

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 $G(s)$,

$$G(s) = \frac{K}{(T_1s + 1)(T_2s + 1)}$$

其中 $K = 120, T_1 = 8.3\text{Sec}, T_2 = 6.2\text{Sec}$; $u(k)$ 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 σ_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 $k \rightarrow \infty$ 时, 应该有 $\tilde{g}(k) \rightarrow 0$ 。

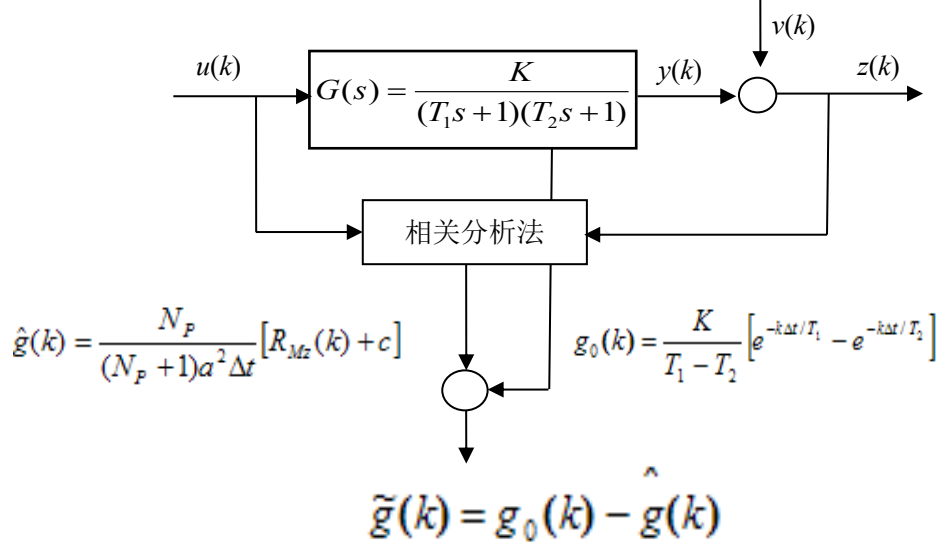


图 1 相关分析法辨识脉冲响应原理框图

三、实验要求

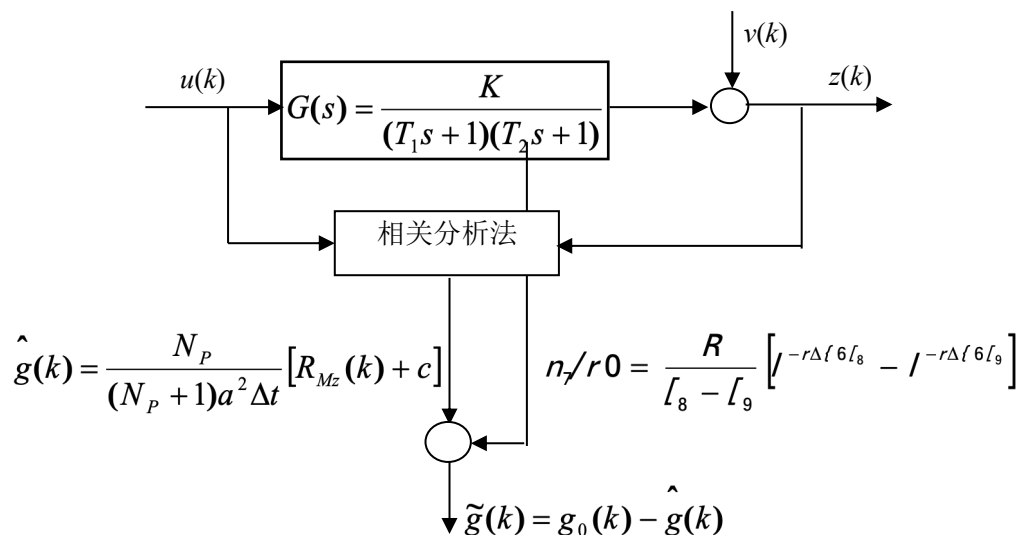
进行方案设计, 模拟过程传递函数, 获得输出数据, 用M序列作为辨识的输

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 $K=120, T_1=8.3\text{Sec}, T_2=6.2\text{Sec}$ ； $u(k)$ 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 σ_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 $k \rightarrow \infty$ 时，应该有 $\tilde{g}(k) \rightarrow 0$ 。



