**实验11 离散系统的描述模型及其转换**

1518班 15352408 张镓伟

**一、实验目的**

(1) 了解离散系统的基本描述模型。  
　　(2) 掌握各种模型相互间的关系及转换方法。  
　　(3) 熟悉MATLAB中进行离散系统模型间转换的常用子函数。

二、**实验涉及的MATLAB子函数**

**1. tf2zp  
　　 功能：**将系统传递函数(tf)模型转换为系统函数的零-极点增益(zpk)模型。 **调用格式：** ［z，p，k］＝tf2zp(num，den)；输入系统传递函数模型中分子(num)、分

母(den)多项式的系数向量，求系统函数的零-极点增益模型中的零点向量z、

极点向量p和增益系数k。其中z、p、k为列向量。

**2. zp2tf  
　　 功能：**将系统函数的零-极点增益(zpk)模型转换为系统传递函数(tf)模型。 **调用格式：** ［num，den］＝zp2tf(z，p，k)；输入零-极点增益(zpk)模型零点向量z、极

点向量p和增益系数k，求系统传递函数(tf)模型中分子(num)、分母(den)

多项式的系数向量。

**3. tf2sos  
　　 功能：**将系统传递函数(tf)模型转换为系统函数的二次分式(sos)模型。 **调用格式：** ［sos，g］＝tf2sos(num，den)；输入系统传递函数模型中分子(num)、分

母(den)多项式的系数向量，求系统函数的二次分式模型的系数矩阵sos、

增益系数g。

**4. sos2tf**  
　　 **功能：**将系统函数的二次分式(sos)模型转换为系统传递函数(tf)模型。  
　　 **调用格式：**  
　　 ［num，den］＝sos2tf(sos，g)；输入系统函数的二次分式模型的系数矩阵

sos、增益系数g(默认值为1)，求系统传递函数模型中分子(num)、分母(den)

多项式的系数向量。

**5. sos2zp**  
　　 **功能：**将系统函数的二次分式(sos)模型转换为系统函数的零-极点增益(zpk)

模型。  
　　 **调用格式：**  
　　 ［z，p，k］＝sos2zp(sos，g)；输入系统函数的二次分式模型的系数矩阵

sos、增益系数g(默认值为1)，求系统函数的零-极点增益模型中的零点向

量z、极点向量p和增益系数k。

**6. zp2sos**  
　　 **功能：**将系统函数的零-极点增益(zpk)模型转换为系统函数的二次分式(sos)

模型。  
　　 **调用格式：**  
　　 ［sos，g］＝zp2sos(z，p，k)；输入系统函数的零-极点增益模型中零点向

量z、极点向量p和增益系数k，求系统函数的二次分式模型的系数矩阵

sos、增益系数g。

**7. ss2tf**  
　　 **功能：**将系统状态空间(ss)模型转换为系统传递函数(tf)模型。  
　　 **调用格式：**  
　　 ［num，den］＝ss2tf(A，B，C，D，xi)；可将系统状态空间(ss)模型转换为

相应的传递函数(tf)模型。xi用于指定变换使用的输入量。

**8. tf2ss**  
　　 **功能：**将系统传递函数(tf)模型转换为系统状态空间(ss)模型。  
　　 **调用格式：**  
　　 ［A，B，C，D］＝tf2ss(num，den)；将系统传递函数(tf)模型转换为系统状

态空间(ss)模型。num按s降幂排列顺序输入分子系数，den按s降幂排列

顺序输入分母系数。

**三、实验原理**

**1.** **离散系统的基本描述模型**

一个线性移不变(LSI)离散系统可以用线性常系数差分方程表示：

 (11-1)

这是系统在时间域的表达式，如果在变换域对系统进行描述，则可以采用以下几种模型。

(1)系统传递函数(tf)模型。对式(11-1)所示的线性常系数差分方程两边进行z变换，可以得到离散LSI系统的系统传递函数：

 (11-2)

(2)零-极点增益(zpk)模型。对式(11-2)表示的系统传递函数进行因式分解，可以得到系统传递函数的零-极点增益模型：

 (11-3)

