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$K=120, T_1=8.3\text{Sec}, \text{T}_2=6.2\text{Sec}$；$u(k)$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$\sigma_v^2$，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$\tilde{g}(k)\to 0$。

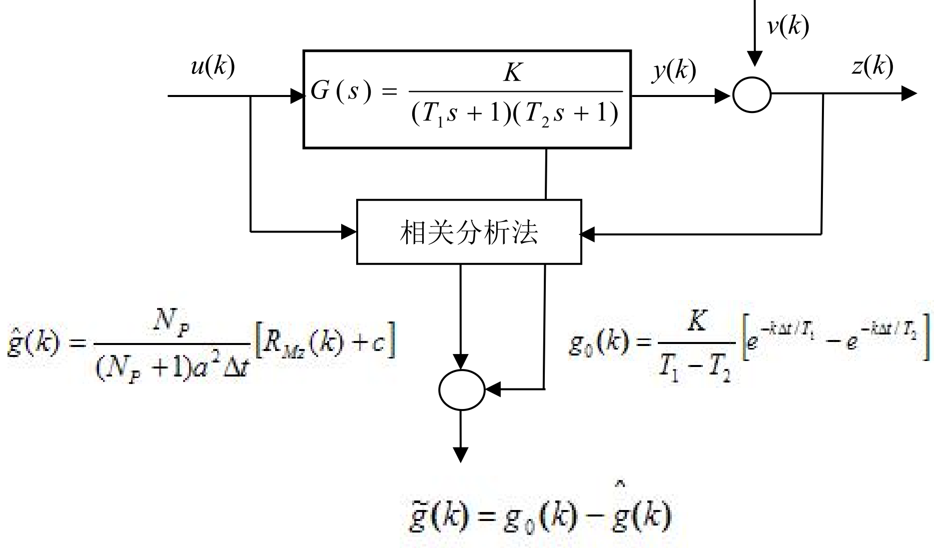


图 1 相关分析法辨识脉冲响应原理框图

三、实验要求

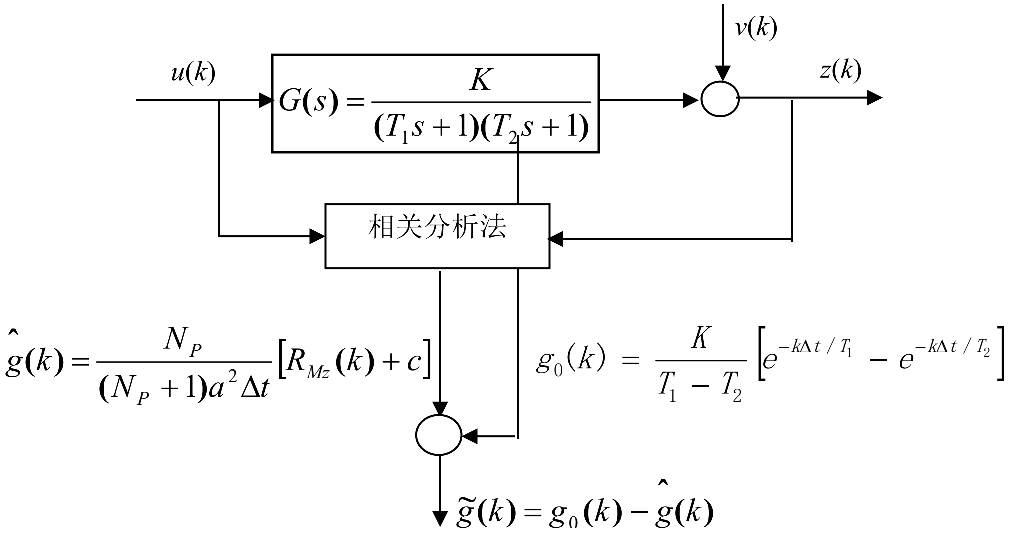
