

基于 MATLAB 的 IIR 数字带通滤波器的设计

设计一:

利用冲击响应不变法设计一个切比雪夫带通滤波器,设计的指标为:

通带边缘频率 $w1 = 0.4\pi, w2 = 0.8\pi$

阻带边缘频率 $w3 = 0.3\pi, w4 = 1.0\pi$

通带波纹 $= 0.5\text{dB}$

阻带衰减 $= 20\text{ dB}$

解: 1. 基本原理

滤波器的传递函数的一般形式为:

$$H(z) = \frac{\sum_{k=0}^M b_k z^{-k}}{1 + \sum_{k=1}^N a_k z^{-k}} = k \frac{\prod_{r=1}^M (z - z_r)}{\prod_{k=1}^N (z - z_p)}$$

■ 当 $M \leq N$, N 阶 IIR 系统; $M > N$, $H(z)$: N 阶 IIR 系统 + $(M-N)$ 阶的 FIR 系统,

■ 以上两种表示等价, 部分分式形式和零极点增益形式

■ IIR 系统的逼近, 就是找到滤波器的系数 a_k, b_k , 或者是系统的零极点和增益 (z, p, k) 。

切比雪夫 I 型的幅度平方函数为:

$$|H(j\Omega)|^2 = \frac{1}{1 + \varepsilon^2 C_N^2\left(\frac{\Omega}{\Omega_c}\right)}$$

$H(j\Omega)$ 的特点如下:

(1) 当 $\Omega=0$ 时, N 为偶数时, $|H(j0)| = \frac{1}{\sqrt{1+\varepsilon^2}}$, 当 N 为奇数时, $|H(j0)|$

$=0$.

(2) 当 $\Omega=\Omega_c$ 时, $|H(j\Omega)| = \frac{1}{\sqrt{1+\varepsilon^2}}$

即此时所有的幅度函数曲线都经过 $\frac{1}{\sqrt{1+\varepsilon^2}}$ 点, Ω_c 即为切比雪夫

滤波器的通带截止频率.

(3) 在通带内, $|H(j\Omega)|$ 在 $1 \pm \frac{1}{\sqrt{1+\varepsilon^2}}$ 之间等波纹地起伏。

(4) 在通带外, 随着 Ω 的增大, $|H(j\Omega)|$ 迅速单调的趋近于零。

该滤波器在通带内具有等波纹起伏特性, 在阻带内则单调下降且具有更大的衰减. 相比于巴特沃斯滤波器, 阶数 N 较小.

2 冲击响应不变法

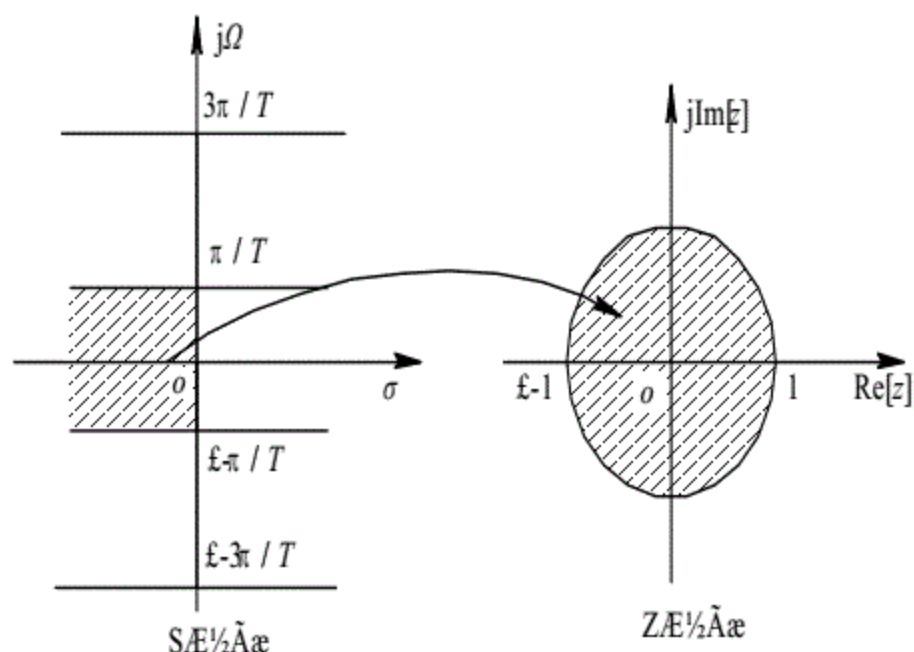
冲击响应不变法是使数字滤波器的单位冲击响应序列 $h(n)$ 模仿模拟滤波器的单位冲击响应 $h_a(t)$ 。将模拟滤波器的单位冲击响应加以等间隔抽样, 使 $h(n)$ 正好等于 $h_a(t)$ 的抽样值, 即满足:

