系统建模与仿真 第一次作业 190320521 刘自涵

蒙特卡罗方法求π

```
N1 = 1000000;
x1 = -1+ 2*rand(1,N1);
y1 = -1+ 2*rand(1,N1);
num2 = sum (x1.^2+y1.^2 < 1);
out2 = 4*num2/N1;
out2 = sprintf('%.12f',out2);
disp(out2);
```

结果

```
>> mtklf_pi
3.142672000000
>>
```

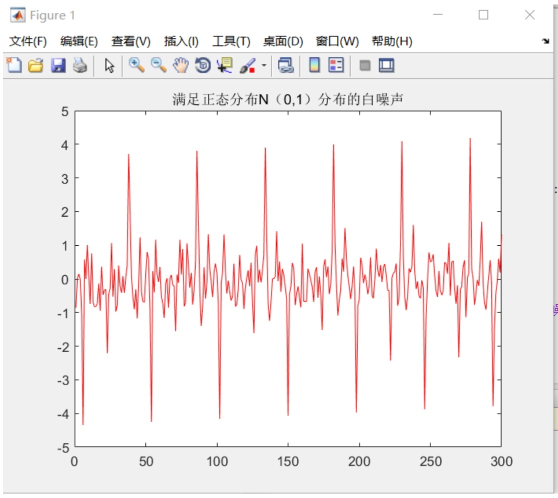
蒲丰投针求π

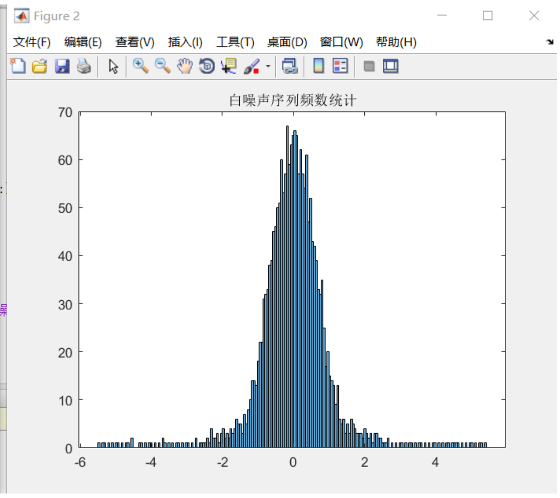
```
a=3; %平行线宽度
l=2;%针长度
N=10000000;%做n次投针试验
x=1.5*rand(1,N);%在[0, a/2]内服从均匀分布随机产生N个数
theta=(pi/2)*rand(1,N); %在[0, pi/2]内服从均匀分布随机产生N个数
n=sum(x<(l/2)*sin(theta));%记录针与平行线相交的次数
p=(2*l*N)/(a*n);%得到pi的值
out1=sprintf('%.12f',p);%控制输出位数
disp(out1)%输出结果
```

结果

```
>> pufengneedle
3.143098847158
>>
```

190320521 刘自涵 第二次作业 系统建模与仿真






```matlab
%白噪声产生程序
A=7; x0=1; M=16384; f=2; N=24000;
for k=1:N
x1=A*x0;
x2=mod(x1,M);
v1=x2/M;
v(:,k)=v1;
x0=x2;
v0=v1;
end

B=reshape(v,[2000,12]);
for B_length = 1:2000
    C(B_length,:)=(sum(B(B_length,:))-12*0.5)/(sqrt(12/12));
end
k=1:300;
noise_draw = C(1:300);
figure(1);
plot(k,noise_draw,'r');
title('满足正态分布N（0,1）分布的白噪声');
figure(2);
histogram(C,200);
title('白噪声序列频数统计')
```

系统建模与仿真第三次作业 190320521 刘自涵

```
clc;
clear;
N=5;
y = [0.3;0.5;-0.2;0.6;0.83];
u = [2.1;-2.7;0.8;1.5;-2.1];
Y = y(2:5);
PHI = zeros(4,3);
for j = 1:N-1
        PHI(j,1)=- y(j);
        PHI(j,2)= u(j+1);
        PHI(j,3)= u(j);
end
theta_hat = (inv(PHI'*PHI))* PHI'*Y;
disp(theta_hat);
```

命令行窗口

```
   -1.4862
    0.3456
    0.4569
```

哈尔滨工业大学（深圳）

《系统建模与仿真》课程实验报告

（2020-2021 秋季学期）

课程名称 ： 系统建模与仿真

题　　目 ： 利用相关分析法辨识脉冲响应

班级学号 ： 190320521

学生姓名 ： 刘自涵

2020 年 10 月 25 日

一、实验目的

通过仿真实验掌握利用相关分析法辨识脉冲响应的原理和方法。

二、实验内容

图 1 为本实验的原理框图。系统的传递函数为$G(s)$，

$$G(s)=\frac{K}{(T_1 s+1)(T_2 s+1)}$$

其中$K=120, T_1=8.3\mathrm{Sec}, \mathrm{T}_2=6.2\mathrm{Sec}$；$u(k)$和$z(k)$分别为过程的输入和输出变量；$v(k)$为测量白噪声过程，服从正态分布，均值为零，方差为$\sigma_v^2$，记作$v(k)\sim N(0,\sigma_v^2)$；$g_0(k)$为系统脉冲响应的理论值，$\hat{g}(k)$为系统脉冲响应的估计值，$\tilde{g}(k)$为系统脉冲响应的估计误差。

过程的输入驱动采用 M 序列，输出受到白噪声$v(k)$的污染。根据过程的输入和输出数据$\{u(k), z(k)\}$，利用相关分析算法辨识系统脉冲相应。

根据输出过程的脉冲响应值$\hat{g}(k)$，并与过程脉冲响应理论值$g_0(k)$比较，得到过程脉冲响应估计误差值$\tilde{g}(k)$，当$k\to\infty$时，应该有$\tilde{g}(k)\to 0$。

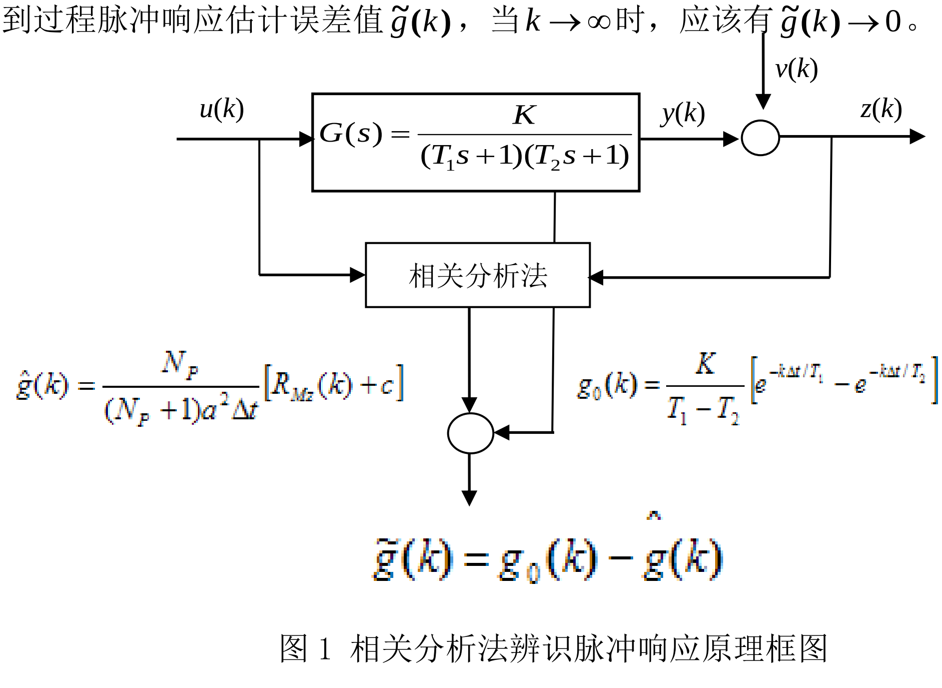


图 1 相关分析法辨识脉冲响应原理框图

三、实验要求

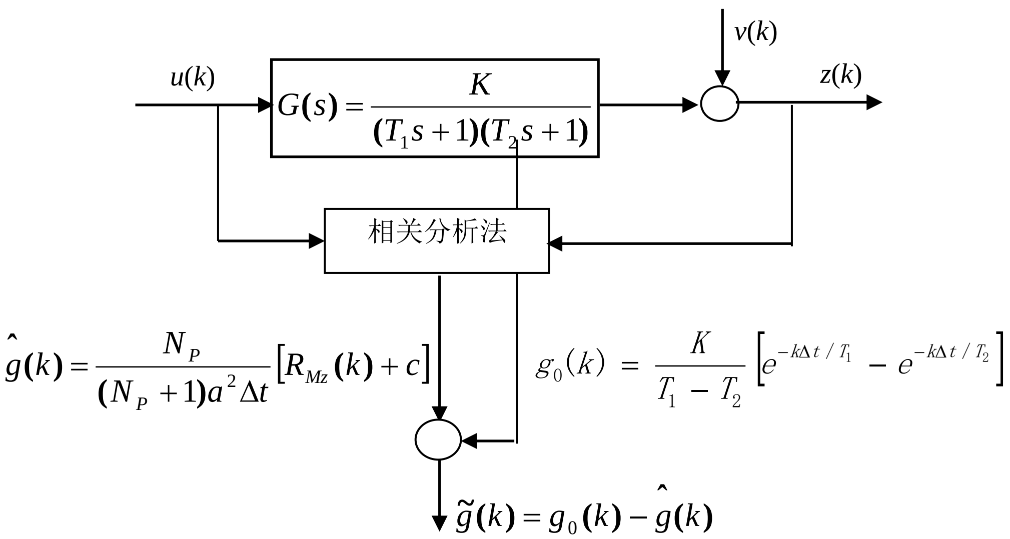
进行方案设计，模拟过程传递函数，获得输出数据，用M序列作为辨识的输

入信号，噪声采用标准正态分布的白噪声，计算互相关函数，脉冲响应估计值、脉冲响应理论值和脉冲响应估计误差，计算信噪比，画出实验流程图，用MATLAB编程实现。

四、实验原理

下图为本实验的原理框图。系统的传递函数为 $G(s)$，其中 $K=120, T_1=8.3\text{Sec}, \text{T}_2=6.2\text{Sec}$；$u(k)$和$z(k)$分别为系统的输入和输出变量；$v(k)$为测量白噪声，服从正态分布，均值为零，方差为$\sigma_v^2$，记作$v(k)\sim N(0,\sigma_v^2)$；$g_0(k)$为系统的脉冲响应理论值，$\hat{g}(k)$为系统脉冲响应估计值，$\tilde{g}(k)$为系统脉冲响应估计误差。