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 $-R_{Mz}(N_p)$ 。

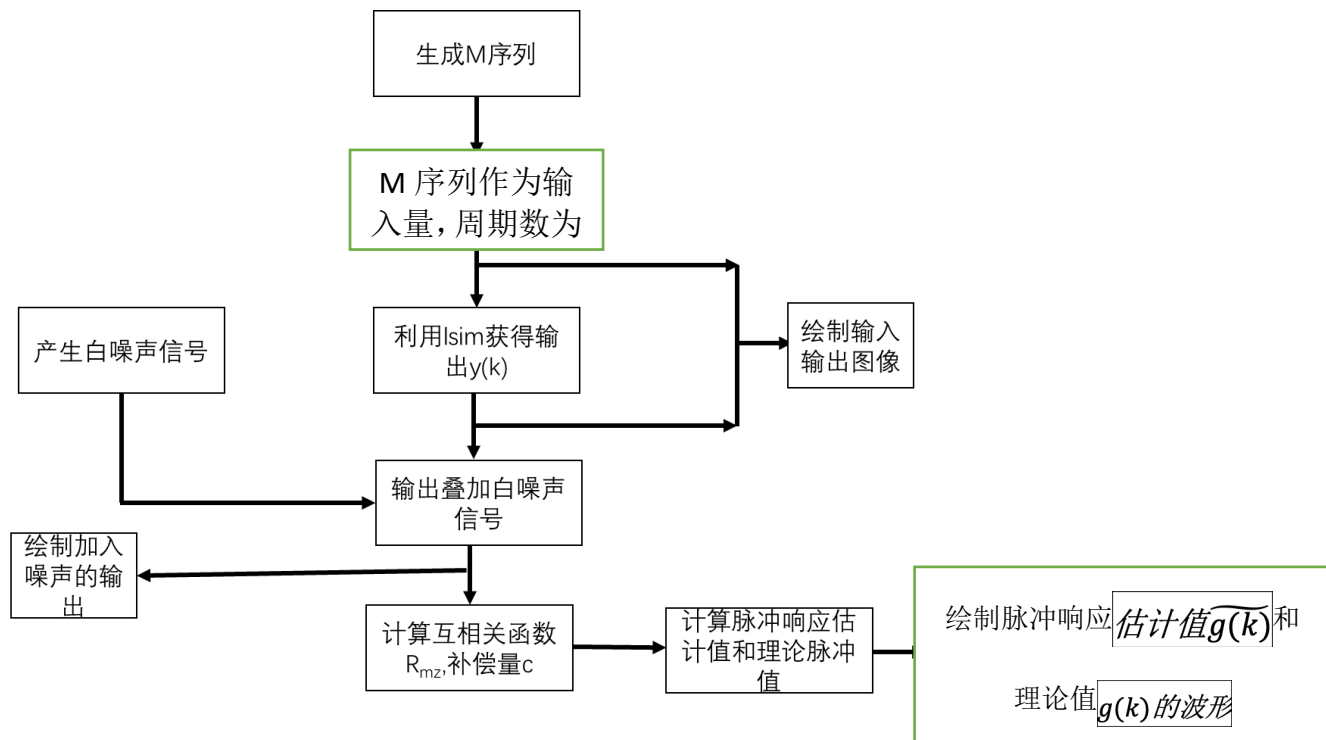
4、计算脉冲响应估计值

- 理论脉冲响应值 $n_{\gamma}/r0 = \frac{R}{l_8 - l_9} \left[l^{-r\Delta f 6 l_8} - l^{-r\Delta f 6 l_9} \right]$