进行方案设计，模拟过程传递函数，获得输出数据，用M序列作为辨识的输

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）

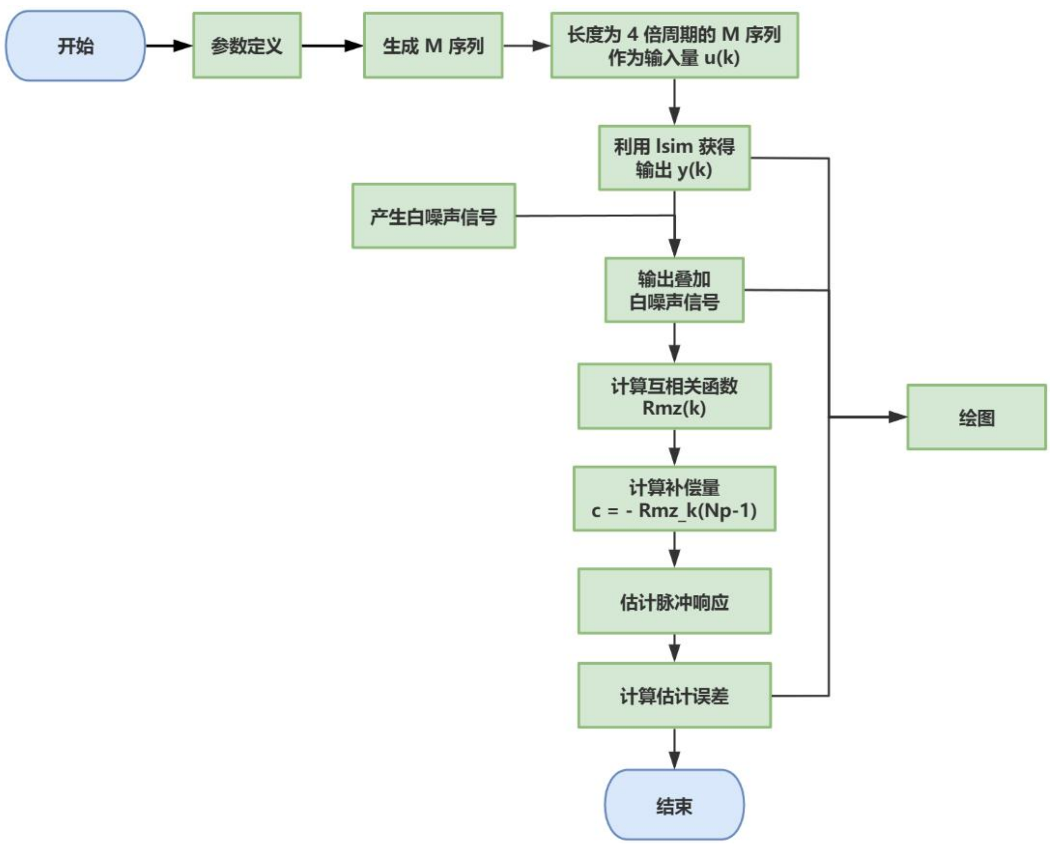
3、 补偿量 c

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

参数定义

```
% 参数定义
register_length = 9; % 寄存器长度
M_init = [1, 1, 0, 0, 0, 1, 0, 0, 1]; % 初始寄存器状态
r = 3; % 生成的 M 序列长度是其周期的 (r+1) 倍
A = 1; % M 序列幅度
T1 = 8.3; T2 = 6.2; K = 120; % 系统参数
delta_t = 1; % 采样间隔
```