系统的输入采用 M 序列（采用实验 1 中的 M 序列即可），输出受到白噪声$v(k)$的污染。根据过程的输入和输出数据$\{u(k), z(k)\}$，利用相关分析法计算出系统的脉冲响应值$\hat{g}(k)$，并与系统的脉冲响应理论值$g_0(k)$比较，得到系统脉冲响应估计误差值$\tilde{g}(k)$，当$k\to\infty$时，应该有$\tilde{g}(k)\to 0$。



1、 利用 lsim()函数获得传递函数$G(s)$的输入和输出数据$\{u(k), z(k)\}$（采样

时间取 1 秒）。

2、 互相关函数的计算

$$R_{Mz}(k)=\frac{1}{rN_P}\sum_{i=N_P+1}^{(r+1)N_P}u(i-k)z(i)$$

其中，r 为周期数，$i=N_P+1$ 表示计算互相关函数所用的数据是从第二个周期开始的，目的是等过程仿真数据进入平稳状态。（可分别令 r =1、3，对比仿真结果）

3、 补偿量 c

补偿量 c 应取－$R_{Mz}(N_P-1)$，不能取－$R_{Mz}(N_P)$。因为 $R_{Mz}(k)$ 是周期函数，则有 $R_{Mz}(N_P)$＝$R_{Mz}(0)$，故不能取－$R_{Mz}(N_P)$。

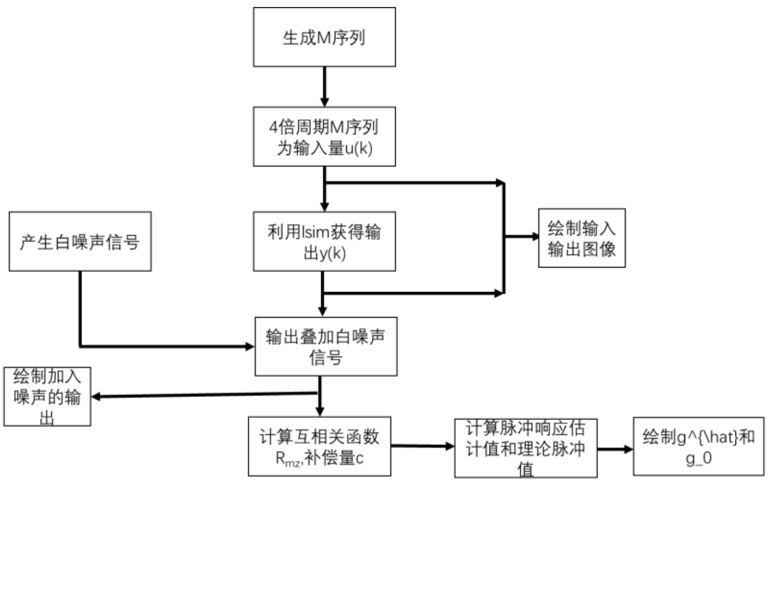
4、 计算脉冲响应估计值

- 理论脉冲响应值 $g_0(k)=\frac{K}{T_1-T_2}\left[e^{-k\Delta t/T_1}-e^{-k\Delta t/T_2}\right]$

- 脉冲响应估计值 $\hat{g}(k)=\frac{N_P}{(N_P+1)a^2\Delta t}\left[R_{Mz}(k)+c\right]$

- 脉冲响应估计误差 $\delta_g=\sqrt{\sum_{k=1}^{N_P}\left(g_0(k)-\hat{g}(k)\right)^2 \Big/ \sum_{k=1}^{N_P}\left(g_0(k)\right)^2}$

五、实验框图



六、实验程序代码

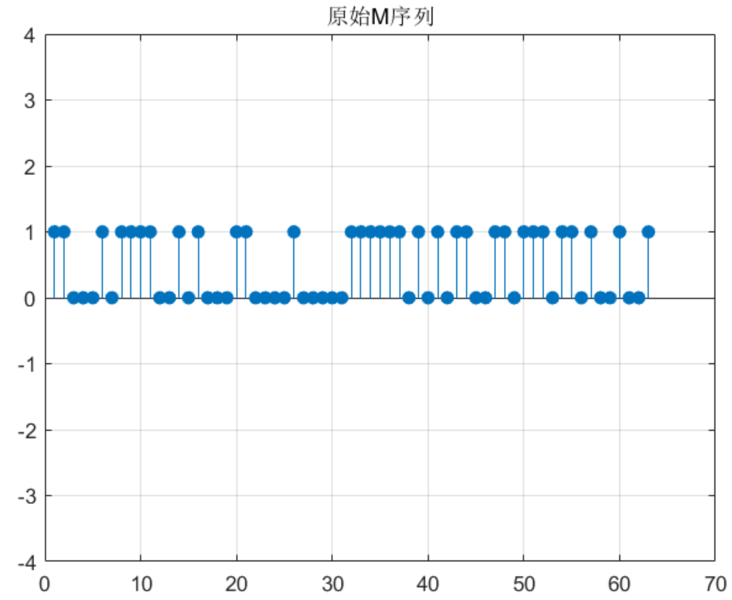
主程序

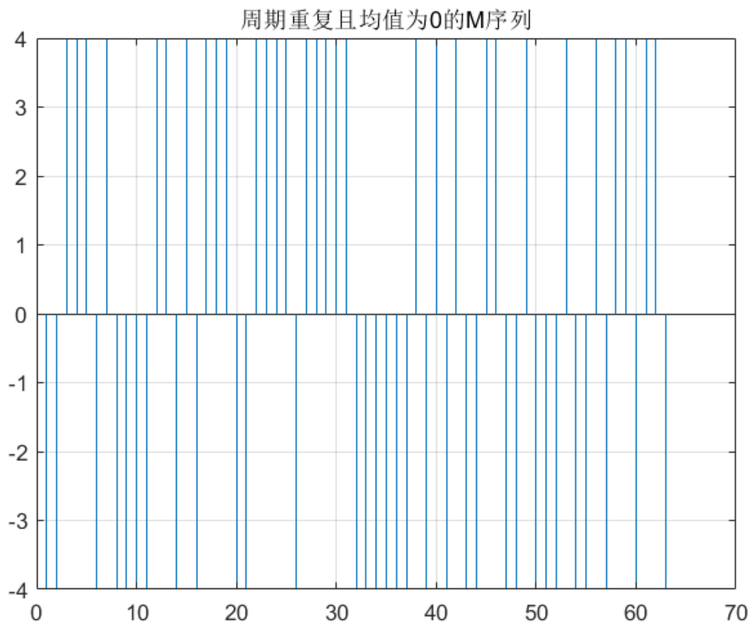
```
% 9位寄存器生成M序列
clc;
clear;
N=6;
a=5;
M=[0,0,0,1,1,1];
Np = 2^N -1;
u=zeros(1,Np);
Repeat_T = 4;
for i = 1:Np
    %先运算，后移位
     m=xor(M(1),M(6));
    for j = 6:-1:2
        M(j)=M(j-1);
    end
    M(1)=m;
    u(i)=M(6);
end

figure(1);
stem(u,'filled');
grid on;
k=1:Np;
u = (1-2*u)*a;
ylim([-4,4]);
title('原始M序列');
M_line = repmat(u,1,(Repeat_T+1));
x =1:Np*(Repeat_T+1);

figure(2);
stem(u,'filled');
grid on;
ylim([-4,4]);
title('周期重复且均值为0的M序列');

K =120;
T1 = 8.3;
T2 = 6.2;
T0=1;
tt = 1:1:Np*(Repeat_T+1);
```

