$$

$$\alpha = \arctan\left(\frac{v_{az_b}}{v_{ax_b}}\right)$$

$$\beta = \arcsin\left(\frac{v_{ay_b}}{\sqrt{v_{ax_b}^2 + v_{ay_b}^2 + v_{az_b}^2}}\right)$$

相应代码在“Cl_DataPreprocess.mlx”中，关键代码如代码块所示。

```
1 alpha(i) = atan(vzb(i)/vxb(i)); % calculate alpha
2 beta(i) = asin(vyb(i)/sqrt(vxb(i)^2+vyb(i)^2+vzb(i)^2)); % calculate beta
```

} → 估计迎角 α 和侧滑角 β

迎角侧滑角估计，来源于“e3/e3-4/AerodynamicParameter_Identification/Cl_DataPreprocess.mlx”

实验步骤

步骤四：对各个状态量进行滤波处理

同分析与设计实验，利用 MATLAB 自带的“movmean”函数，选择合适的窗口长度即可实现高频噪声滤除（硬件在环实验中注入的噪声较小，因此可以选择较小的窗口长度）。具体只需要运行“Cl_DataPreprocess.mlx”即可得到滤波后的数据“Cm_DataFiltered.mat”。

步骤五：使用最小二乘法和逐步回归法辨识气动参数并验证结果

与设计实验类似，俯仰通道辨识代码如代码段所示。

```
1 clear;  
2 load('Cl_DataFiltered.mat');% load dataa  
3 startpoint = 7000;% set intercept point  
4 endpoint = 21000;
```

→ 清空工作区

→ 从“.mat”文件中读取数据并存储为“data”

} → 设置截取数据起始点、结束点

俯仰力矩通道辨识，来源于“e3/e3-4/AerodynamicParameter_Identification/Cl_Identification.mlx”

实验步骤

步骤五：使用最小二乘法和逐步回归法辨识气动参数并验证结果

俯仰力矩通道辨识，来源于“e3/e3-4/AerodynamicParameter_Identification/Cl_Identification.mlx”

将“data”
拆分成各
变量数据

输入该通
道辨识所
需的无人
机转动惯
量以及几
何参数等

```
5 % split data
6 wbx=data_filtered.wbx(startpoint:endpoint)';
7 wbx_dot=data_filtered.wbx_dot(startpoint:endpoint)';
8 Va=data_filtered.Airspeed(startpoint:endpoint)';
9 dA=data_filtered.dA(startpoint:endpoint)';
10 dR=data_filtered.dR(startpoint:endpoint)';
11 beta=data_filtered.beta(startpoint:endpoint)';
12 wbz=data_filtered.wbz(startpoint:endpoint)';
13 % Aircraft geometric parameters
14 Ix=0.0894;
15 c=0.3;
16 b=1.2;
17 S=0.32;
18 rho = 1.15;
```

调用“regress”
函数进行辨识

```
19 qbar=zeros(length(Va),1);
20 Cl=zeros(length(Va),1);
21 wbx_compensation=zeros(length(Va),1);
22 wbz_compensation=zeros(length(Va),1);
23 for i=1:length(Va)
24     qbar(i)=0.5*rho*Va(i)^2;
25     Cl(i)=(Ix*wbx_dot(i))/(qbar(i)*b*S);
26     wbx_compensation(i)=0.5*wbx(i)*b/(Va(i));
27     wbz_compensation(i)=0.5*wbz(i)*b/(Va(i));
28 end
29 % regress
30 Cl0=ones(length(Va),1);
31 XCl=[Cl0,beta',wbx_compensation,wbz_compensation,dA',dR'];
32 [bCl,bintCl,rCl,rintCl,statsCl]=regress(Cl,XCl);
33 % stepwise
34 stepwise(XCl,Cl);
35 Cl_identify=XCl*bCl;
```

