系统建模与仿真实验指导书

直流伺服系统

(V3.0版)

力矩环系统建模及稳定性分析实验

(一) 实验目的

- 1. 了解机理法建模;
- 2. 掌握控制系统稳定性分析的基本方法;
- (二) 实验设备
- 1. GSMT2014 型直流伺服系统控制平台; 直流伺服系统电控箱; PC (MATLAB 平台)
- (三) 实验原理

系统建模可以分为两种:机理建模和实验建模。机理建模是在了解研究对象的运动规律基础上,通过 物理、化学的知识和数学手段建立起系统内部的输入——输出状态关系。实验建模是通过在研究对象上加 上一系列的研究者事先确定的输入信号,激励研究对象并通过传感器检测其可观测的输出,应用数学手段 建立起系统的输入——输出关系。这里面包括输入信号的设计选取,输出信号的精确检测,数学算法的研 究等内容。

- 1. 机理法建立直流伺服电动机过渡过程的数学模型
- 用下列诸式描述其动态过程:(忽略了系统摩擦阻力)
- 1) 直流伺服电机转矩方程:

$$T = J\frac{d\omega}{dt} + B\omega \tag{1-1}$$

2) 电磁转矩方程:

$$T_e = C_m I_d \tag{1-2}$$

由转矩平衡可得 $T = T_e$,即:

$$J\frac{d\omega}{dt} + B\omega = C_m I_d$$

等式两边进行拉普拉斯变换,即:

$$Js\omega(s) + B\omega(s) = C_m I_d(s)$$

整理可得:

$$\frac{\omega(s)}{I_d(s)} = \frac{C_m}{Js + B} \tag{1-3}$$

式(1-3)是以电枢电流 I_{a} 为输入,以角速度 ω 为输出时,直流电动机的传递函数.

参数列表:

参数	参数名称	值	单位
T _e	电磁转矩		
Т	电机转矩		
I _d	电枢电流		А
ω	电动机的角速度		rad/s

J	电机轴上的转动惯量	J	kgm ²
В	阻尼系数	$0.05 C_{_m}$	Nm/ (rad/s)
C_m	转矩常数	0.0644	Nm/A
ENC	电机编码器	4000	PPR

阻尼单位注释:由于阻尼力矩 $T_b = B\omega = k\omega C_m$,由于 T_b 单位为Nm, ω 单位为rad/s,所以B单位为Nm/rad/s

直流电机模型为:公式(1-3)

$$\frac{\omega(s)}{I_d} = \frac{C_m}{J^* s + B}$$

带入 $B \approx B_1 = k^* C_m$, (此处k=0.05, 经验值)可得:

$$\frac{\omega(s)}{I_d} = \frac{\frac{1}{k}}{\frac{J}{k*C_m}*s+1}$$

系统参数为:

 $C_m = 0.0644$ $J = 0.00029 kgm^2$ $B = 0.05 * C_m$

直流伺服系统数学模型为':

$$\frac{\omega(s)}{I_d} = \frac{20}{0.09s + 1} \tag{1-4}$$

2. 稳定性分析

机理法建模建出系统模型传递函数为:

$$\frac{\omega(s)}{I_d} = \frac{20}{0.09s+1}$$

由此可以看出,直流伺服电机转速与电流的传递函数是个一阶系统,有一个极点(-11.1),在 s 的左 半平面,所以系统是稳定的。可以通过 MATLAB 中 Simulink 中仿真验证,也可以连接直流伺服系统进行 实时控制验证。

(四) 实验步骤

1. 机理法模型稳定性分析实验

注意**:**

1) 运行 MATLAB 程序前需把两个电机驱动器模式改为力矩环运行。大带轮通过同步带同时连



1) 建立系统的 MATLAB 仿真模型,参考程序如图 2.1.1(并保存名为"ModleTest_sim",保存默认格式为"slx"