$$h(n) = h_a(nT)$$

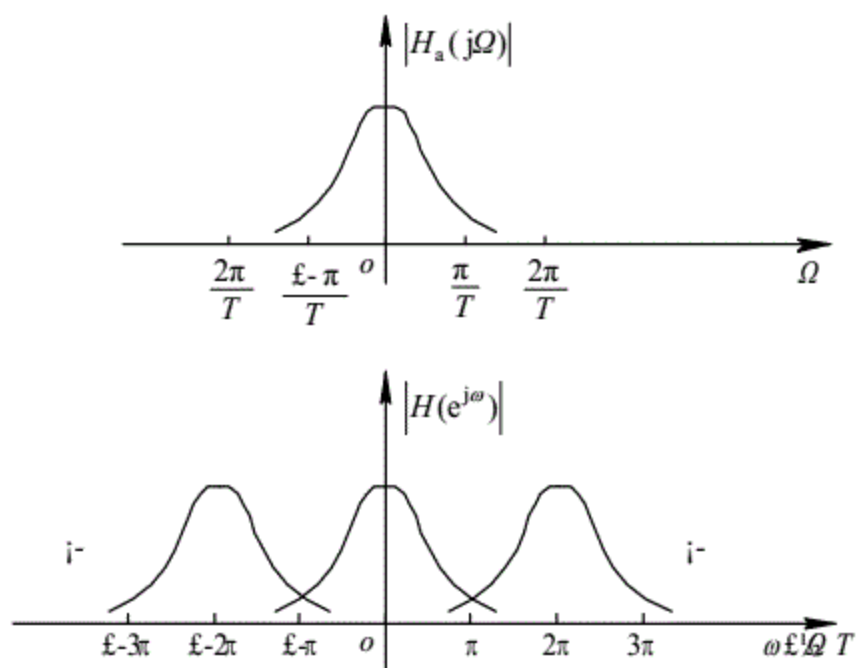
既有: $H(z)|_{z=e^{sT}} = \frac{1}{T} \sum_{k=-\infty}^{\infty} H_a(s-j\frac{2\pi}{T}k)$

冲击响应不变法是将模拟滤波器的 s 平面变换成数字滤波器的 z 平面。冲击响应不变法使得数字滤波器的单位脉冲响应完全模仿模拟滤波器的单位冲激响应, 也就是时域逼近良好。一个线性相位的模拟

滤波器通过脉冲响应不变法得到的仍然是一个线性相位的数字滤波器。

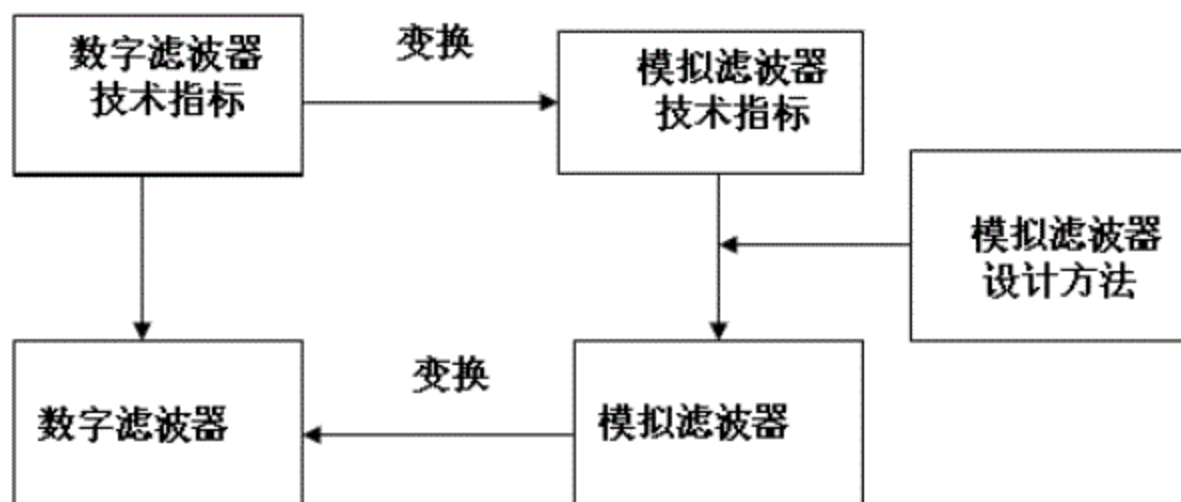


但是，由于任何一个实际的模拟滤波器频率响应都不是严格限带的，变换后就会产生频率响应的混叠失真。这时数字滤波器的频响就不同于原模拟滤波器的频响，而带有一定的失真。当模拟滤波器的频率响应在折叠频率以上处衰减越大、越快时，变换后频率响应混叠失真就越小。这时，采用脉冲响应不变法设计的数字滤波器才能得到良好的效果。



3. 设计流程:

- (1) 确定数字滤波器的技术指标。
- (2) 将数字滤波器的技术指标转换成相应的模拟滤波器的技术指标。
- (3) 按照模拟低通滤波器的技术指标设计过渡模拟低通滤波器。
- (4) 用脉冲响应不变法，将模拟滤波转换成数字滤波器



4. matlab 实现主要步骤:

(1)先由数字频率转化为模拟的频率,由于采用脉冲响应不变法,所以:

$$\Omega_{\text{omega1}} = f_s * w_1; \quad \Omega_{\text{omega2}} = f_s * w_2;$$

$$\Omega_{\text{omega3}} = f_s * w_3; \quad \Omega_{\text{omega4}} = f_s * w_4;$$

$$\text{采样频率 } f_s = 2000\text{HZ}$$

由函数

$$[N, W_n] = \text{cheblord}(W_p, W_s, R_p, R_s, 's');$$

$$[z, p, k] = \text{cheblap}(N, R_p); \quad \text{求出模拟低通滤波器。}$$

(2)再通过 $[bt, at] = \text{lp2bp}(bb, aa, W_o, B_w)$ 函数实现由模拟低通滤波器向带通滤波器的转换.

$$\text{其中: } W_o = \sqrt{\Omega_{\text{omega1}} * \Omega_{\text{omega2}}};$$

$$B_w = \Omega_{\text{omega2}} - \Omega_{\text{omega1}};$$

进行频率变换后实现了带通模拟滤波器

(3): 采用脉冲响应不变法:利用函数:

$$[bz, az] = \text{impinvar}(bt, at, f_s)$$

将上述带通模拟滤波器转换为数字带通滤波器.