为计算动压，
根据固定翼姿
态动力学方程
计算滚转力矩
系数，以及对
滚转角速率和
偏航角速率进
行修正

使用“stepwise”函数进行辨识
为使用辨识出的气动参数计
算俯仰力矩系数

实验步骤

步骤五：使用最小二乘法和逐步回归法辨识气动参数并验证结果

辨识效果如左图， R^2 达到了 0.55，基本符合预期，拟合效果如下图。

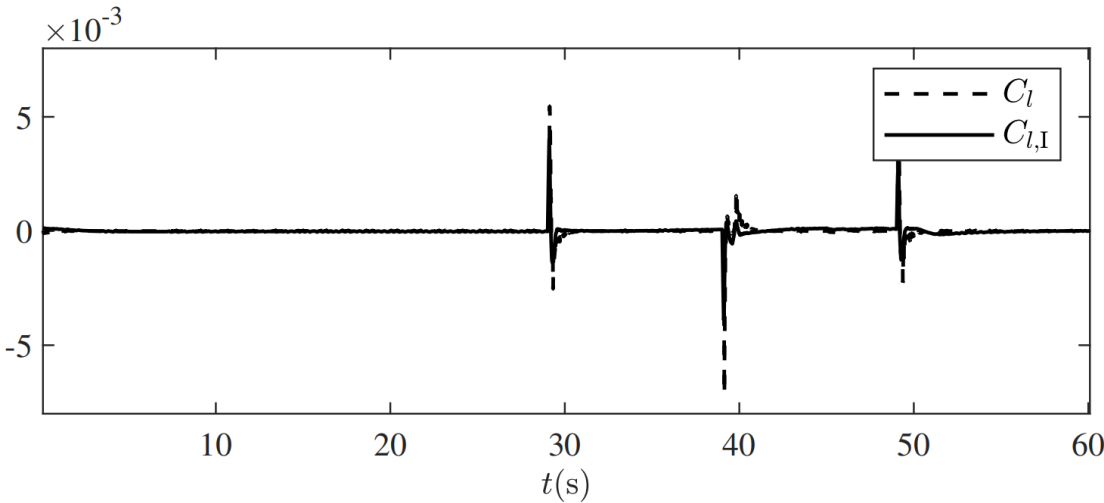


图. 滚转力矩通道硬件在环拟合效果

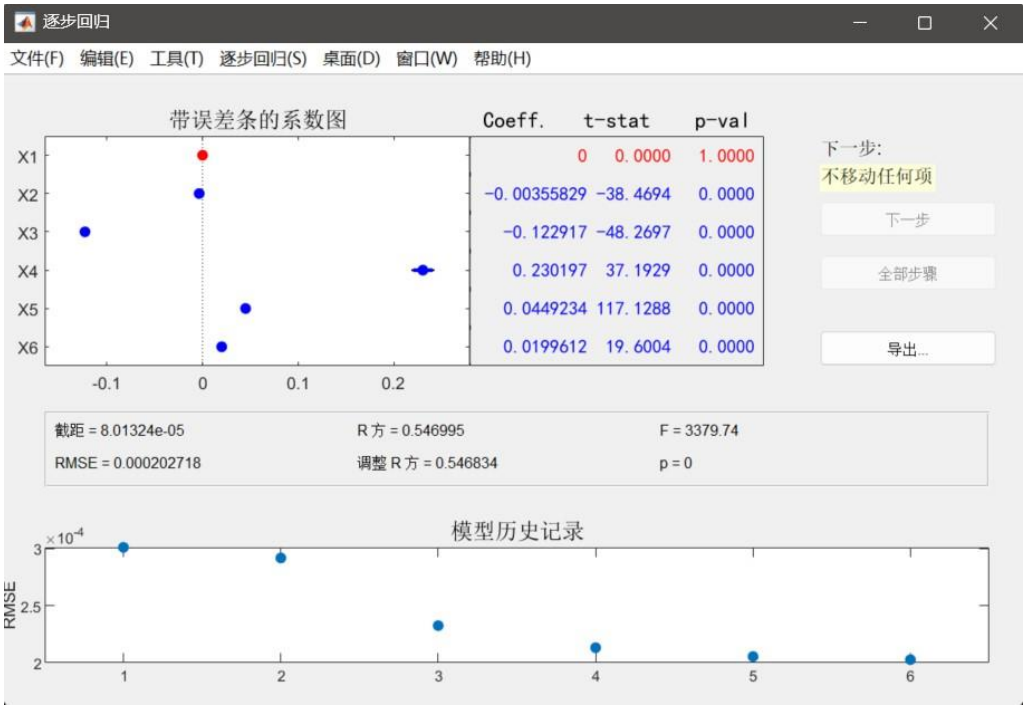


图. 滚转力矩通道硬件在环逐步回归辨识结果

讨论

(1) 辨识结果与真值对比如表所示。可见气动导数 C_{l_β} 以外，辨识结果与真值数量级相同，数值相差不大。

气动参数	C_{l_0}	C_{l_β}	C_{l_p}	C_{l_r}	$C_{l_{\delta_a}}$	$C_{l_{\delta_r}}$
真值	0	-0.04	-0.414	0.399	0.0677	0.0168
辨识结果	0.00008	-0.00355829	-0.122917	0.230197	0.0449234	0.0199612

表. 滚转力矩通道辨识结果与真值对比

(2) 在硬件在环实验中 C_{l_β} 辨识结果不太准确的原因在于：

- 侧滑角不是通过传感器测量得到，而是通过地面系下的速度与姿态解算机体系下的速度，进而再根据机体系下的速度和空速来估计迎角和侧滑角，这两步估计导致硬件在环实验中 C_{l_β} 辨识的结果较大误差。
- 由于该项相关的状态量和气动导数乘积的数量级较小，在小机动范围内对模型的影响不大。

(3) 从 R^2 和滚转力矩系数拟合效果来看，辨识结果总体是较为准确的，可以较好的描述滚转力矩通道的特性。从辨识出的参数得到的传递函数可以满足后续控制器设计的要求。



首先进行制导模型高度通道辨识硬件在环仿真



大纲



1. 实验原理
2. 基础实验
3. 分析实验
4. 设计实验
5. 硬件在环仿真实验
6. 本章小结

- (1) 通过基础实验，了解固定翼无人机制导模型，掌握数据时间配准预处理方法，使用 MATLAB 系统辨识工具箱完成固定翼制导模型的辨识。
- (2) 通过分析实验，以俯仰力矩通道为例，辨识式中的的气动参数，了解气动参数辨识的流程和思想。
- (3) 在设计实验中，在固定翼无人机 Simulink 模型中，设计合适的输入信号，激励并采集数据；辨识滚转力矩通道的气动导数；验证得到的非线性模型的吻合程度。
- (4) 通过硬件在环仿真实验，使用 CubePilot/Pixhawk 自驾仪进行实验得到所需的数据，完成固定翼无人机的制导模型系统辨识。
- (5) 通过以上实验，让读者较为深入掌握系统辨识的方法。

如有疑问，请到 <https://rflysim.com/course> 查询更多信息。

致谢



□ 感谢刘润潇为本讲PPT准备做出的贡献



□ 更多信息请访问公众号和网站



rfly.buaa.edu.cn