函数定义

```
% 生成 M 序列
function u = generate_m_sequence(M_init, r, A)
    Np = 2 ^ length(M_init) - 1; % M 序列周期
    Num = (r + 1) * Np; % M 序列长度
    u = zeros(1, Num);
    M = M_init;

    for i = 1:Num
        u(i) = M(length(M_init)); % 取出最后一位作为当前输出
        m = xor(M(length(M_init)), M(5)); % 异或生成反馈位
        M = [m M(1:length(M_init) - 1)]; % 移位操作
    end

    u = A * (1 - 2 * u); % 值域为 (-A, A) 的 M 序列
end
```

```
% 生成白噪声
function noise = generate_white_noise(len, mean, stddev)
    % 通过统计近似抽样法，利用均匀分布随机数生成正态分布随机数

    n = 40; % 生成的(0,1)均匀分布随机数的数量

    % 生成标准正态分布的随机数
    uniform_samples = rand(len, n); % 生成 len 行 n 列的均匀分布随机数
    normal_samples = sum(uniform_samples, 2) - n / 2; % 累加后减去期望 n/2
    normal_samples = normal_samples / sqrt(n / 12); % 归一化，方差为 1
    noise = normal_samples * stddev + mean; % 乘以标准差并加上均值
    noise = noise'; % 转置为行向量
end
```

```
% 获取系统响应
function Y = get_system_response(u, T1, T2, K)
    Gs = tf(K, [T1 * T2, T1 + T2, 1]); % 系统传递函数
    tt = 0:length(u) - 1; % 时间向量
    Y = lsim(Gs, u, tt)'; % 计算系统响应并转置为列向量
end
```

```
% 估计脉冲响应 g_theoretical、g_estimated、g_error 分别为理论脉冲响应 g0_k、估计脉冲响应 ghat_k、误差值 delta_g_k
function [g_theoretical, g_estimated, g_error] = estimate_impulse_response(u, Y_noise, T1, T2, K, Np, A, delta_t, r)
    % 计算理论脉冲响应
    time = 0:delta_t:(Np - 1) * delta_t;
    g_theoretical = (K / (T1 - T2)) * (exp(-time ./ T1) - exp(-time ./ T2));

    % 计算互相关系数 Rmz_k
    Rmz_k = zeros(1, Np);

    for tch = 1:Np
        SUM = 0;

        for tcl = Np + 1:(r + 1) * Np
            SUM = SUM + u(tcl - tch) * Y_noise(tcl);
        end

        Rmz_k(tch) = SUM / (r * Np);
    end

    % 估计脉冲响应 ghat_k, 补偿量 c = -Rmz_k(Np-1)
    g_estimated = Np * (Rmz_k - Rmz_k(Np - 1)) / ((A ^ 2) * (Np + 1) * delta_t);

    % 误差值计算
    g_error = g_theoretical - g_estimated;
end
```

```
% 计算估计误差
function sigma_g = calculate_estimation_error(g_theoretical, g_estimated)
    error_sum = sum((g_theoretical - g_estimated) .^ 2);
    square_sum = sum(g_theoretical .^ 2);
    sigma_g = sqrt(error_sum / square_sum); % 均方根误差
end
```