其中 f_s 为采样频率

(4):验证:

预期输出信号:

$$u_zs = \sin(2 * \pi * 451 * nn)$$

加入噪声后的信号为：

$$uz = 0.5 \cos(2\pi \cdot 21 \cdot nn) + uzs + 2 \sin(2\pi \cdot 1000 \cdot nn)$$

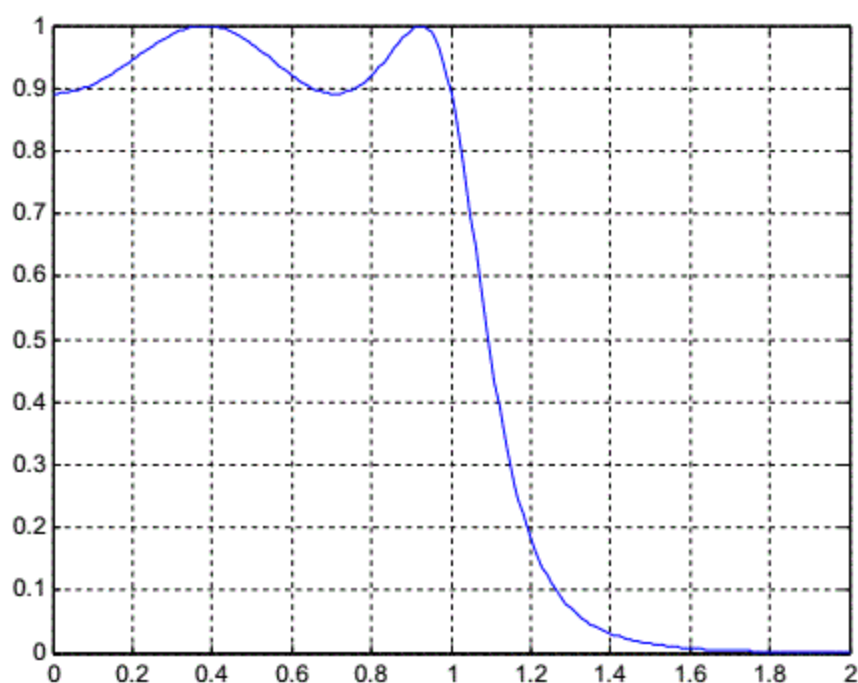
观察经过该滤波器后的输出波形。

其中 $F_n = 100$;

$$dn = 1/fs; n = 0:F_n-1; \quad nn = n \cdot dn;$$

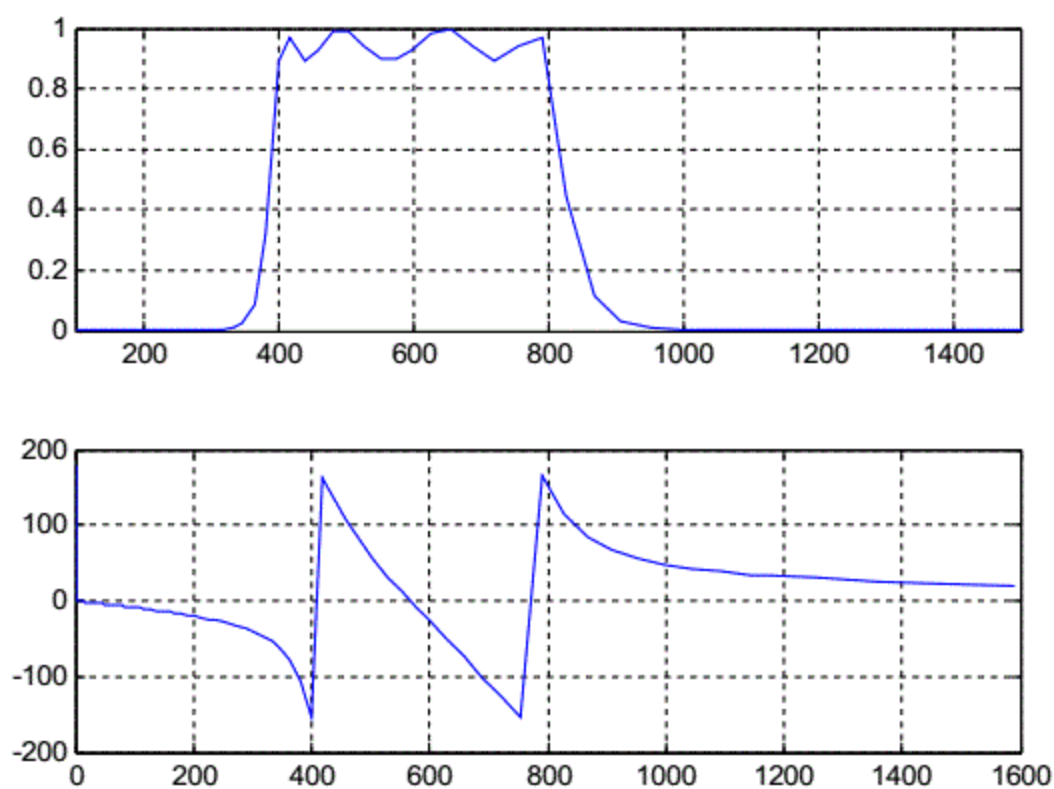
5. matlab 实现结果：

(1) 根据滤波器的设计指标, 首先实现一个模拟的低通滤波器, 其曲线如图一所示:



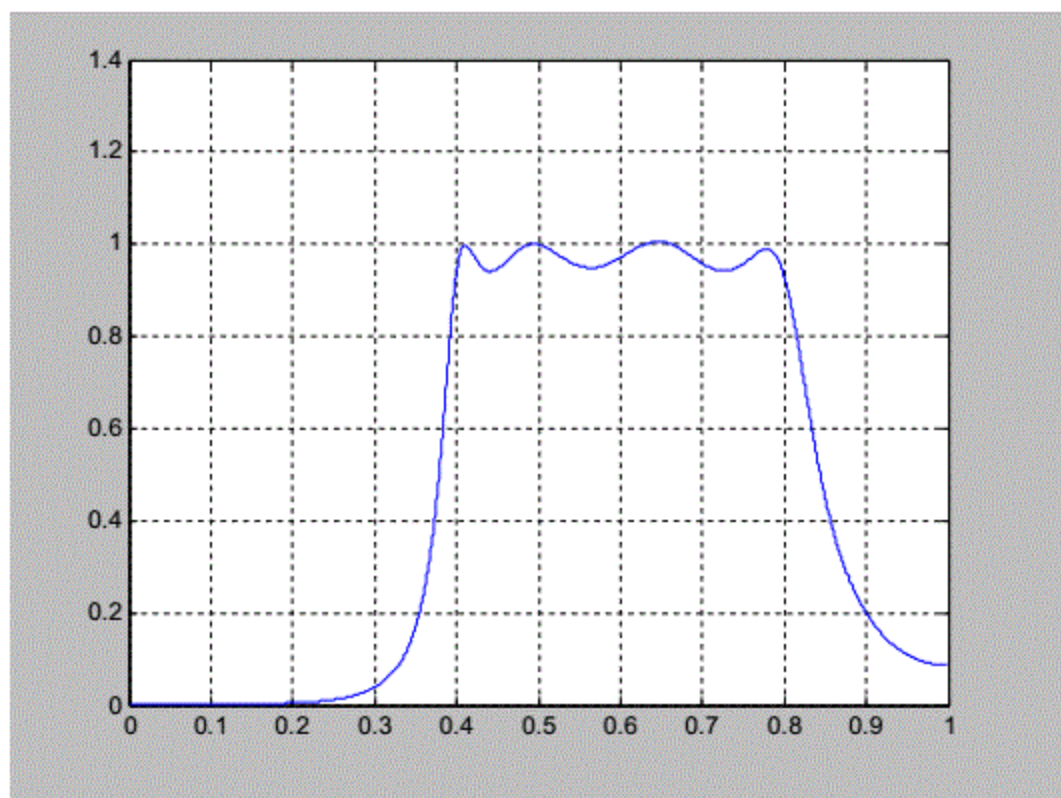
图一：模拟低通滤波器

(2) 将设计的模拟低通滤波器进行频率变换, 变换成模拟的带通滤波器, 其幅频, 相频曲线如图 2 所示:

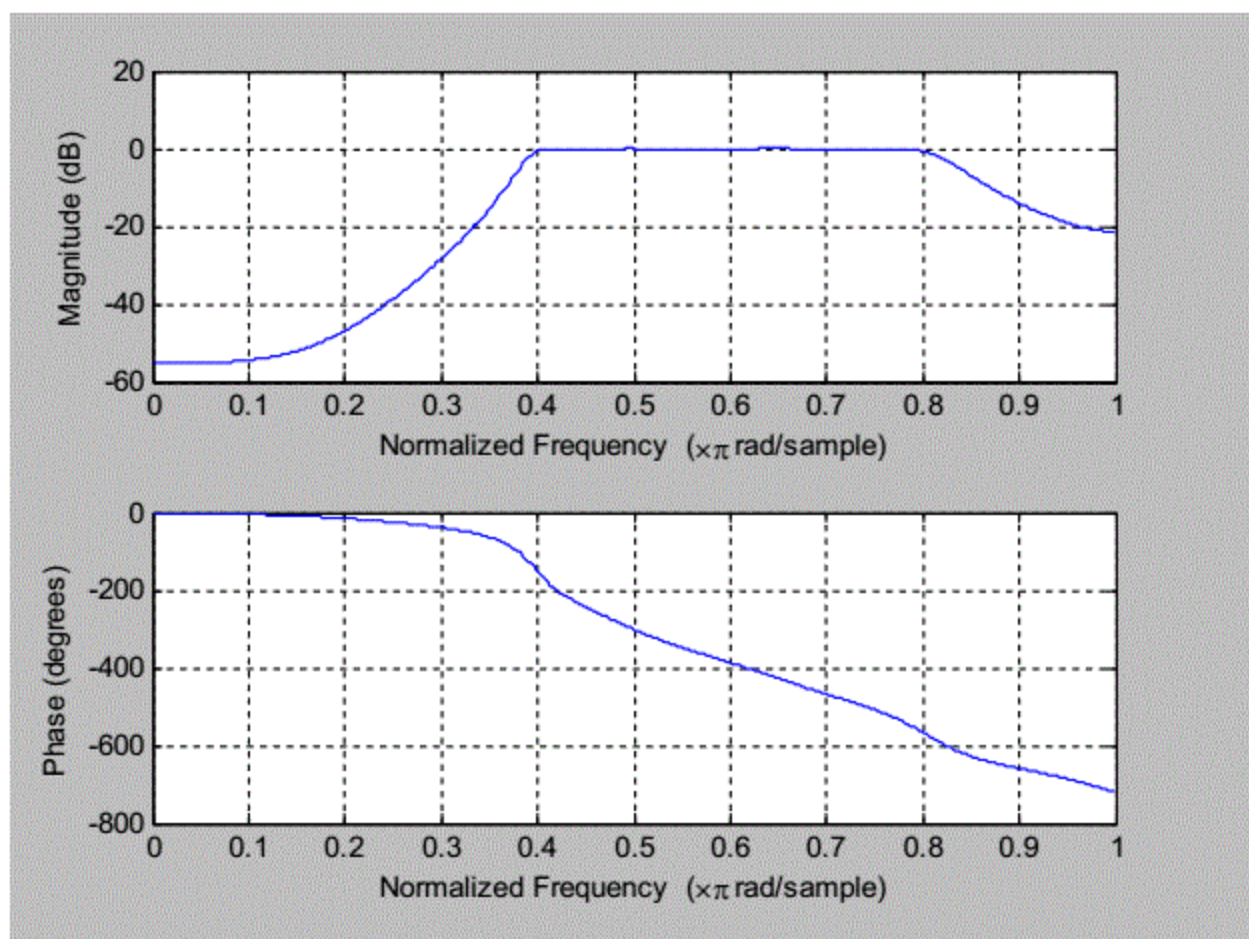


图二:模拟带通滤波器

(3) 冲击响应不变法实现数字带通滤波器的带通曲线如图三所示:



其幅频相频曲线如图四所示:



图四:带通滤波器的幅频和相频曲线

- (5) 为了对设计的数字带通滤波器的性能进行验证, 本设计加入了噪声信号, 如图五所示即为原来理想的信号波形

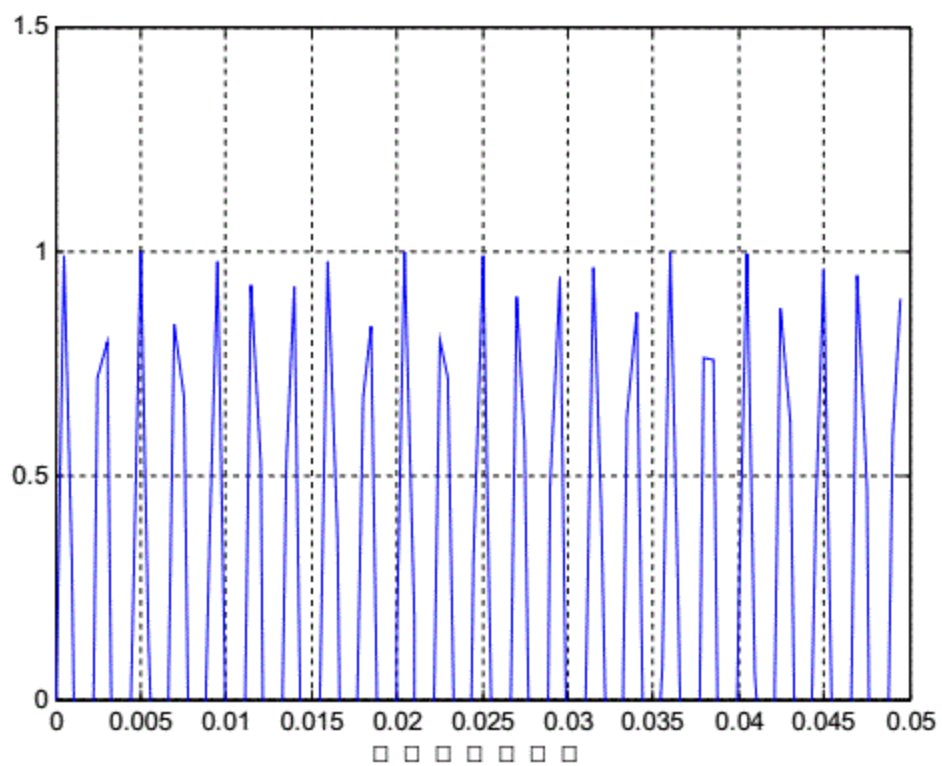
$$uzs=\sin(2*\pi*451*nn)$$

- (6) 如图六所示即为加入噪声信号后的波形曲线。

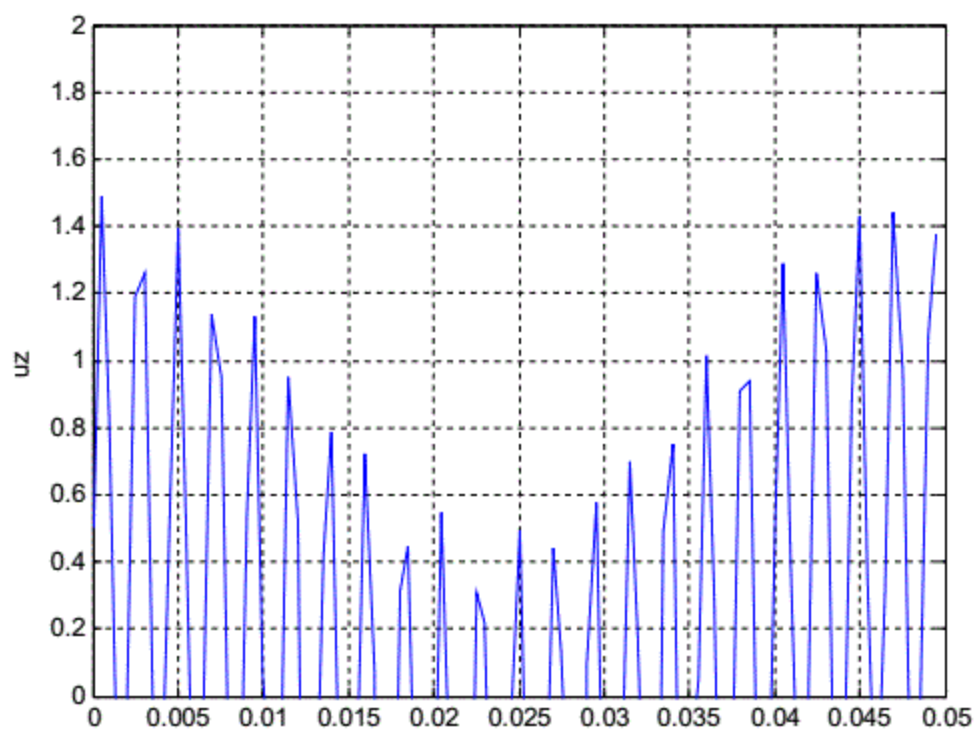
噪声信号的曲线方程为:

$$uz=0.5*\cos(2*\pi*21*nn)+uzs+2*\sin(2*\pi*1000*nn)$$

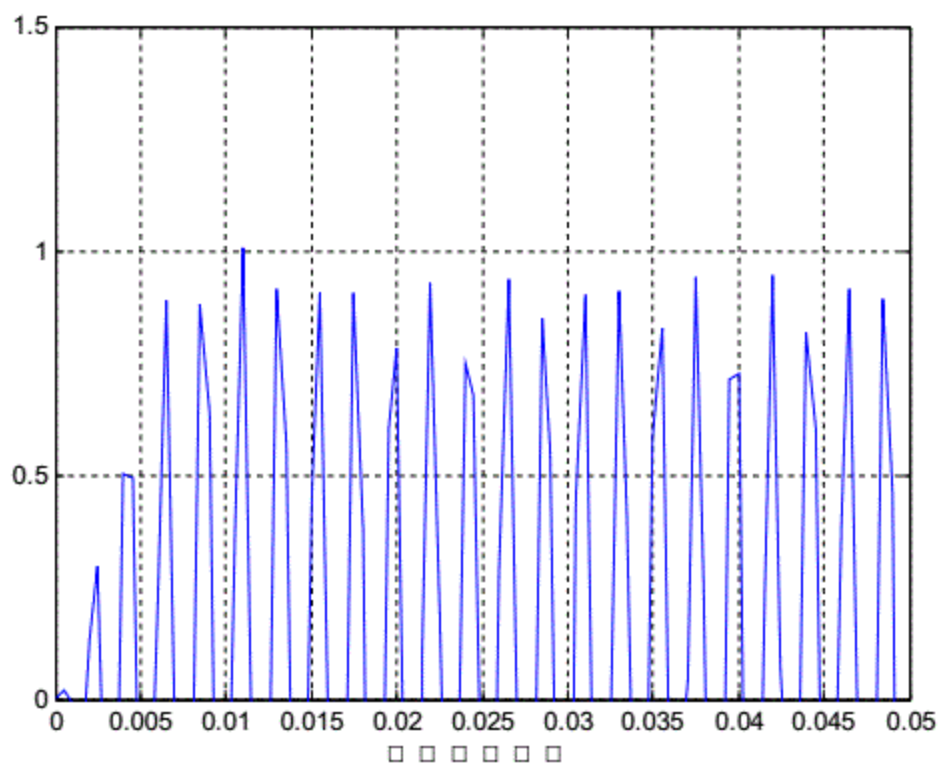
- (7) 如图七所示即为通过滤波器后输出的波形



图五:理想波形



图六:噪声信号



图七:滤波后输出波形

Matlab 程序代码:

```
w1 = 0.4*pi,w2 = 0.8*pi;    %digital signal freq
```

```
Rp=0.5;Rs = 20;
```

```
w3 = 0.3*pi,w4=1*pi;
```

```
fs = 2000;
```

```
Omega1 =fs*w1;
```

```
Omega2 = fs*w2;
```

```
Omega3 =fs*w3;
```

```
Omega4 = fs*w4;
```

```
f1 =Omega1/2/pi;
```

```
f2 = Omega2/2/pi;
```

```

f3 =Omega3/2/pi;
f4 = Omega4/2/pi;
Wp = [Omega1, Omega2];
Ws = [Omega3, Omega4];
[N, Wn]=cheblord(Wp, Ws, Rp, Rs, 's');
[z, p, k] = cheblap(N, Rp);
[bs, as]    = zp2tf(z, p, k);
n = 0:0.01:2;
[hs, ws]=freqs(bs, as, n);
figure;
plot(ws, abs(hs).^2); %low pass filter
grid;
bb=k*real(poly(z));
aa=real(poly(p));
Wo = sqrt(Omega1*Omega2);
Bw  = Omega2 - Omega1;
[bt, at] = lp2bp(bb, aa, Wo, Bw);
[h, w]    = freqs(bt, at);
figure;
subplot(2, 1, 1);plot(w/2/pi, (abs(h)).^2);
axis([100 1500 0 1]);
grid;

```

```

subplot(2,1,2);plot(w/2/pi,angle(h)*180/pi);

grid;

H = [tf(bt,at)];

[bz, az]=impinvar(bt,at,fs);

[hz,wz]=freqz(bz,az,fs);

figure;

plot(wz/pi,abs(hz));

grid;

figure;

freqz(bz,az);

Fn = 100;

dn= 1/fs;n=0:Fn-1;

nn= n*dn;

uzs=sin(2*pi*451*nn);

uz = 0.5*cos(2*pi*21*nn) + uzs+ 2*sin(2*pi*1000*nn);

figure;

plot(nn,uz);

grid;

axis([0 0.05 0 2]);

ylabel('uz');

Y = filter(bz,az,uz);

figure;

```

```
plot(nn, uzs);  
axis([0 0.05 0 1.5]);  
xlabel('预输出信号波形');  
grid;  
  
figure;  
plot(nn, Y);  
axis([0 0.05 0 1.5]);  
xlabel('实际输出波形');  
grid;
```

设计二:

设计一 Kalman 滤波器的设计

1. 概要:

Kalman 滤波理论是 Wiener 滤波理论的发展, 具有如下特点(1): 数学公式用状态空间概念描述; (2): 它的解释递推计算的, 与 Wiener 滤波器不同, Kalman 滤波器是一种自适应滤波器。

对于一般的时变系统, 系统状态方程和观测方程可表示为一下形式:

(1) 状态方程(state equation)

$$\mathbf{x}(n) = \mathbf{F}(n, n-1)\mathbf{x}(n-1) + \mathbf{\Gamma}(n, n-1)\mathbf{v}_1(n-1)$$