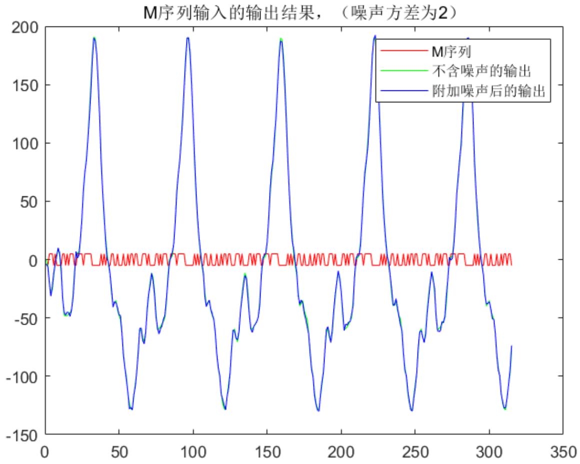
```
den = [51.46,14.5,1];
num = 120;
G=tf(num,den);
y = lsim(G,M_line,tt);
y=y';
figure(3);
plot(M_line,'r');
hold on;
plot(y,'g');
white_noise = create(2);
z=y+(white_noise(1:length(y)))';
hold on;
plot(z,'b');
title('M序列输入的输出结果，（噪声方差为2）');
legend('M序列','不含噪声的输出','附加噪声后的输出')

%互相关函数的计算
for k = 1:Np
    Rmz(k)=0;
    for hxg_i = (Np+1):((Repeat_T+1)*Np)
        Rmz(k)=Rmz(k)+M_line(hxg_i-k)*z(hxg_i);
    end
    Rmz(k)=Rmz(k)/(Repeat_T*Np);
end

M = eye(Np)+ones(Np);
%计算脉冲估计g_hat
x_cord = 1:Np;
g_hat = (Np/((Np+1)*(a^2)*T0))*M*Rmz';
figure(4);
plot(x_cord,g_hat(1:Np));
title('脉冲响应估计值');

g0(Np)=0;
for k=1:Np
    g0(k)=(K/(T1-T2))*(exp(-k*T0/T1)-exp(-k*T0/T2));
end
figure(5);
plot(x_cord,g0','r');
title('脉冲响应理论值');

%计算估计误差
error_sum=0;
square_sum=0;
```

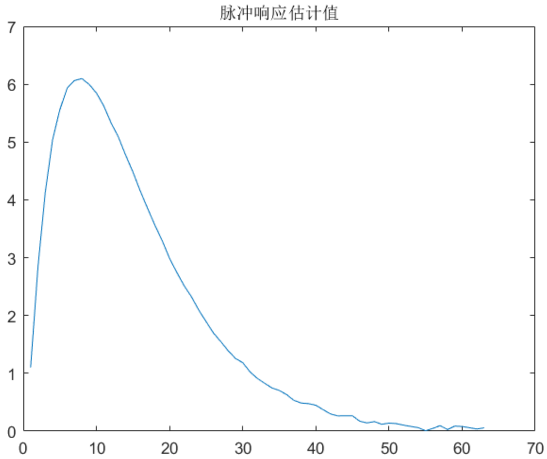
```
for k = 1:Np
    error_sum = error_sum+(g0(k)-g_hat(k))^2;
    square_sum = square_sum+(g0(k))^2;
end
sigma_g = sqrt(error_sum/square_sum);
disp(['脉冲估计误差为：',num2str(sigma_g)]);
```

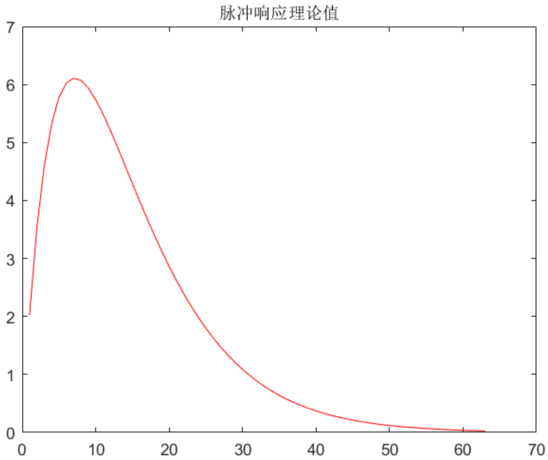
七、实验结果及分析











脉冲估计误差为：0.064751

fx >>

八、实验结论

实验给定模型，以 M 序列作为输入，在输出结果上叠加满足正态分布的白噪声,得到输出数据。通过利用维纳霍夫方程，计算得到脉冲响应估计输出曲线，通过已知的模型计算得到脉冲响应理论输出计算，计算得到脉冲估计误差为 0.064751。可以看出，利用维纳霍夫方程和 M 序列进行系统辨识的相关分析法在分析经典线性系统时有较好的效果。

哈尔滨工业大学（深圳）

《系统建模与仿真》课程实验报告

（2020-2021 秋季学期）

课程名称 ： 系统建模与仿真

题　　目 ： 利用递推最小二乘算法辨识模型参数

班级学号 ： 190320521

学生姓名 ： 刘自涵

2020 年 10 月 29 日

一、实验目的

通过仿真实验掌握利用递推最小二程方法辨识差分方程模型参数的原理和方法。

二、实验内容

给出系统的差分方程为

$$
\begin{aligned}
x(k) &= a_1x(k-1)+a_2x(k-2)+b_1u(k-1)+b_2u(k-2) \\
z(k) &= x(k)+v(k)
\end{aligned}
$$

其中 $a_1=1.5, a_2=-0.7, b_1=1, b_2=0.5$ ； $u(k), x(k)$和$z(k)$ 分别为过程的输入，状态和输出变量； $\boldsymbol{v(k)}$ 为测量白噪声过程，服从正态分布，均值为零，方差为 σ_v^2 ，记作 $\boldsymbol{v(k) \sim N(0, \sigma_v^2)}$ 。

过程的输入驱动采用 M 序列，输出受到白噪声 $\boldsymbol{v(k)}$ 的污染。根据过程的输入和输出数据 $\{\boldsymbol{u(k), z(k)}\}$，利用递推最小二乘算法辨识系统模型参数 a_1, a_2和b_1, b_2 。

三、实验要求

进行方案设计，模拟过程进行仿真，获得输出数据，用M序列作为辨识的输入信号，噪声采用根方差 $\sigma_v=0.01$ 的正态分布白噪声，通过最小二乘辨识模型参数，计算参数估计值与理论值之间的误差，画出相应的仿真曲线，分析噪声及算法对辨识结果的影响。

四、实验原理

最小二乘法的辨识准则是使残差平方和最小，对残差方程对参数向量取极值，若 $\left(\mathbf{\Phi}^T\mathbf{\Phi}\right)$ 的逆矩阵存在，可以得到

$$
\boldsymbol{\theta} = \left(\mathbf{\Phi}^T\mathbf{\Phi}\right)^{-1}\mathbf{\Phi}^T\mathbf{Y}
$$

即为基本最小二乘法的实验原理。

为了解决 $n+N$ 组观测数据时的参数估计值已知，又得到了一组新的观测值 $\left(u\left(n+N+1\right), y\left(n+N+1\right)\right)$，如何采用最小二乘法进行在线估计新的估计值问题。

首先根据已有的数据，写出测量矩阵、参数矩阵、噪声矩阵、和输出向量的关系利用基本最小二乘法计算得到参数估计值矩阵。然后将新获得的数据加入原有的矩阵方程，对矩阵

分块，可得到新的参数估计与新数据和原有数据的递推关系式为

$$\boldsymbol{\theta}_{N+1} = \mathbf{P}_{N+1}\left(\boldsymbol{\Phi}^{\mathbf{T}}{}_{N}\mathbf{Y}_{N} + \boldsymbol{\psi}_{N+1}y_{N+1}\right)$$

$$\mathbf{P}_{N+1} = \left(\mathbf{P}_{N}{}^{-1} + \boldsymbol{\psi}_{N+1}\boldsymbol{\psi}^{\mathbf{T}}{}_{N+1}\right)^{-1}$$

利用矩阵的求逆引理，可以求得 LS 的递推算法

$$\boldsymbol{\theta}_{N+1} = \boldsymbol{\theta}_{N} + \mathbf{P}_{N}\boldsymbol{\psi}_{N+1}\left(1+\boldsymbol{\psi}^{T}{}_{N+1}\mathbf{P}_{N}\boldsymbol{\psi}_{N+1}y_{N+1}\right)^{-1}\left(y_{N+1} - \boldsymbol{\psi}^{T}{}_{N+1}\boldsymbol{\theta}_{N}\right)$$

其中

$$\boldsymbol{\theta}_{N+1} = \boldsymbol{\theta}_{N} + \mathbf{K}_{N+1}\left(y_{N+1} - \boldsymbol{\psi}^{T}{}_{N+1}\boldsymbol{\theta}_{N}\right)$$

$$\mathbf{K}_{N+1} = \mathbf{P}_{N}\boldsymbol{\psi}_{N+1}\left(1+\boldsymbol{\psi}^{T}{}_{N+1}\mathbf{P}_{N}\boldsymbol{\psi}_{N+1}y_{N+1}\right)^{-1}$$