主程序

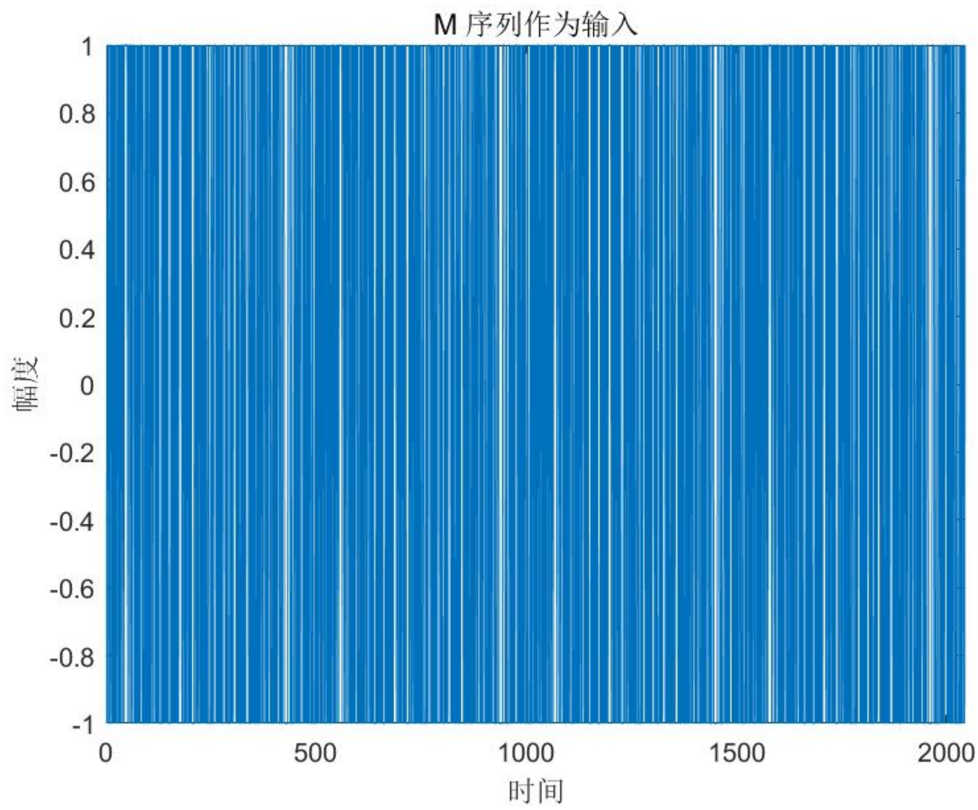
```
u = generate_m_sequence(M_init, r, A); % 生成 M 序列

figure;
plot(u);
title("M 序列作为输入");
xlim([0, length(u)]);
xlabel('时间');
ylabel('幅度');

Np = 2 ^ length(M_init) - 1; % M 序列周期
Num = (r + 1) * Np;

% 生成白噪声
noise = generate_white_noise(length(u), 0, 1); % 均值 0, 方差 1

figure;
histogram(noise, 50);
title('高斯白噪声分布');
xlabel('噪声值');
ylabel('频率');
```

```
% 获取系统输出
Y = get_system_response(u, T1, T2, K);
Y_noise = Y + noise; % 带有噪声的系统响应

figure;
hold on;
plot(noise, 'g');
plot(Y, 'r');
plot(Y_noise, 'b');
legend(["白噪声", "系统响应", "带噪声的系统响应"]);
title('系统响应及噪声');
xlim([0, length(Y)]);
xlabel('时间');
ylabel('幅度');
hold off;

% 估计脉冲响应
[g_theoretical, g_estimated, g_error] = estimate_impulse_response(u, Y_noise, T1, T2, K, Np, A, delta_t, r);

% 绘制脉冲响应结果
figure;
hold on;
plot(1:Np, g_theoretical, 'r');
plot(1:Np, g_estimated, 'b');
plot(1:Np, g_error);
title("脉冲响应估计结果");
legend(["理论值", "估计值", "误差值"]);
xlim([0, Np]);
hold off;
```