状态向量: $\mathbf{x}(n) \in \mathbb{C}^{N \times 1}$

状态转移矩阵: $\mathbf{F}(n, n-1) \in \mathbb{C}^{N \times N}$

状态噪声输入矩阵: $\mathbf{\Gamma}(n, n-1) \in \mathbb{C}^{N \times S}$

系统状态噪声: $\mathbf{v}_1(n-1) \in \mathbb{C}^{S \times 1}$

其中噪声 $\mathbf{v}_1(n)$ 为均值为零的高斯白噪声

(2) 观测方程(measurement equation)

$$\mathbf{z}(n) = \mathbf{C}(n)\mathbf{x}(n) + \mathbf{v}_2(n)$$

观测向量: $\mathbf{z}(n) \in \mathbb{C}^{M \times 1}$

观测矩阵: $\mathbf{C}(n) \in \mathbb{C}^{M \times N}$

观测噪声: $\mathbf{v}_2(n) \in \mathbb{C}^{M \times 1}$

其中噪声 $\mathbf{v}_2(n)$ 为均值为零的高斯白噪声

2. 应用:

假设一理想质点的运动轨迹是一理想的正弦波.

$$\text{如: } y=5*\sin (\pi*t*0.5)$$

但是当质点受到外部环境的干扰后就会产生一定的偏移, 利用函数 $\text{noise}=\text{randn}(1, 200)$ 产生随机的信号进行模拟, 所以实际的质点偏离中心位移的曲线为 $y_{\text{true}}= y + \text{noise}$.

可令质点的运动模型为 $y = \sin (x)$, 质点做匀速直线运动。令其经过的位移为 $s(n)$, 速度为 $v(n)$, 所以有:

$$s(n)=s(n-1)+Tv(n-1)$$

$$v(n)=v(n-1)$$

对于质点的干扰可以视为质点产生的加速度, 从而有:

$$s(n)=s(n-1)+Tv(n-1)+\frac{1}{2}T^2a(n-1)$$

$$v(n)=v(n-1)+Ta(n-1)$$

其中: a 表示质点受到的加速度

T 表示采样的间隔

从而状态方程可以表示为:

$$\begin{pmatrix} s(n) \\ v(n) \end{pmatrix} = \begin{pmatrix} 1 & T \\ 0 & 1 \end{pmatrix} \begin{pmatrix} s(n-1) \\ v(n-1) \end{pmatrix} + \begin{pmatrix} \frac{1}{2}T^2 \\ T \end{pmatrix} a(n-1)$$

令 $x(n)=(s(n), v(n))^T$, $\Gamma(n, n-1)=\begin{pmatrix} \frac{1}{2}T^2 \\ T \end{pmatrix}$, $v_1(n-1)=a(n-1)$
状态方程化为:

$$\mathbf{x}(n) = \mathbf{F}(n, n-1) \mathbf{x}(n-1) + \mathbf{\Gamma}(n, n-1) \mathbf{v}_1(n-1)$$

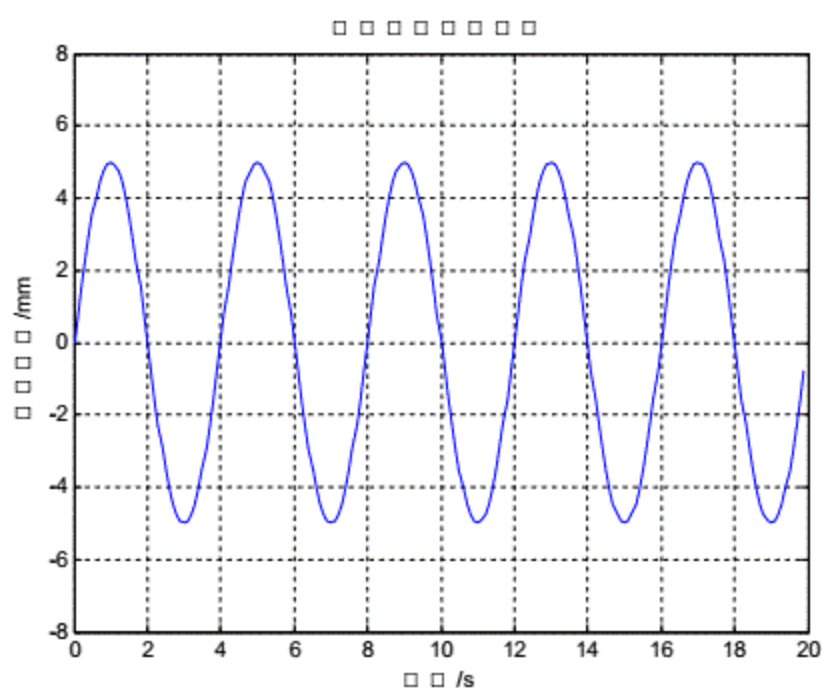
其中状态转移矩阵 $F(n, n-1) = \begin{pmatrix} 1 & T \\ 0 & 1 \end{pmatrix}$
 观测方程 $z(n) = C(n)x(n) + V_2(n)$

其中 $C(n) = (1 \ 0)$;