图 2.1.1 系统开环仿真结构图

2) 设置输入阶跃信号为 1,即双击 Step 模块设置 Final value 为 1。

Step Output a step. Parameters Step time: 0 Initial value: 0 Final value: 1 Sample time: 0 V Interpret vector parameters as 1-D V Enable zero-crossing detection
Output a step. Parameters Step time: O Initial value: O Final value: 1 Sample time: O Interpret vector parameters as 1-D I Enable zero-crossing detection
Parameters Step time: 0 Initial value: 0 Final value: 1 Sample time: 0 Interpret vector parameters as 1-D V Enable zero-crossing detection
Step time: Q Initial value: Q Final value: 1 Sample time: Q Interpret vector parameters as 1-D V Enable zero-crossing detection
O Initial value: 0 Final value: 1 Sample time: 0 ✓ Interpret vector parameters as 1-D ✓ Enable zero-crossing detection
Initial value: 0 Final value: 1 Sample time: 0 Interpret vector parameters as 1-D Interpret vector parameters as 1-D
0 Final value: 1 Sample time: 0 Interpret vector parameters as 1-D Interpret vector parameters as 1-D
Final value: 1 Sample time: 0 Interpret vector parameters as 1-D Interpret zero-crossing detection
1 Sample time: 0 Interpret vector parameters as 1-D Interpret zero-crossing detection
Sample time: 0 Interpret vector parameters as 1-D Interpret zero-crossing detection
0 ☑ Interpret vector parameters as 1-D ☑ Enable zero-crossing detection
 Interpret vector parameters as 1-D Enable zero-crossing detection
✓ Enable zero-crossing detection
OK Cancel Help Apply

3) 点击 🕑 运行,双击 Scope 模块,得到系统仿真曲线

K Scope		P					
File Tools View Simulation Help	File Tools View Simulation Help						
🎯 • 🍓 🕟 🕪 🎟 🏞 • 🕰 • 💭 • 🖨							
20			······································				
15							
5							
	2 4 5	C 7 0					
Ready	J 1 J	0 / 0	T=10.000				

4) 开环系统稳定,稳态值 20rad/s,超调量: $\sigma=0$,调节时间: $T_s=0.35s$

- 2. 实验法建模
- 1) 打开桌面程序 "DampingDemo.slx", 如图 2.1.2 控制架构。

Damping Demo



3) 双击"Step"模块,设置"Finalvalue"为1

🚡 Source Block Parameters: Step 🛛 🗙					
Step					
Output a step.					
Parameters					
Step time:					
0					
Initial value:					
0					
Final value:					
1					
Sample time:					
0					
✓ Interpret vector parameters as 1-D					
₩ Enable zero-crossing detection					
OK Cancel Help App	ly				

4) 双击"RealControl"模块,其中蓝色模块"GTSSetDAC"和"GTSGetPos"设置轴号为2,"GTSSetDAC1" 设置轴号为1;



- 5) "filter"模块为滤波模块,建议添加;
- 6) 双击 "Scope" 模块,设置 "Save data to workspace" 中 "Vaiable name" 为 info, "Format" 为 "Structure With Time";



- 7) 点击"່ "编译程序;
- 8) 点击" 🕑"运行程序,建议修改运行时间至少在 30~100 秒之间;
- 9) 程序运行后,在 workspace 中可以看到采集到的实验数据;

	2 C 🗗	Search	Documentat	ion 🔎 🗖
(a) Preference Layout → Parallel → ENVIRONMENT	Help	Request S	y Support	
				,
. 🔊	Workspa	се		\odot
	Name 🔺		Value	Min
	<mark>∃≣</mark> info		<1x1 struct:	>
	•		1	

10) 提取 info 中的数据,在 matlab 控制台输入:

input = info.signals.values(:, 1);

output = info.signals.values(:, 2);