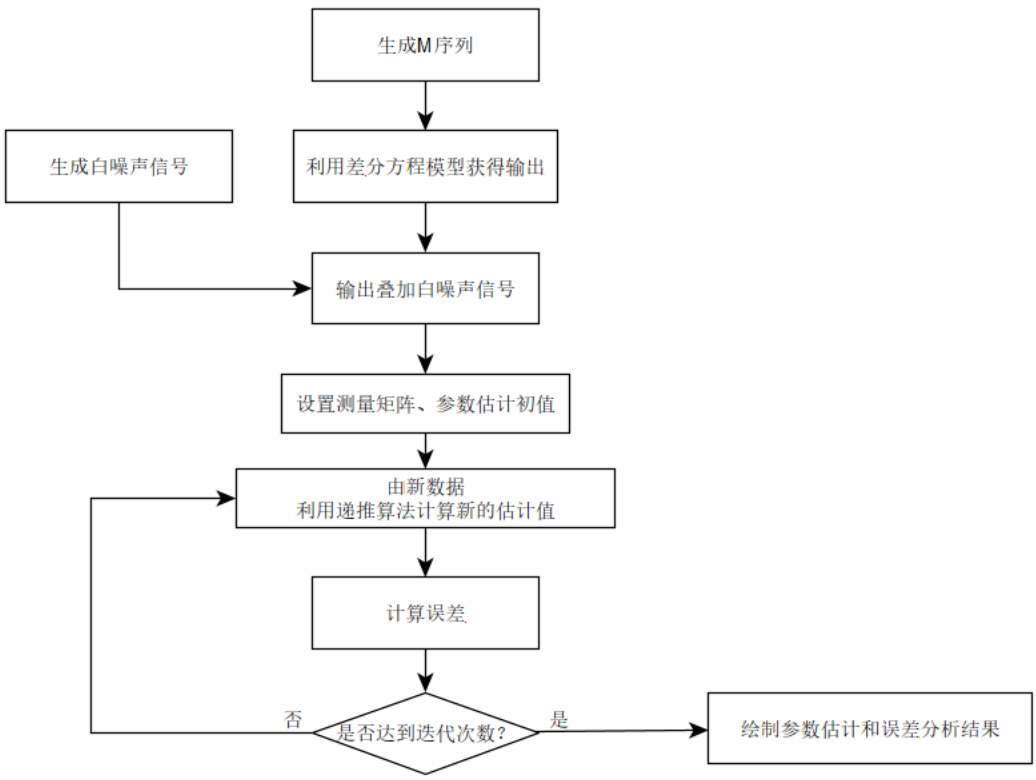
$$\mathbf{P}_{N+1} = \mathbf{P}_{N} - \mathbf{P}_{N}\boldsymbol{\psi}_{N+1}\left(1+\boldsymbol{\psi}^{T}{}_{N+1}\mathbf{P}_{N}\boldsymbol{\psi}_{N+1}\right)^{-1}\boldsymbol{\psi}^{T}{}_{N+1}\mathbf{P}_{N}$$

递推算法运行需要两个初值 $\boldsymbol{\theta}_0$ 、 $\mathbf{P}_0$ ，获取方法为

(1) 记录一组少量的 IO 数据 $\left(N_0 > (2n+1)\right)$ ，采用 LS 方法估计出 $\boldsymbol{\theta}_0$ 、 $\mathbf{P}_0$ ；

(2) 直接取 $\mathbf{P}_0 = c^2\mathbf{I}_{(2n+1)\times(2n+1)}$, c 为充分大的数。

五、实验框图



六、实验程序代码

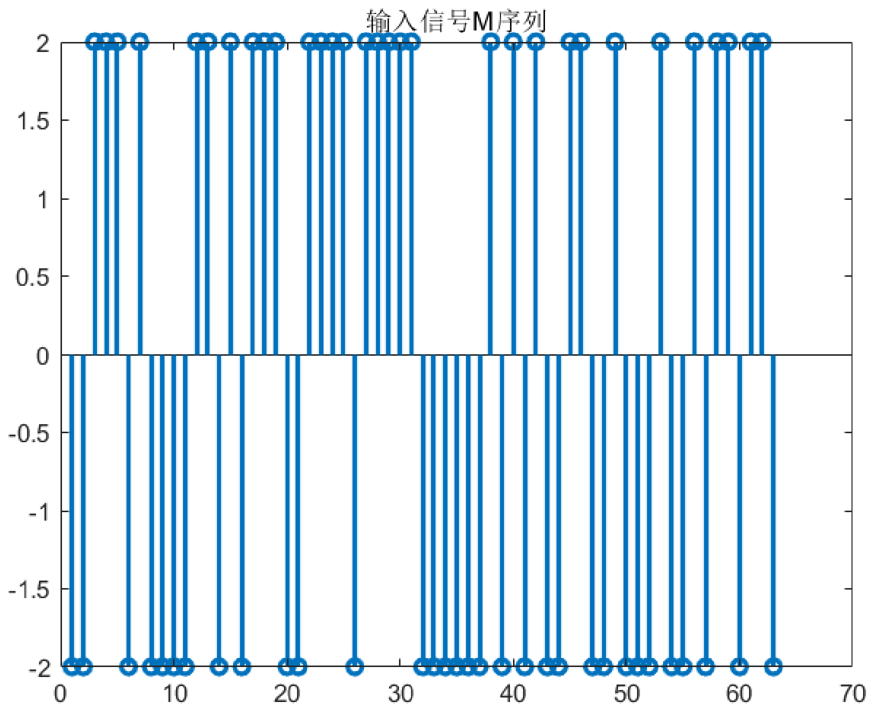
主程序

```
% 9位寄存器生成M序列
clc;
clear;
N=6;
a=2;
M=[0,0,0,1,1,1];
Np = 2^N -1;
u=zeros(1,Np);
for i = 1:Np
    %先运算，后移位
     m=xor(M(1),M(6));
    for j = 6:-1:2
        M(j)=M(j-1);
    end
    M(1)=m;
    u(i)=M(6);
end
figure(1);
u = (1-2*u)*a;
stem(u,'linewidth',2);
title('输入信号M序列')
x=zeros(1,Np);
z=zeros(1,Np);
x(2)=0,x(1)=0;
%待辨识的模型 获得理想输出
```

```
a1_s = 1.5;
a2_s = -0.7;
b1_s = 1;
b2_s = 0.5;
for k = 3:Np

x(k)=a1_s*x(k-1)+a2_s*x(k-2)+b1_s*u(k-1)+b2_s*u(k-2);
end
noise = white_n(0.01,length(x))';
%获得实际输出
z=x+noise;
figure(2);
hold on;
plot(z,'r','linewidth',1);
plot(x,'g','linewidth',1);
title("理想输出和噪声污染后输出");
legend('理想输出','噪声污染后输出');
%设置模型初始值
c0 = [0.1,0.1,0.1,0.1]';
p0 = 10^6*eye(4,4);
pri_set = [-a1_s,-a2_s,b1_s,b2_s]';
%按照模型公式，此处a1 a2 添加负号
pri_eva = [c0,c0,zeros(4,Np-2)];
pri_error = [c0-pri_set,c0-pri_set,zeros(4,Np-2)];
y = zeros(4,Np);
for k =3:Np
    h_new = [-z(k-1),-z(k-2),u(k-1),u(k-2)]';
```

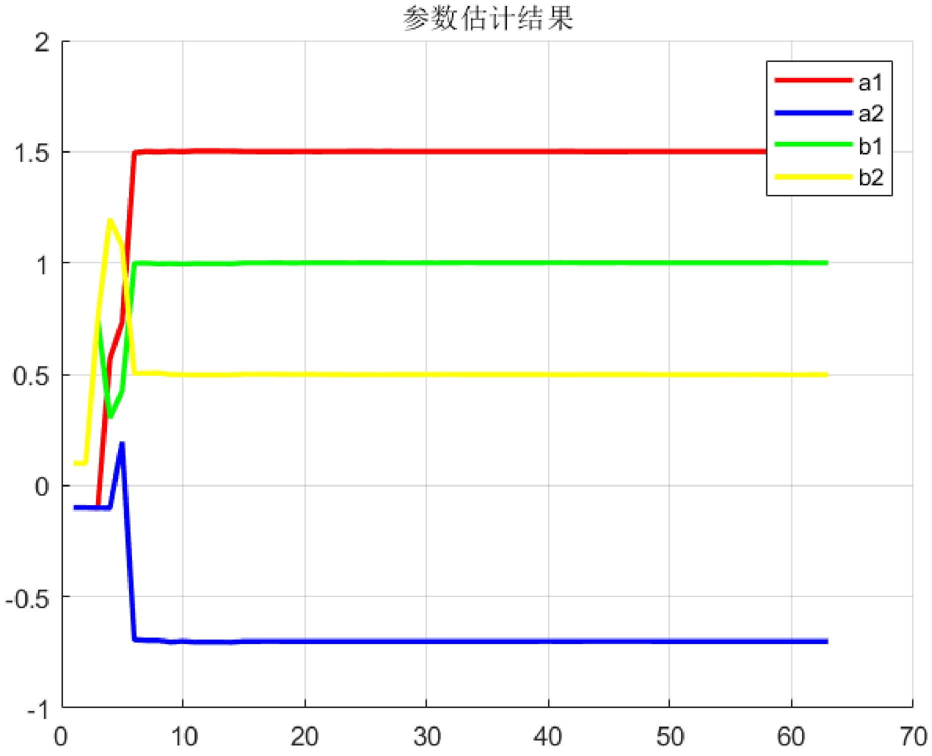
```
    temp = inv(h_new'*p0*h_new+1);
    K = p0*h_new*temp;
    c0 = c0+K*(z(k)-h_new'*c0);
    pri_eva(:,k)=c0;
    pri_error(:,k) = pri_eva(:,k)-pri_set;
    p0 = p0-p0*h_new*temp*h_new'*p0;
end
figure(3);
hold on;
grid on;
plot(pri_eva(1,:)*-1,'r','linewidth',2);
plot(pri_eva(2,:)*-1,'b','linewidth',2);
plot(pri_eva(3,:),'g','linewidth',2);
plot(pri_eva(4,:),'y','linewidth',2);
title("参数估计结果");
legend('a1','a2','b1','b2');
figure(4)
hold on;
grid on;
plot(pri_error(1,:)*-1,'r','linewidth',2);
plot(pri_error(2,:)*-1,'b','linewidth',2);
plot(pri_error(3,:),'g','linewidth',2);
plot(pri_error(4,:),'y','linewidth',2);
title("参数估计误差");
legend('a1','a2','b1','b2');
```