$V_2(n) = \text{noise}$

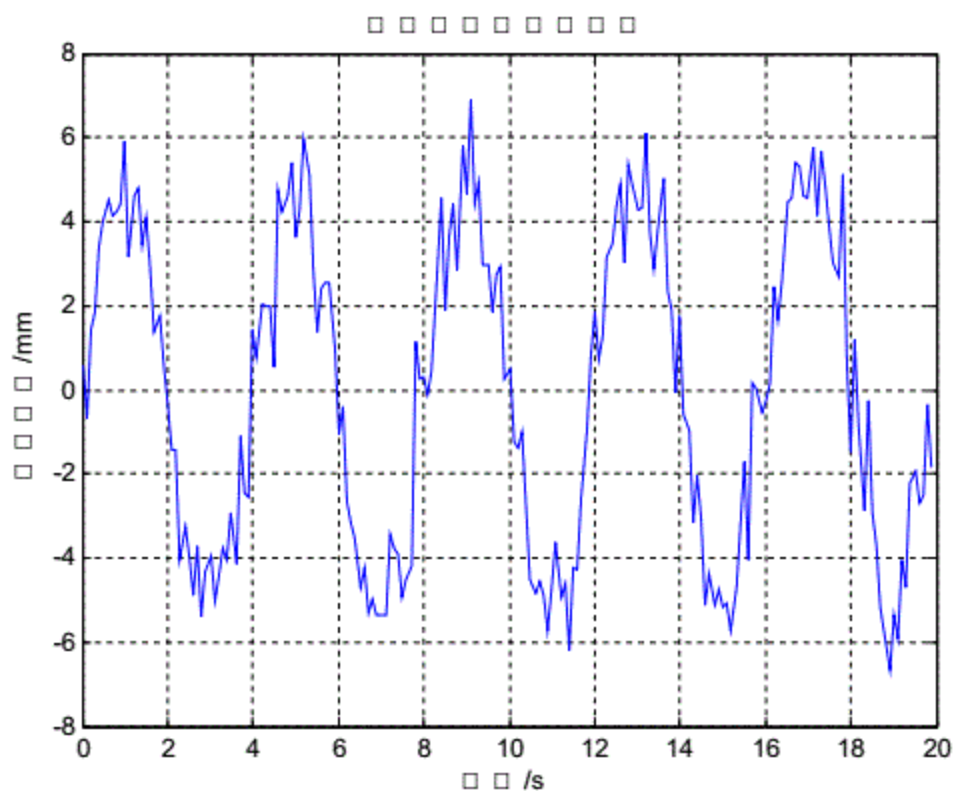
3. matlab 实现:

(1) 理想质点的运动轨迹如图一所示:



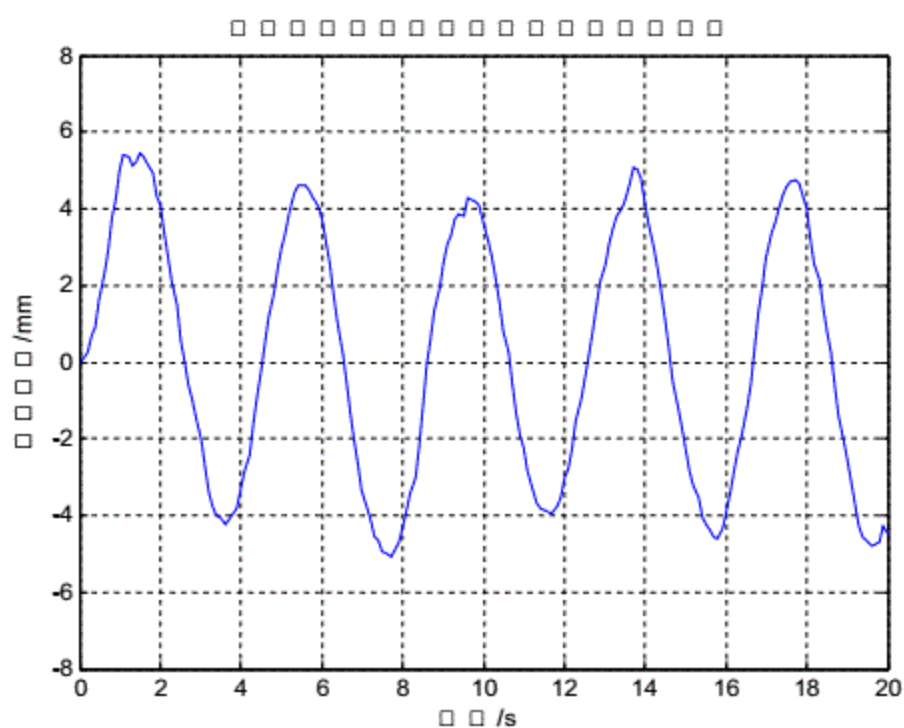
图一:理想质点运动轨迹

(2) 实际的质点的运动轨迹, 在 matlab 中用加上均值为零的高斯白噪声进行模拟, 波形如图二所示:



图二：实际的质点运动轨迹

(3) 经过 Kalman 滤波器后的输出的波形如图三所示：



图三：输出质点的位移

4. matlab程序代码

```
%-----产生正弦波信号-----%

T=0.1;

t=0:T:20-0.1;

y=5*sin(pi*t*0.5);

figure;

plot(t,y);

grid;

axis([0,20,-8,8]);

xlabel('时间/s');

ylabel('质点位移/mm');

title('理想质点运动轨迹');

%-----添加噪声-----%

%noise=randn(1,200);

noise = randn(1,200);

noise=noise-mean(noise);

noise=sqrt(var(noise))*noise/sqrt(var(noise));

Z=y+noise;

figure;

plot(t,Z);

grid;
```

```

axis([0, 20, -8, 8]);

xlabel(' 时间/s ');

ylabel(' 质点位移/mm ');

title(' 实际的质点运动轨迹 ');

%-----建立系统模型-----%

X=zeros(2, 200);

X(:, 1)=[0, 1];

F=[1, T; 0, 1];

V=[1/2*(T)^2 T]'; %V=[T T];    how to get the value of matrix V

%V=[1 1]';

C=[1, 0];

P=[0, 0; 0, 1];

%Q1=(0.25)^2;

%Q2=(0.25)^2;

Q1=cov(noise);

Q2=cov(noise);

%-----卡尔曼算法-----%

for n=1:200

    Kg=P*C'/(C*P*C'+Q2);

    X(:, n)=X(:, n)+Kg*(Z(:, n)-C*X(:, n));

    X(:, n+1)=F*X(:, n);

    P=(eye(2, 2)-Kg*C)*P;

```

```

        P=F*P*F' +V*Q1*V' ;

end

%-----%

figure

t=0:0.1:20;

Y=X(1, :);

plot(t, Y);

grid;

axis([0, 20, -8, 8]);

xlabel(' 时间/s ');

ylabel(' 质点位移/mm ');

title(' 卡尔曼滤波后输出质点最优化运动位移');

```