注意:"Vaiable name",修改时,则将 info 改为自己修改的变量名。

建议:不同输入信号下,修改 intput 和 output 的名称,以免数据覆盖报错。

APPS				0 ¢ Ð	② Search Dod	umentation	<mark></mark>
Import Save Workspace Variable	Analyze Code	Simulink Library	() Preferences Layout ↓ Parallel ↓	s ? Help	Community Sequest Support Community Sequest Support Community Sequest Support Sequest Support	ort	
VARIABLE	CODE	SIMULINK	ENVIRONMENT		RESOURCES		
▶ paradox ▶ Desktop ▶ GMST2004-III	▶ Speed ▶						- P
Command Window			\odot	Workspa	ice		
>> input = info.signals.valu	es(:, 1);			Name 🔺	Val	ue	Min
>> output = info.signals.val	ues(:, 2);			🗄 info	<1)	1 struct>	
$f_{x} >>$				input	<61 t <61	190x1 double> 190x1 double>	1000 0
			Ī				

11) 在控制台输入"ident"打开系统辨识工具箱



12) 点击 "Import data",选择"Time domain data"导入数据, Input 设置 input, Output 设置为 output, Startingtime 设置为 0, Samplinginterval 设置为 0.001,最后点击"Import"导入系统辨识数据;
 ✓ Import Data

Data Format for Signals					
Time-Domain	I Signals				
Works	space Variable				
Input:	input				
Output:	output				
Data	Information				
Data name:	mydata				
Starting time	0				
Sampling interv	al: 0.001				
	More				
Import	Reset				
Close	Help				
System Identification Tool - Untitled					
File Options Window Help					
Operations					
< Preprocess					
mydata f					
mydata					
Working Data					
t l					
Estimate>					
Uata Views To To	Model Views				
	Model reside				
Erequency function					
	mydata Noise spectrum				
Trash Validation Data Data set mydata inserted. Double click on icon (right mouse) for text information.					

13) 点击 "Estimate-->" 选择 "TransferFunctionModels", 点击 "Estimate" 开始辨识一阶系统;

A Transfer Functions
Model name: tf2 🖉
Number of poles: 1 Number of zeros: 0
O Continuous-time ○ Discrete-time (Ts = 0.001) □ Feedthrough
 I/O Delay Estimation Options
Estimate Close Help

注意:如果辨识的拟合度低于 70%,建议重新采集数据,再进行辨识; 对于 step 信号,辨识的拟合度一般会很低,可能达不到 70%;但对于扫频信号和 M 序列信号,二阶传递

- 函数和多项式模型的拟合度会在80%以上。
- 14) 点击"Estimate-->"选择"TransferFunctionModels",并设置"Number of poles"为 2,设置"Number of zeros" 99 TT 4/5 合臣 > 日 -1 占十 "1 为

勺 1	l,	点击	"Estimate"	开始辨识二阶系统 ;

A Transfer Functions
Model name: tf1 /
Number of poles: 2
Number of zeros: 1
O Continuous-time ○ Discrete-time (Ts = 0.001) Feedthrough Feedthrough Feedthrough Seedthrough Seedthrough Seedthrough Feedthrough Seedthrough Seedthrough
▶ I/O Delay
Estimation Options
Estimate Close Help

15) 点击 "Estimate-->" 选择 "PolynomialModels", 设置 "Orders" 为[441], 点击 "Estimate" 开始辨识; 备注说明:多项式模型[441],其中第一个"4"代表输出 y 的阶数;第二个"4"代表输入 u 的阶数, 最后一个"1"代表误差为白噪声。

📕 Polynomial and State Space Models 🛛 🔲 🛛 🔀					
Structure:	ARX: [na nb nk]				
Orders:	[441]				
Equation:	А	y = Bu + e			
Method:	ARX				
Domain:	Continuous	Oiscrete (0)	.001 seconds)		
Add noise inte	egration ("ARIX"	model)			
Input delay:	0				
Name:	arx441				
Focus: P	rediction 👻	Initial state:	Auto 👻		
Dist.model: Es	timate	Covariance:	Estimate 💌		
Display progress Stop iterations					
Order Selection Order Editor					
Estimate Close Help					

16) 辨识完成后,输出辨识结果"tf1"为一阶系统辨识模型,"tf2"为二阶系统辨识模型,"arx441"为多 项式模型,勾选"Modeloutput",查看辨识效果;