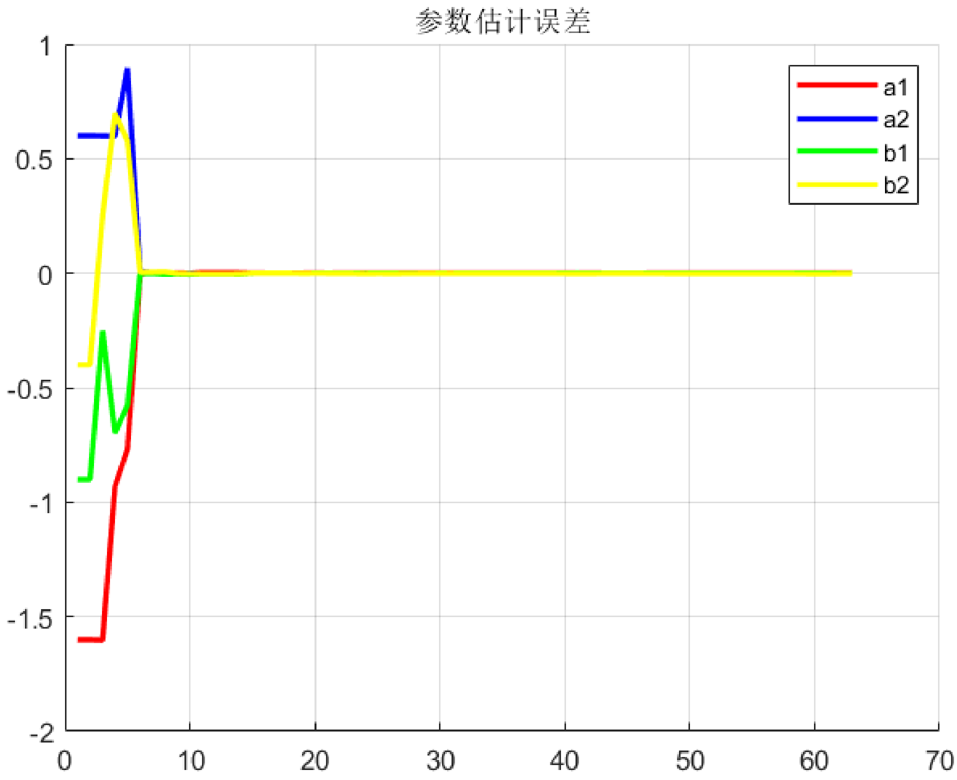
七、实验结果及分析



输入信号M序列
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
0
10
20
30
40
50
60
70

理想输出和噪声污染后输出
理想输出
噪声污染后输出
20
15
10
5
0
-5
-10
-15
-20
-25
0
10
20
30
40
50
60
70


参数估计结果
a1
a2
b1
b2
2
1.5
1
0.5
0
-0.5
-1
0
10
20
30
40
50
60
70


参数估计误差
a1
a2
b1
b2
1
0.5
0
-0.5
-1
-1.5
-2
0
10
20
30
40
50
60
70

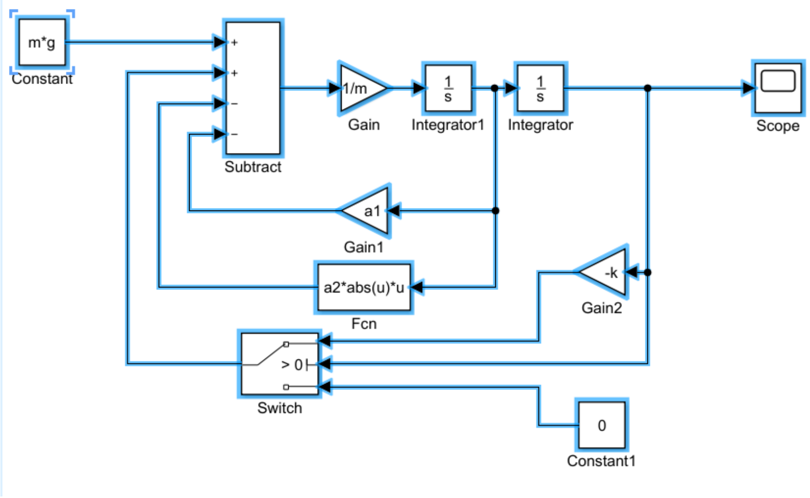
实验仿真的结果可以看出，输入为均值为 0 的 M 序列时，且在方差较小的白噪声影响下，递推最小二乘法对系统参数的估计能够取得较好的效果，在此实验中，在第 10 次迭代就已经得到较好的收敛效果，参数估计误差很快趋于 0。

八、实验结论

递推最小二乘法减少反复计算测量矩阵的逆的步骤，能根据实时产生的新数据，利用先前的计算结果不断估计参数。解决了在 n+N 组观测数据时的参数估计值已知，得到一组新的观测值 (u(n+N+1),y(n+N+ 1))，如何采用最小二乘法进行在线估计新的估计值问题。

系统建模与仿真 第四次上机实验报告 190320521 刘自涵

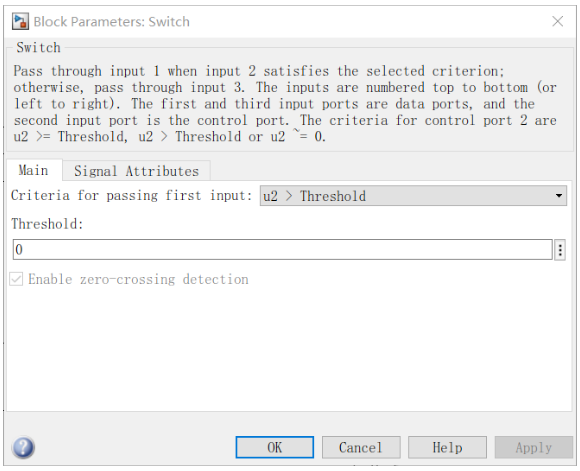
例 5



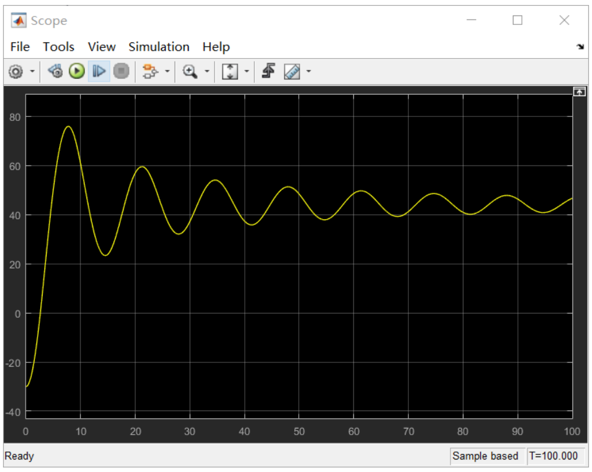
仿真截图

Nam	Value	DataType	Dimension	Complexit	Mi	Ma	Uni
a1	1	double (auto)					
a2	1	double (auto)					
g	9.800000000000001	double (auto)					
k	20	double (auto)					
m	90	double (auto)					