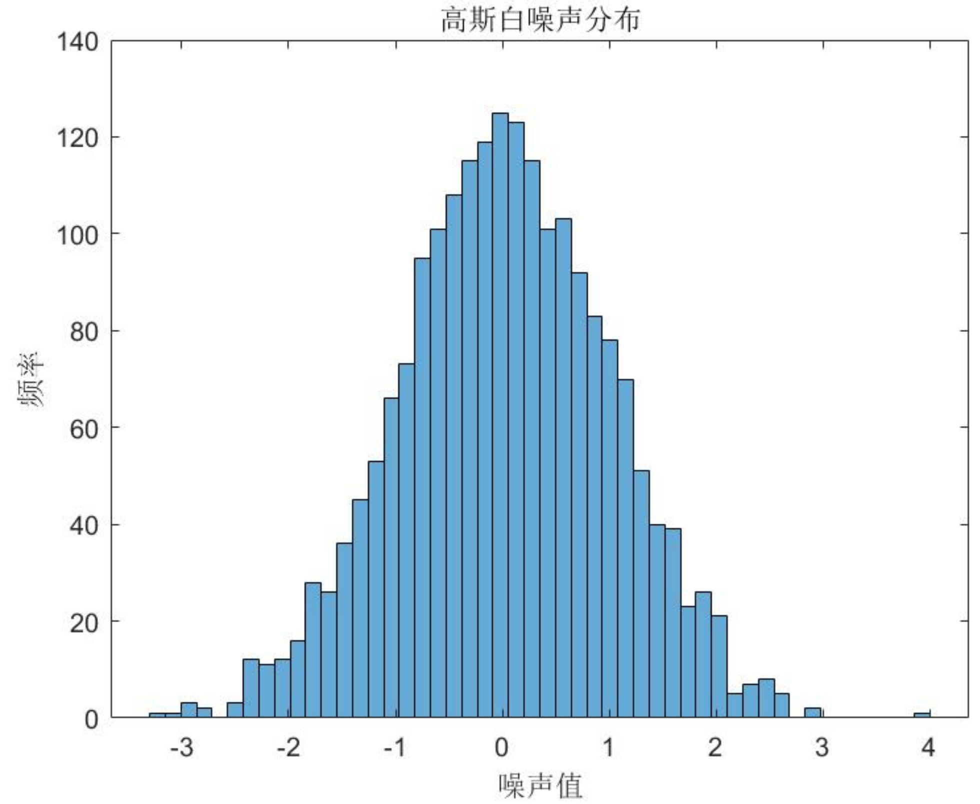
```
% 调用计算误差的函数
sigma_g = calculate_estimation_error(g_theoretical, g_estimated);
disp(['脉冲响应估计误差为: ', num2str(sigma_g)]);
```