🛃 System Identification Tool - Untitled								
File Options Window	Help			Model Output: y1	- • ×			
Import data	Operations	Import	t models 🔹	File Options Style Channel Help E	xperiment			
mydata	< Preprocess		12 arx441	Measured and simulated model output 25 20 مېلې مېلې مېلې مېلې مېلې مېلې مېلې مېل	Best Fits f/2: 79.29 f/1: 77.61 arx441: 75.03			
	Estimate>			10 5				
Data Views	To To Workspace LTI Viewer	📝 Model output	Model Views	0 · ·				
🔲 Data spectra	200	Model resids	Frequency resp	-5 <u>5</u> 1015]			
Frequency function	(())) Trash	mydata Validation Data	Zeros and poles Noise spectrum	Time				

17) 双击"tf1"、"tf2"及"arx441",即可看到系统辨识模型;

Data/model Info: tf1					
Model name:	tf1				
Color:	[0,0,1]				
From input "u1" to output "y1": 161.5 					
Diary and Notes					
Data/model Info: tf2					
Model name:	tf2				
Color:	[0.25,0.75,0.25]				
From input "u1" to output "y1 -133.7 s + 9271 	": r of zeros: 1 s: 4	E			
Diary and Notes					

```
Model name:arx441Color:[1,0,0]Discrete-time ARI model: A(z)y(t) = B(z)u(t) + e(t)<br/>A(z) = 1 - 1.109 z^2 - 1 - 0.3391 z^2 - 2 + 0.1026 z^2 - 3 + 0.346 z^2 - 4<br/>B(z) = 0.09457 z^2 - 1 - 0.09517 z^2 - 2 - 0.06655 z^2 - 3 + 0.08448 z^2 - 4Name: arx441<br/>Sample time: 0.001 secondsParameterization:<br/>Polynomial orders: na=4 nb=4 nk=1<br/>Number of free coefficients: 8<br/>Use "polydata", "getpvec", "getcov" for parameters and their uncertainties.Status:<br/>Estimated using ANI on time domain data "mydata".
```

% Import mydata Opt = arxOptions; arx441 = arx(mydata,[4 4 1], Opt);

18) 记录系统辨识结果(以下仅为参考,不同系统模型会不同):

 $G_0 = \frac{161.5}{s + 8.163}$

 $G_0 = \frac{-133s + 9271}{s^2 + 57.96s + 468.8}$

19) 重复步骤 1)-11),将图 2.1.2 中的阶跃信号(step)换成扫频信号(Chirp Signal),如下图所示:

	Source Block Parameters: Chirp Signal				
M) • ignal	chirp (mask) (link)				
	Output a linear chirp signal (sine wave whose frequency varies linearly with time).				
	Parameters				
	Initial frequency (Hz):				
	Target time (secs):				
	100				
	Frequency at target time (Hz):				
	1				
	📝 Interpret vector parameters as 1-D				
	OK Cancel Help Apply				

提示: Frequency at target time(Hz),可以设置为 1~10 之间。 程序运行时间建议设置为 90 秒。如果运行时间设置为 100s,生成的数据可能只有两个数。

20) 重复步骤 1)-11),将图 2.1.2 中的扫频信号(Chirp Signal)换成伪随机序列信号(M序列),可以 采用如下程序方式生成:





21)运行 M 序列信号输入的程序,建议运行时间小于 30s。

22)采用自己编写的最小二乘辨识程序(程序一定要准确),分别采集相应的输入输出数据,利用最小 二乘辨识程序辨识系统模型,记录下来并与系统辨识工具箱辨识的结果进行比较。

(五)实验记录与实验结果分析

1、实验法建模实验数据记录

模型	阶跃信号	扫频信号	M 序列
一阶系统开 环传递函数			
	拟合度:	拟合度:	拟合度:
二阶系统开 环传递函数			
	拟合度:	拟合度 :	拟合度:
多项式模型			
	拟合度:	拟合度:	拟合度:
最小二乘辨 识模型			

2、对比分析工具箱辨识模型与最小二乘辨识模型的区别。

3、去掉 simulink 程序中的滤波模块,分析滤波器的作用。

提示:可以使用扫频信号或 M 序列,获取一组数据,对系统模型进行辨识,分析模型的拟合效果。