Model Workspace 数据设置

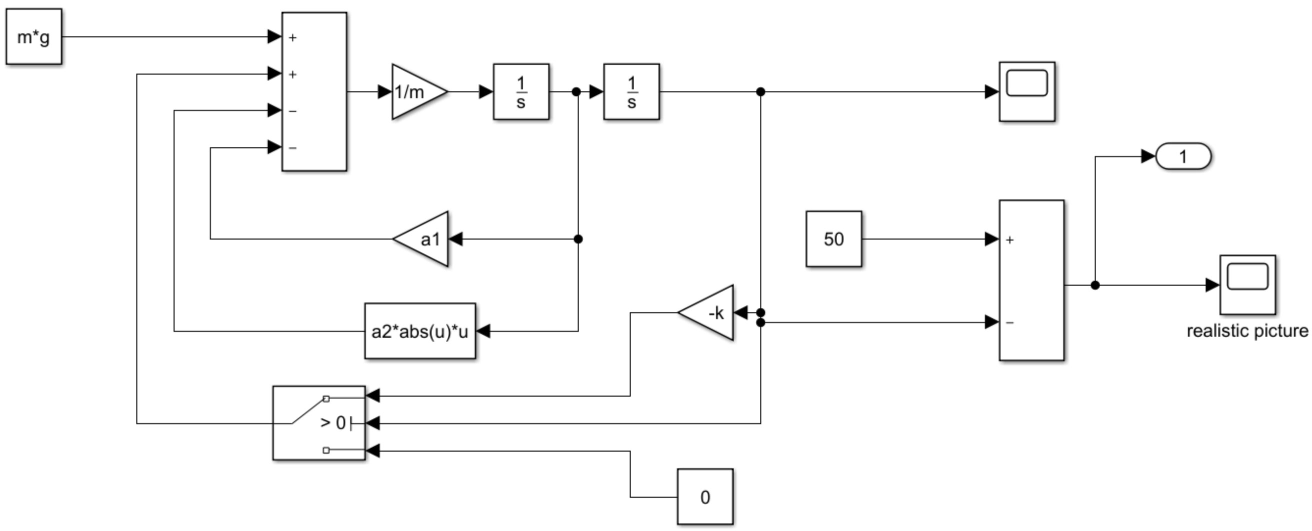


switch 开关设置



仿真结果

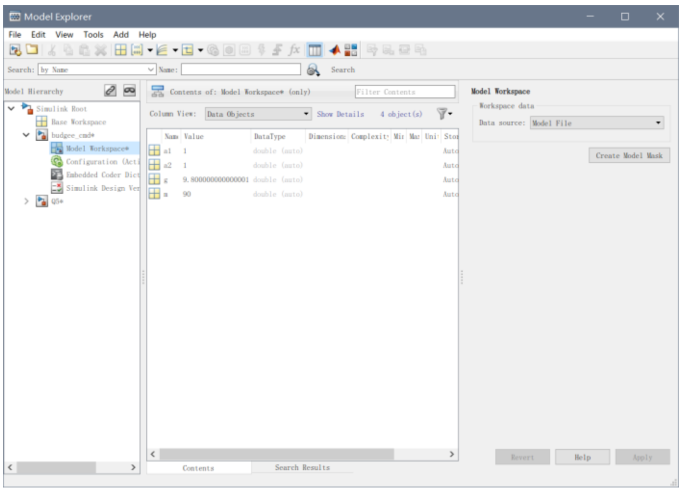
例 6



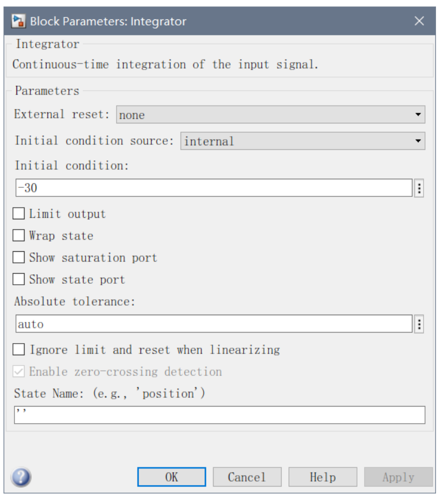
Simulink 仿真截图

```
clc;
clear;
for k=1:50
    [t,x,y]=sim('budgee_cmd');
    if min(y)>0
        break
    end
end
disp(['The minimum safe k is:',num2str(k)])
```