(3)极点留数(rpk)模型。当式(8-3)模型中的极点均为单极点时，可以将式(11-3)分解为部分分式，表示为系统的极点留数模型：

 (11-4)

(4)二次分式(sos)模型。离散LSI系统函数经常包含复数的零、极点，把每一对共轭零点或共轭极点多项式合并，就可以得到二次分式模型：

 (11-5)

(5)状态变量(ss)模型。系统的状态方程可表示为：

 (11-6)

其中W为N维状态向量，X为R维输入向量，Y为M维输出向量

A为N\*N矩阵，称系统矩阵，B为N\*R矩阵，称输入矩阵或控制矩阵，

C为M\*N矩阵，称为输出矩阵，D为M\*R矩阵，称直接传输矩阵

表示为传递函数形式：

 (11-7)

在MATLAB中提供了上述各种模型之间的转换函数。这些函数为系统特性的分析提供了有效的手段。

**2. 系统传递函数(tf)模型与零-极点增益(zpk)模型间的转换**

**例11-1** 已知离散时间系统的传递函数  
　　   
　　 求系统的零点向量z、极点向量p和增益系数k，并列出系统函数的零-

极点增益模型。

**解** MATLAB程序如下：  
　　 num=[0,10,0];  
 den=[1,-3,2];  
 [z,p,k] = tf2zp(num, den)

程序运行结果如下：  
　　 z＝   
　　　　 0  
　　 p＝   
　　　　 2   
　　　　 1  
　　 k＝   
　　　　 10  
　　根据程序运行结果，零-极点增益模型的系统函数为

  
　　 **例11-2** 已知离散时间系统的零-极点增益模型



求系统的传递函数(tf)模型。

**解** MATLAB程序如下：  
　　 z=[1,-3]';  
 p=[2,-4]';  
 k=5;  
 [num, den]=zp2tf(z,p,k)

程序运行结果如下：  
　　 num＝   
　　 5 10 －15  
　　 den＝   
　　 1 2 －8  
　　 根据程序运行结果，可知系统的传递函数为



**3.** **系统传递函数(tf)模型与二次分式(sos)模型间的转换**

**例11-3** 将系统传递函数



转换为二次分式模型。

**解** MATLAB程序如下：

num=[1.9,2.5,2.5,1.9];  
 den=[1,-6,5,-0.4];  
 [sos,g]=tf2sos(num,den)

程序运行结果如下：  
　 sos =  
 1.0000 1.0000 0 1.0000 -5.0198 0  
 1.0000 0.3158 1.0000 1.0000 -0.9802 0.0797  
　 g＝  
　 1.9000

根据程序运行结果，可求出二次分式为



**例11-4** 已知系统的二次分式模型为

****

试将其转换为系统传递函数(tf)模型。

**解** MATLAB程序如下：  
　 sos=[1.0000 1.0000 0 1.0000 -0.5000 0;  
 1.0000 -1.4000 1.0000 1.0000 0.9000 0.8000];  
 g=4;  
 [num,den]=sos2tf(sos,g)

**程序运行结果如下：** num＝4.0000 －1.6000 －1.6000 4.0000  
　　 den＝ 1.0000 0.4000 0.3500 －0.4000根据程序运行结果，可求出系统传递函数为

****

**4. 零-极点增益(zpk)模型与二次分式(sos)模型间的转换**　　 **例11-5** 已知离散时间系统(如例11-2)的零-极点增益模型

，求系统的二次分式模型。

**解** MATLAB程序如下：  
　　 z=[1,-3]';  
 p=[2,-4]';  
 k=5;  
 [sos,g]=zp2sos(z,p,k)

程序运行结果如下：  
　　 sos＝1 2 －3 1 2 －8  
　　 g＝5  
　　 根据程序运行结果，可求出二次分式为



**例11-6** 已知离散时间系统的二次分式模型(如例11-3)为



求系统的零-极点增益模型。

**解** MATLAB程序如下：

sos =[1.0000 1.0000 0 1.0000 -5.0198 0;  
 1.0000 0.3158 1.0000 1.0000 -0.9802 0.0797];  
 g = 1.9;  
 [z,p,k]= sos2zp(sos,g)