- 脉冲响应估计值 $\hat{n}/r0 = \frac{U_w}{U_w + 80h^9 \Delta f} \left[Y_{T_s}/r0 + j \right]$

- 脉冲响应估计误差 $\delta_n = \sqrt{\frac{\sum_{r=8}^{U_w} (n_{\gamma}/r0 - \hat{n}/r0)^9}{\sum_{r=8}^{U_w} (n_{\gamma}/r0)^9}}$

五、实验框图



六、实验程序代码

主程序

```
% 定义 LFSR 的初始状态和反馈多项式
initial_state = [1, 0, 0, 0, 0, 0, 1, 1];
```

```

feedback_polynomial = [0, 0, 0, 1, 1, 1, 0, 1];

% 计算M序列的长度
sequence_length = 2^length(initial_state) - 1;

% 生成M序列
m_sequence = zeros(1, sequence_length);
for i = 1:sequence_length
    m_sequence(i) = initial_state(8);
    feedback = mod(sum(feedback_polynomial.*initial_state) , 2);
    initial_state = [feedback, initial_state(1:7)];
end

% M序列幅值归一化
for i = 1:sequence_length
    if m_sequence(i) == 0
        m_sequence(i) = 1;
    else
        m_sequence(i) = -1;
    end
end

% 绘制M序列
figure;
subplot(2, 2, 1);
plot(1:sequence_length, m_sequence);
xlabel('Time/s');
ylabel('Amplitude');
title('M Sequence');

% 生成输入信号
% 定义周期数
period = 5;
input = zeros(1, period * sequence_length);
for i = 1:5 * sequence_length
    input(i) = m_sequence(mod(i, sequence_length) + 1);
end
sequence_length = period * sequence_length;

% 定义传递函数
T1 = 8.3;
T2 = 6.2;
K = 120;

```

```

numerator = K;
denominator = [T1 * T2, T1 + T2, 1];

% 创建系统模型
sys = tf(numerator, denominator);

% 定义时间步长和时间向量
dt = 1; % 时间步长
t = 0:dt:(sequence_length - 1) * dt; % 时间向量

% 生成白噪声
noise = randn(1, sequence_length); % 生成服从正态分布的白噪声
disp(size(noise));
subplot(2, 2, 2);
plot(t, noise);
xlabel('Time/s');
ylabel('Amplitude');
title('White Noise');

% 使用 lsim 函数模拟系统响应
output = lsim(sys, input, t);
subplot(2, 2, 3);
plot(t, output);
xlabel('Time/s');
ylabel('Output');
title('Output without Noise');

% 添加白噪声
output = output + noise.';

% 绘制输出信号
subplot(2, 2, 4);
plot(t, output);
xlabel('Time/s');
ylabel('Output');
title('Output with Noise');
hold off;

compensate = -Correlation(length(m_sequence)-1, input, output,
length(m_sequence), period);

corr = zeros(1, length(m_sequence));
g_est = zeros(1, length(m_sequence));
g_the = zeros(1, length(m_sequence));