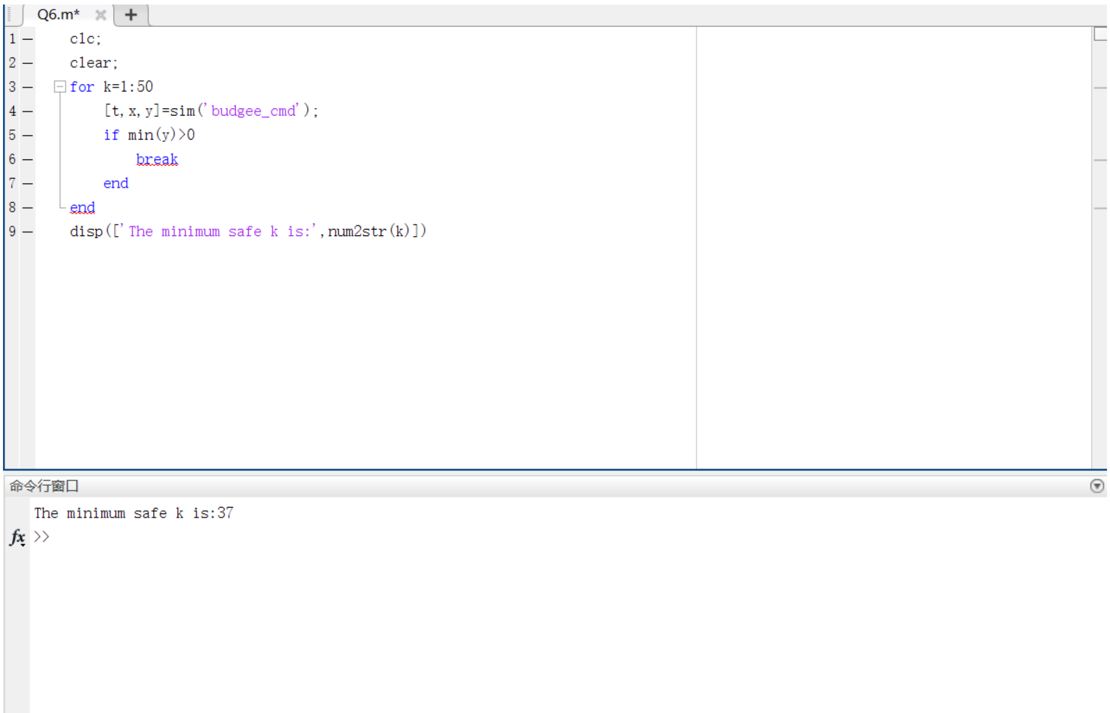
脚本文件代码



Model Workspace 数据设置

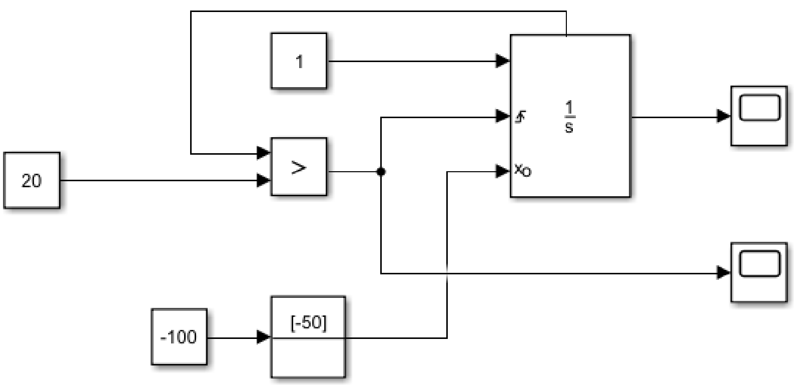


积分器初始值设置

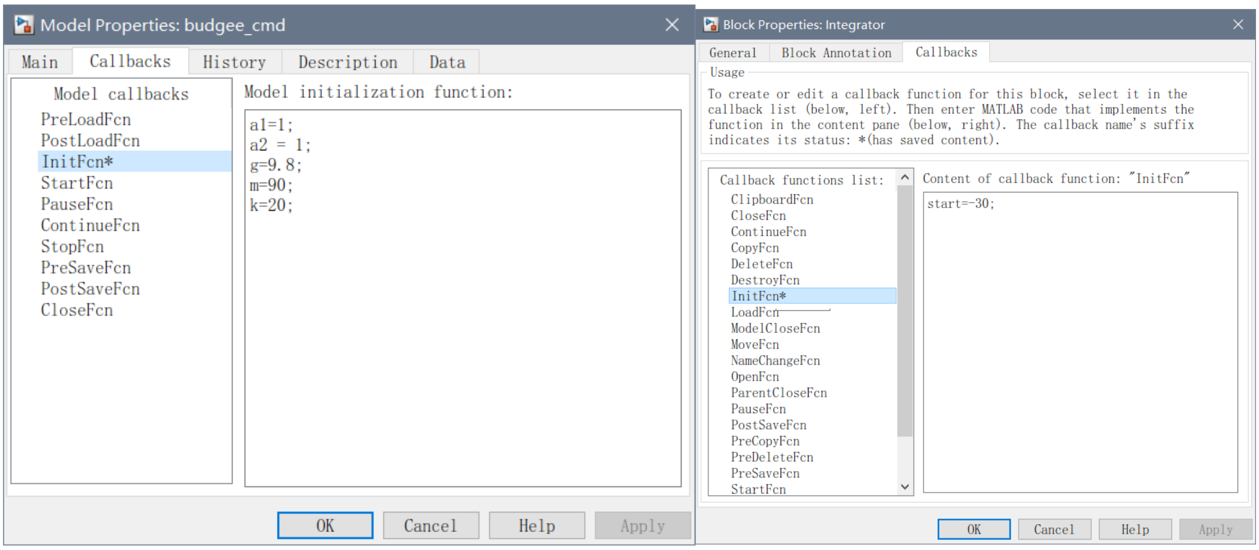


脚本运行结果

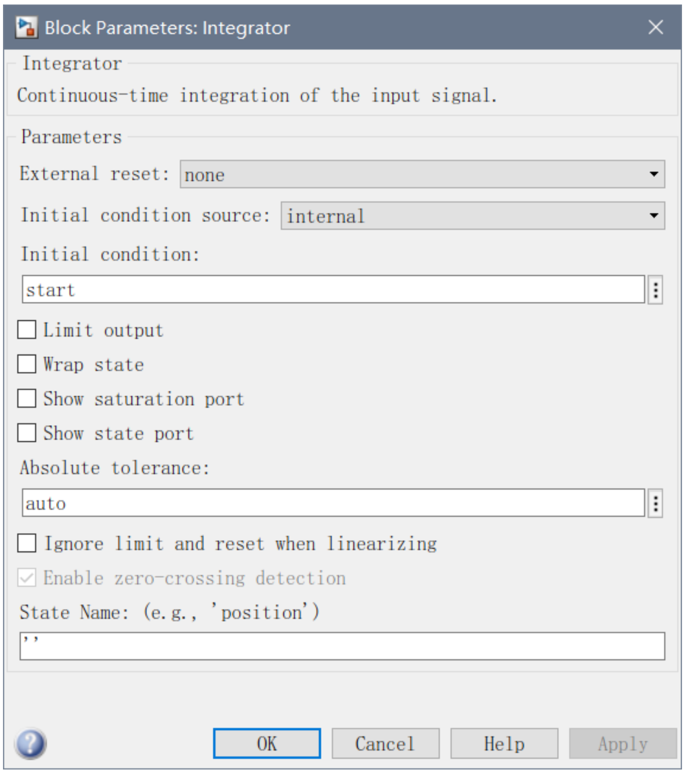
例 7 使用模型和模块的参数函数进行参数初始化



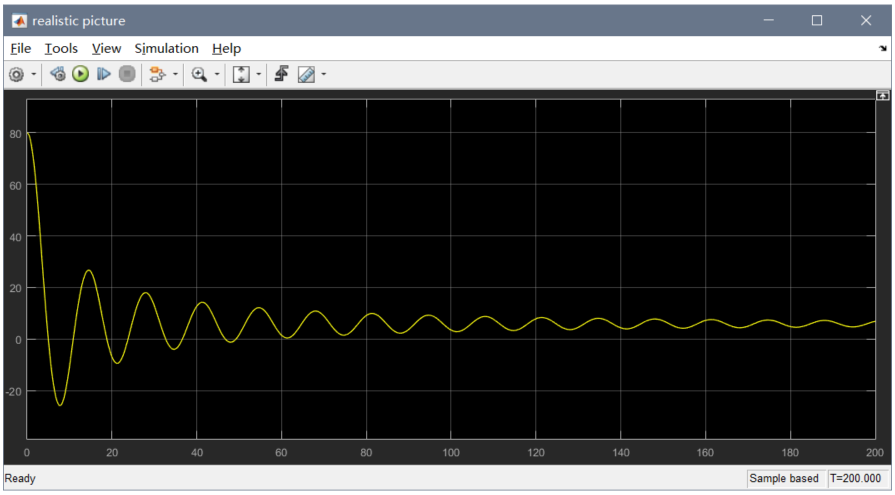
Simulink 仿真截图



模型与模块回调函数设置

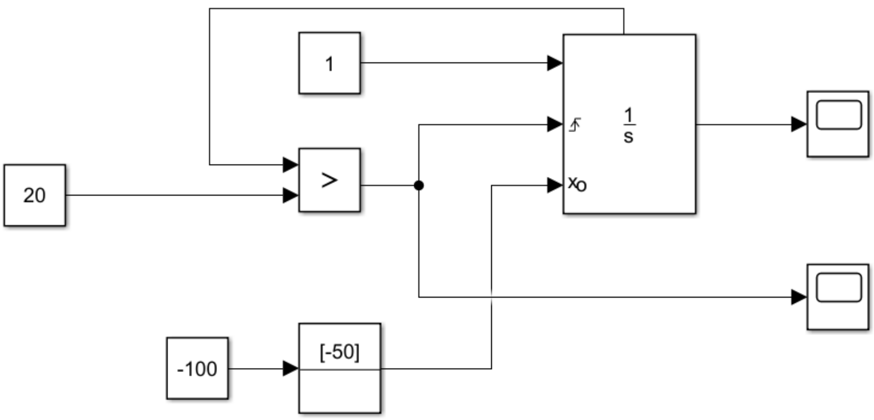


积分器对应变量名设置

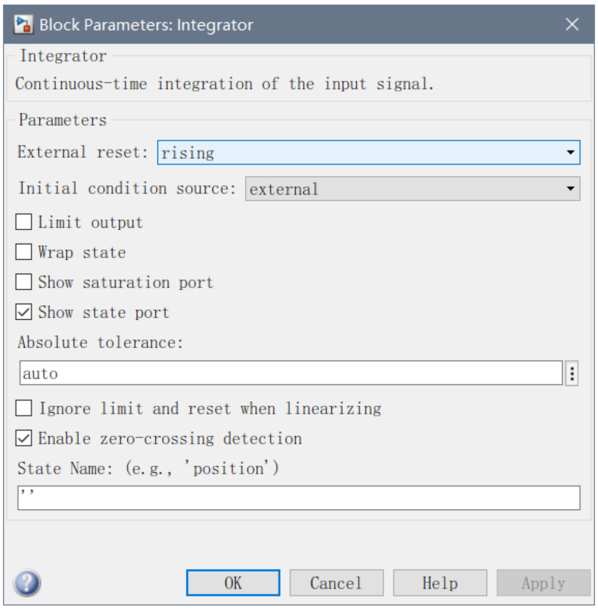


仿真运行结果

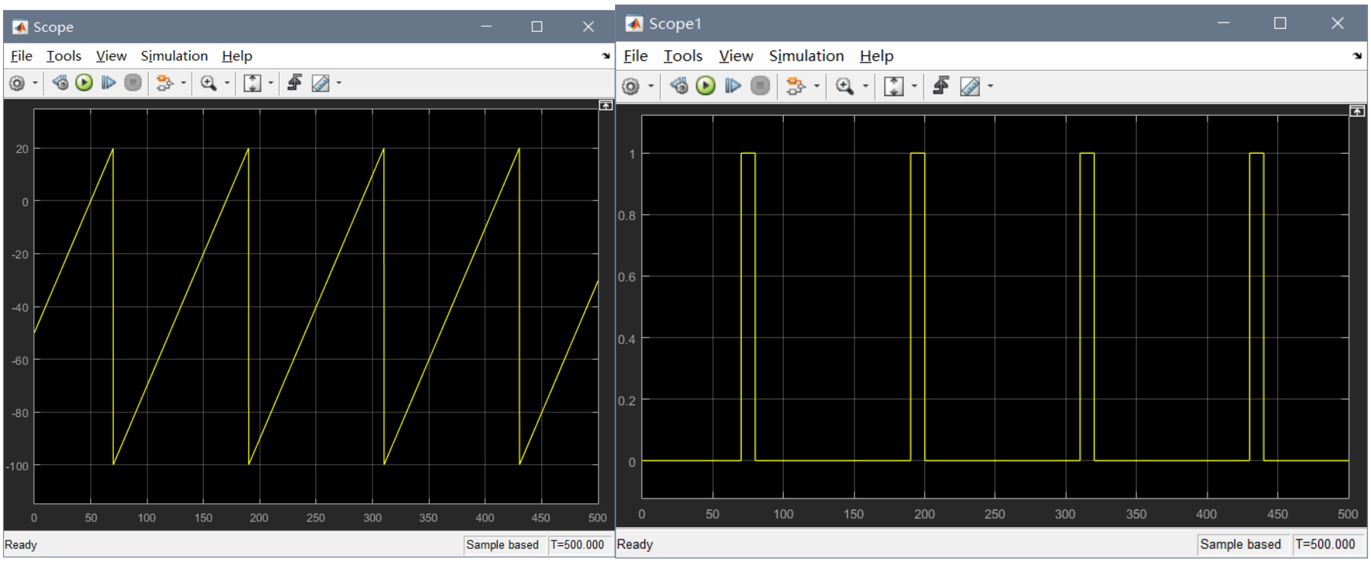