程序运行结果如下：  
z =  
 －1.0000   
 －0.1579 ＋0.9875i  
 －0.1579 － 0.9875i  
p =  
 5.0198  
 0.8907  
 0.0895  
k =1.9000

根据程序运行结果，零-极点增益模型的系统函数为



**5. 系统传递函数(tf)模型与极点留数(rpk)模型间的转换**

在实验10中，我们用部分分式法求系统函数的z反变换，实际上也就是利用residuez子函数，将系统的传递函数(tf)模型转换为极点留数(rpk)模型。反之，利用residuez子函数，还能将系统的极点留数(rpk)模型转换为传递函数(tf)模型。

**例11-7** 已知离散时间系统的传递函数(tf)模型(如例11-1)为

，求系统的极点留数(rpk)模型。

**解** MATLAB程序如下：  
　　 num=[0,10,0];  
 den=[1,-3,2];  
 [r,p,k] = residuez(num, den)

程序运行结果如下：  
　　r＝   
　　 　10   
　　 －10  
　　p＝   
　　　　2   
　　　　1  
　　k＝0  
　　根据程序运行结果，极点留数(rpk)模型为



**例11-8** 已知离散时间系统的极点留数(rpk)模型为

求系统的传递函数(tf)模型。

**解** MATLAB程序如下：  
　　　 r=[2,-1,1]';  
 p=[1,0.5,-0.5]';

k=0;  
 [num, den]=residuez(r,p,k)

程序运行结果如下：  
　　 num＝2.0000 －1.0000 0.5000 　 　0  
　　 den＝1.0000 －1.0000 －0.2500 0.2500  
　　 根据程序运行结果，可求出系统传递函数为



**6.系统传递函数(tf)模型与状态变量(ss)模型间的转换**

**例11-9** 将系统传递函数

  
转换为状态变量模型。

**解** MATLAB程序如下：  
　　 　num=[1.9,2.5,2.5,1.9];  
 den=[1,-6,5,-0.4];  
 [A,B,C,D]=tf2ss(num,den)

程序运行结果如下：  
　　 A＝6.0000 　－5.0000 　0.4000  
　　 1.0000 　　0 0  
　　 　　　0 　 1.0000 0  
　　 B＝1  
　　　　 0  
　　　　 0  
　　 C＝13.9000 －7.0000 2.6600  
　　 D＝1.9000  
　　 将以上数据代入式(11-6)，可得到系统的状态方程。  
　　 同理，如果知道式(11-6)，则可以由状态变量模型转变为系统传递函数形

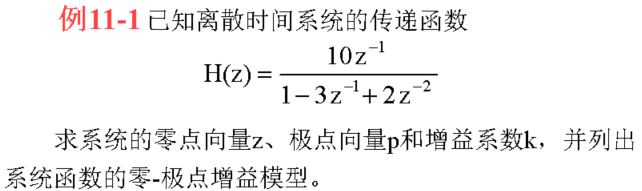
式。

**四、实验任务**

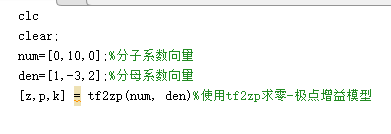
(1) 阅读并输入实验原理中介绍的例题程序，理解每一条语句的含义，观察程

序输出数据及公式。

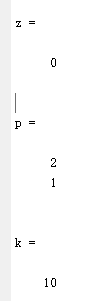
以例11-1为例。



代码及注释：



程序输出结果:



根据程序运行结果，零-极点增益模型的系统函数为



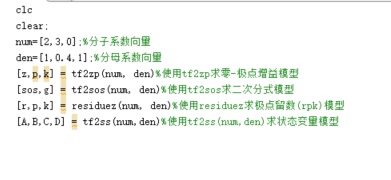
(2) 已知离散时间系统的传递函数(tf)模型：

要求将其转换为：

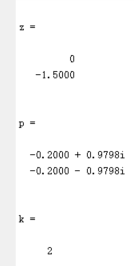
①零-极点增益(zpk)模型；  
　　②二次分式(sos)模型；  
　　③极点留数(rpk)模型；  
　　④状态变量(ss)模型。

Matlab代码如下：



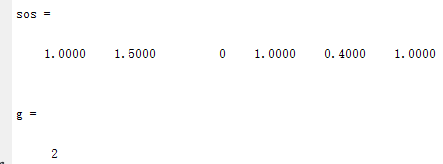
程序输出结果：

①零-极点增益模型：



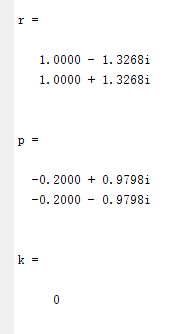
根据输出结果，可得零-极点增益模型的系统函数为

②二次分式(sos)模型:



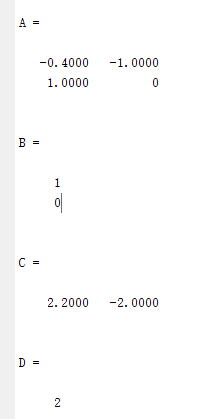
根据输出结果，可得二次分式模型为

　　 ③极点留数(rpk)模型



根据输出结果，可得极点留数模型为

　　 ④状态变量(ss)模型。



根据输出结果，可得状态变量模型为

W(n+1)=W(n) + X(n)

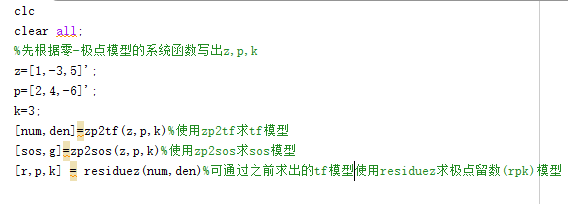
Y(n+1)=W(n) + [2]X(n)

(3) 已知离散时间系统的零-极点增益(zpk)模型为



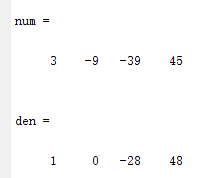
要求将其转换为：  
　　①传递函数(tf)模型；  
　　②二次分式(sos)模型；  
　　③极点留数(rpk)模型。

Matlab代码如下：



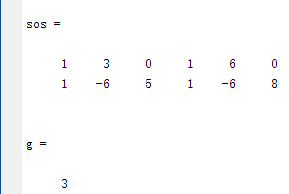
程序输出结果：

①传递函数(tf)模型；



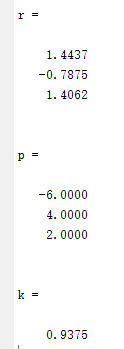
根据输出结果可得传递函数模型为：

　　 ②二次分式(sos)模型；



根据输出结果可得二次分式模型为：

　　 ③极点留数(rpk)模型。



根据输出结果，可得极点留数模型为

(4) 思考题：  
　①回答预习思考题: 离散系统有几种常用的系统描述模型？它们的公式如

何？

答：有种常用的系统描述模型。

1. 系统传递函数(tf)模型



2. 零-极点增益(zpk)模型



3. 极点留数(rpk)模型



4. 二次分式(sos)模型



5. 状态变量(ss)模型。系统的状态方程可表示为：



表示为传递函数形式：



其中W为N维状态向量，X为R维输入向量，Y为M维输出向量

A为N\*N矩阵，称系统矩阵，B为N\*R矩阵，称输入矩阵或控制矩阵，

C为M\*N矩阵，称为输出矩阵，D为M\*R矩阵，称直接传输矩阵

②通过本实验，你能进行哪些系统描述模型之间的转换？

答：tf –>zpk

tf –>rpk

tf –>sos

tf –>ss

zpk –>tf

zpk –>rpk

zpk –>sos