```

```

for i = 1:length(m_sequence)
    corr(i) = Correlation(i, input, output, length(m_sequence), period);
    g_est(i) = g_estimate(1, dt, length(m_sequence), compensate, corr(i));
    g_the(i) = g_theory(i-1, T1, T2, dt, K);
end

% 绘制脉冲响应函数估计值
figure;
plot(1:length(m_sequence), g_est, 1:length(m_sequence), g_the,
1:length(m_sequence), g_the-g_est);
xlabel('Time/s');
ylabel('Amplitude');
title('Impulse Response Function Estimate');

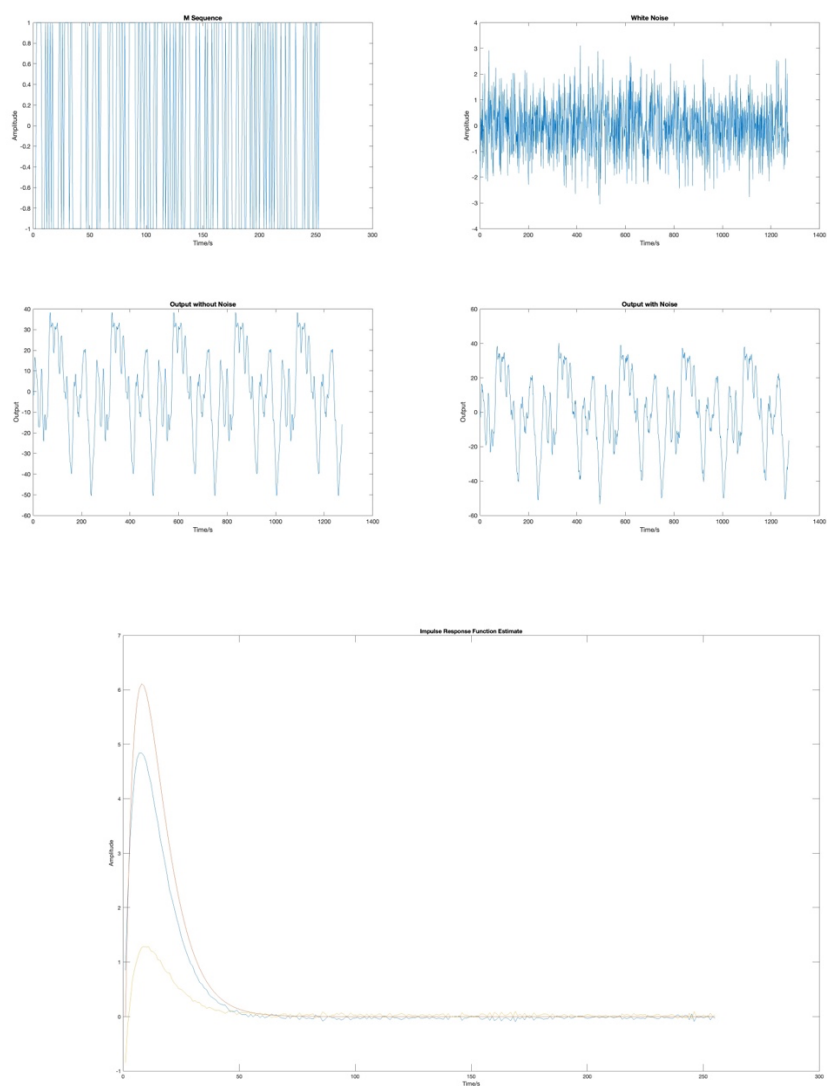
% 互相关函数
function output = Correlation(k, input1, input2, length, period) % k 为延迟, input1
为 M 序列, input2 为输出信号, length 为序列长度
    output = 0;
    denominator = period * length;
    for i = length+1:(period)*length
        output = output + input1(i-k) * input2(i);
    end
    output = output * (1 / denominator);
end

% 脉冲响应函数估计值
function output = g_estimate(amplitude, delta_t, length, compensate, corr)
    output = length * (corr + compensate) / ((length + 1) * amplitude^2 * delta_t);
end

% 脉冲响应函数理论值
function output = g_theory(k, T1, T2, delta_t, amplitude)
    output = amplitude*(1/(T1-T2)) * (exp(-k * delta_t / T1) - exp(-k * delta_t
/ T2));
end

```

七、实验结果及分析



八、实验结论

实验生成了 M 序列和白噪声，并将 M 序列作为输入使用相关法辨识系统的脉冲响应，最后一幅图为辨识结果和真实响应的关系，可以看到，两者比较接近，表明实验效果较好。