例 8



Simulink 仿真截图

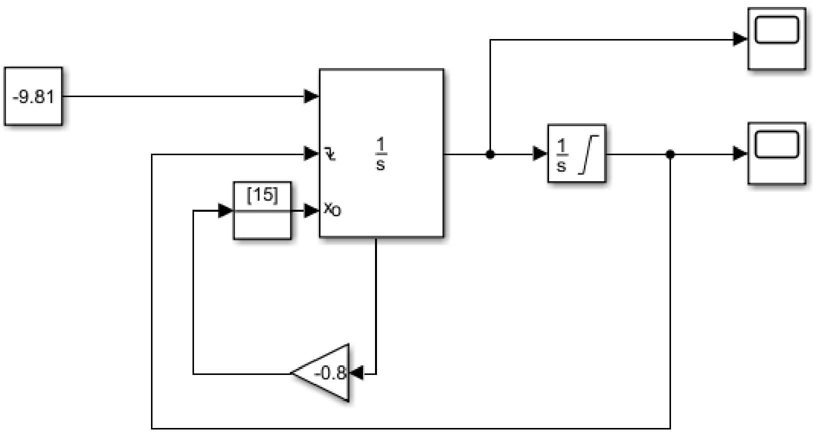


积分器相关设置

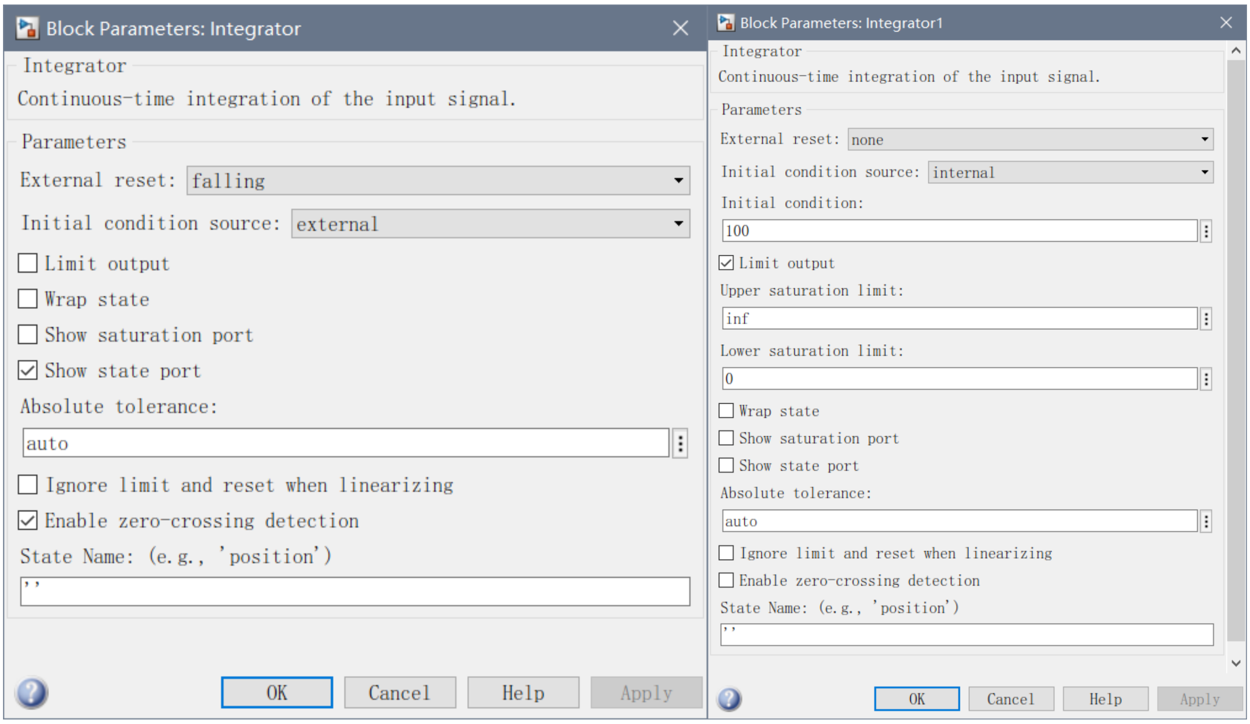


仿真运行结果

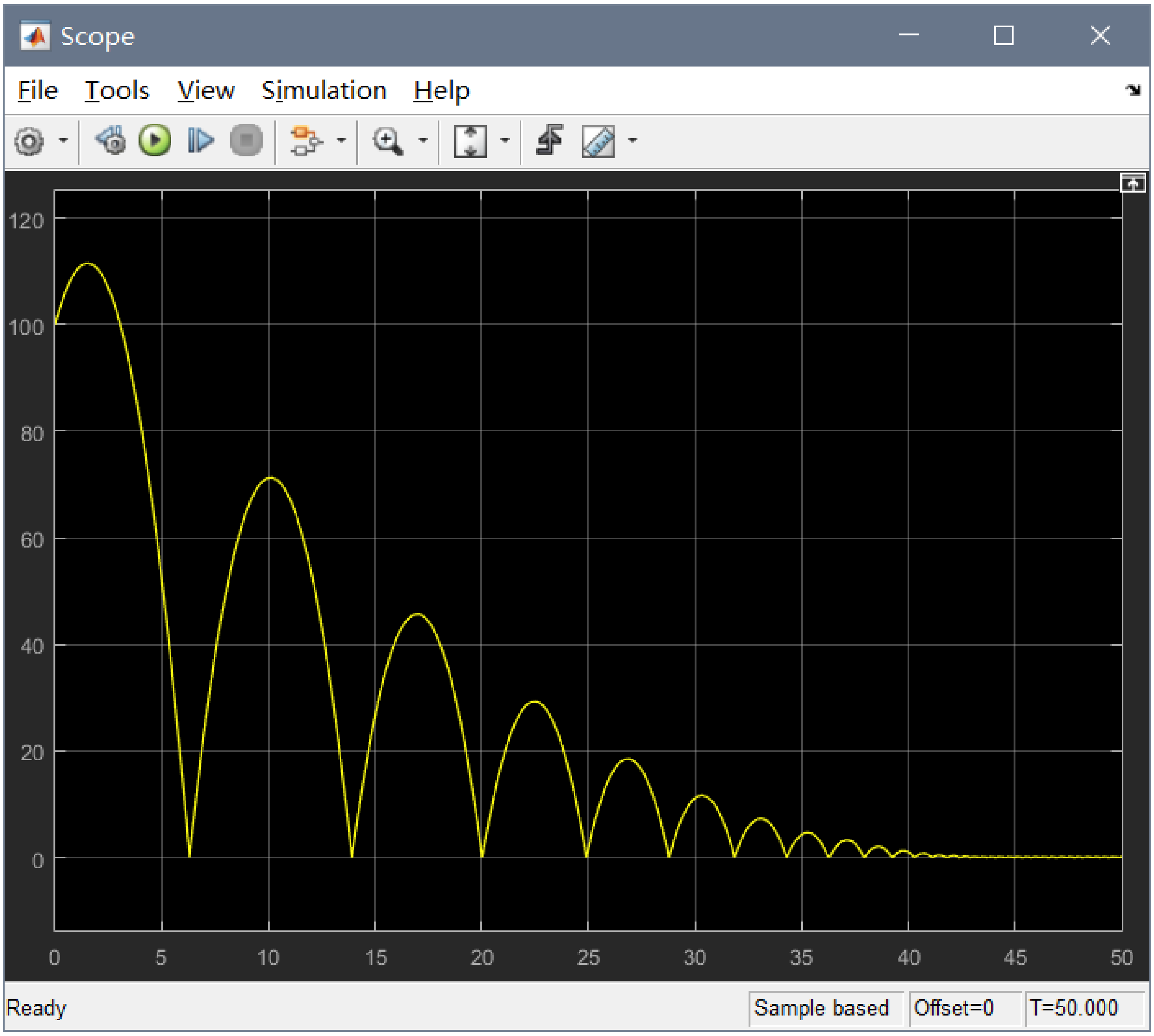
例 9



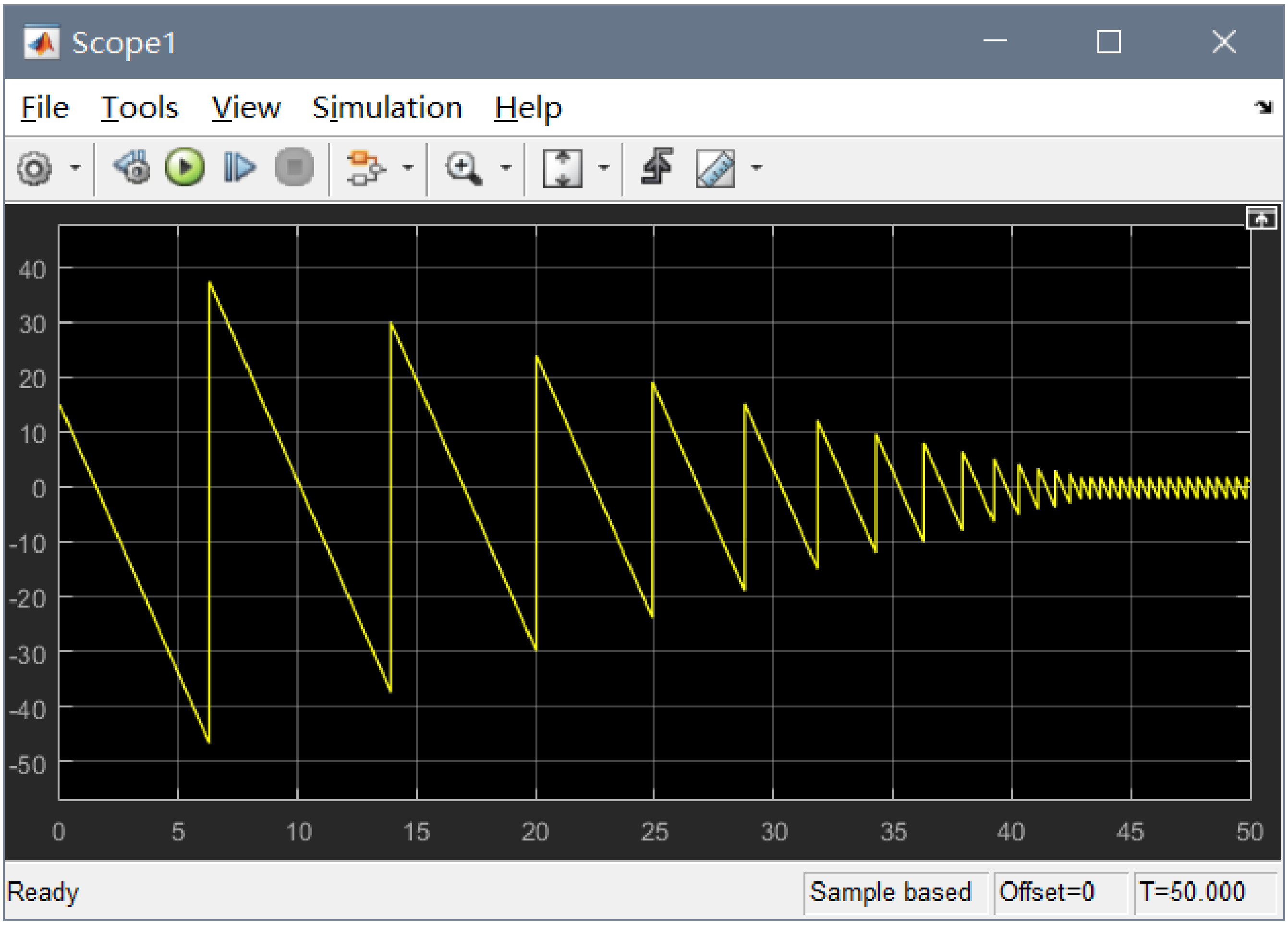
Simulink 仿真截图



积分器设置（积分器特性设置（左）以及球的初始高度（右））



小球位置仿真结果


Scope1
File Tools View Simulation Help
40
30
20
10
0
-10
-20
-30
-40
-50
0 5 10 15 20 25 30 35 40 45 50
Ready
Sample based
Offset=0
T=50.000
小球速度仿真结果