七、实验结果及分析

1、输入的 M 序列，其长度为周期的 4 倍：

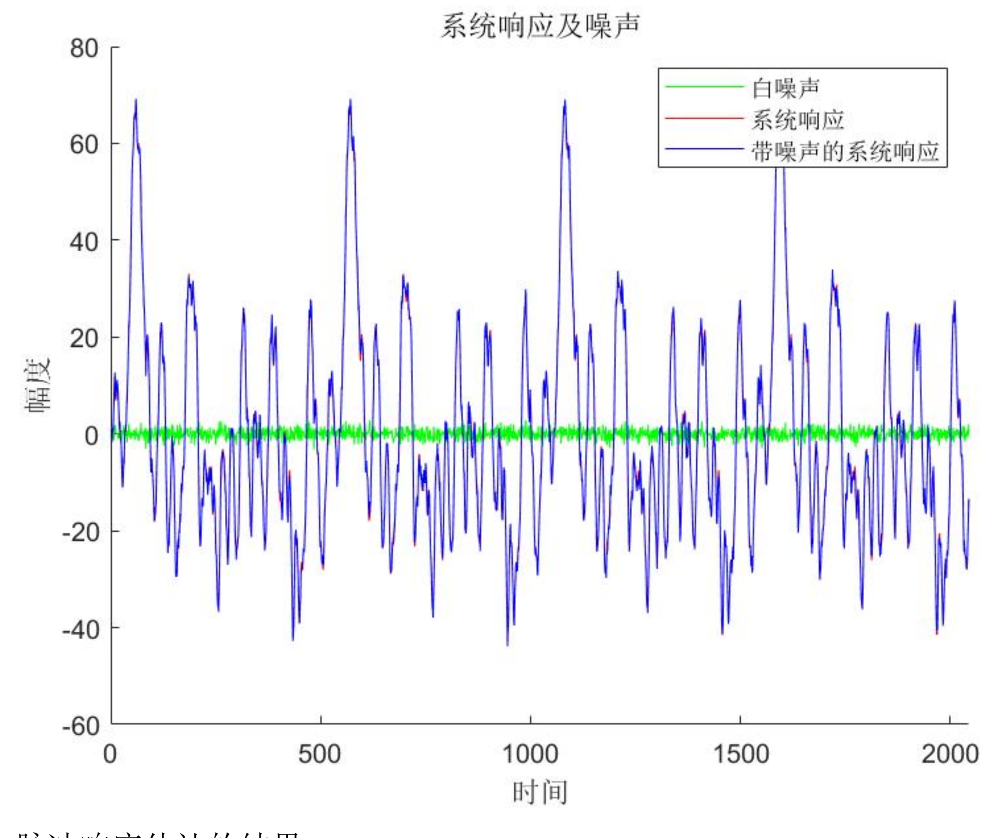


2、使用统计近似抽样法产生的高斯白噪声：

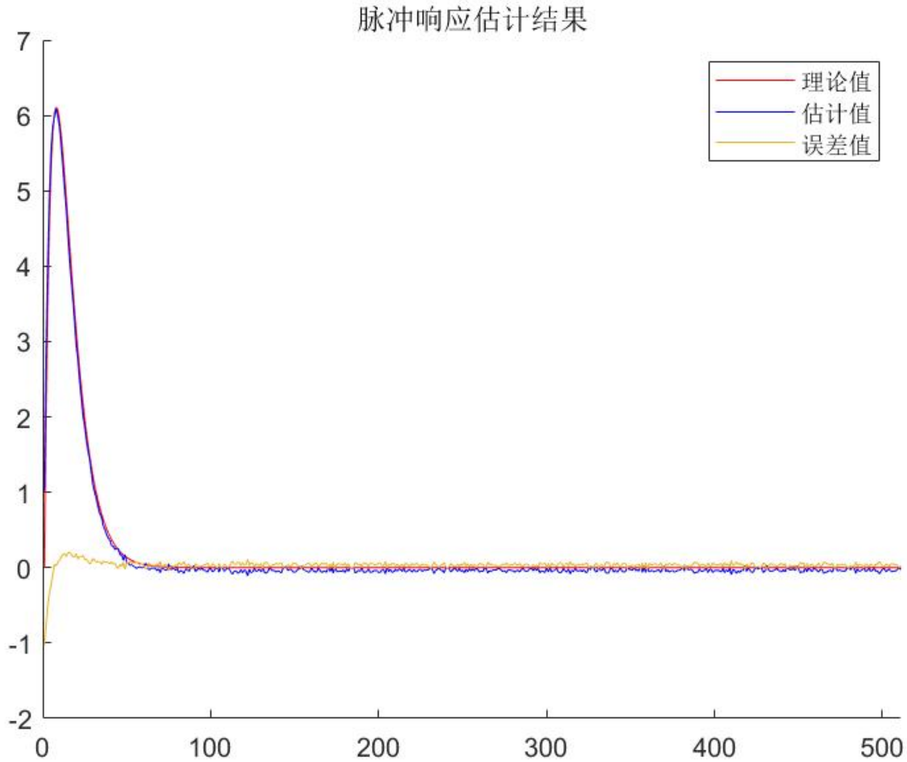


这张图说明使用统计近似抽样法产生的高斯白噪声很好地符合高斯分布的特性。

3、系统响应以及噪声：



4、脉冲响应估计的结果：



从绘图结果中可以观察到，脉冲响应估计值与理论值之间存在较好的重合。这表明通过所采用的估计方法，能够有效地捕捉到系统的动态特性。并且当 k→∞ 时，估计误差趋近于零。

5、实验十次并取平均值：

脉冲响应估计误差为：0.077441

八、实验结论

实验生成了 M 序列以及满足标准正态分布的白噪声，并将 M 序列作为输入，使用相关分析法辨识系统的脉冲响应，利用维纳霍夫方程辨识得到脉冲响应的估计值。

通过与公式得到的脉冲响应理论值相比，计算得到脉冲估计误差为 0.077441。这一数值表明理论模型与估计模型之间存在一定程度的差异。

本次实验成功估计了系统的脉冲响应，并计算了其误差。虽然存在一定误差，但所采用的方法和模型在理论上是有效的，能够为后续的研究和应用提供基础。