《信号与系统》课程设计----

基于小波变换的信号分解与重构仿真

Decomposition and Reconstruction of Signal based on Wavelet Transform

11 级 1 班刘北北 2011019060027

【主要内容】

1、一维连续小波变换的实现(cwt)

2、一维离散小波变换与重构(dwt)

【基本原理】

1、 傅里叶变换

原始信号中有一些信息是很难获取的，为了获得更多的信息，我们就需要对原始信号进行数学变换。

1822 年法国数学家傅里叶(J. Fourier 1768-1830)发表的研究热传导理论的“热的力学分析”，提出并证明了将周期函数展开为正弦级数的原理，奠定了傅里叶级数理论的基础[1]。

傅里叶级数理论研究的是把函数在三角函数系下的展开，使得对信号和系统的研究归结为对简单的三角函数的研究。傅立叶变换是用一些基础函数来分析和重构一个函数

在傅里叶变换后的频域中不包含时间信息，逆变换后的时域中不包含频率信息。频率分量则在不同的时间段内出现，虽然相应的时域信号之间差别很大，但两幅频谱图几乎是一致的。而且傅里叶变换不能刻画时域信号的局部特性，对非平稳信号的处理效果不好。

2、窗口傅里叶变换（短时傅里叶变换）（stft）

确定信号局部频率特性的比较简单的方法是在时刻r附近对信号加窗，然后计算傅里叶变换。

X(r,F)=STFT{x(t)}=FT{x(t)w(t- r)}

其中，w(t-r)是一个以时刻r为中心的窗函数。

3、小波变换

为了解决多分辨率的问题，小波变换作为替换短时傅立叶变换的一种方法被提出来。小波分析与短时傅立叶分析采用相同的处理方法，也是用窗口的形式将一个函数与信号相乘（小波变换中这个用来相乘的函数就是小波函数），变换结果被分成在时域内不同的片段。为计算每个单一的频谱分量，需要将窗口宽度改变，这可能是小波变换最重要的特征。

连续小波变换的定义如下：

$$CWT_x^{\psi}(\tau, s) = \Psi_x^{\psi}(\tau, s) = \frac{1}{\sqrt{|s|}} \int x(t)\psi^*\left(\frac{t-\tau}{s}\right) dt$$

这个连续小波变换的定义说明了小波分析就是用来测量基本函数（即小波函数）与信号本身的相似性，这里的相似是指相似的频率分量。计算出来的连续小波变换系数说明了在当前尺度下原始信号与小波信号的近似程度。

小波变换其实是一个含有两个自变量的函数，tau 和 s，分别作为平移和缩放参数。小尺度 a →压缩的小波→快速变换的细节→高频部分；大尺度 a →拉伸的小波→缓慢变换的粗部→低频部分。psi(t)是变换函数，叫做母小波。母小波就是产生其他窗函数的原型。

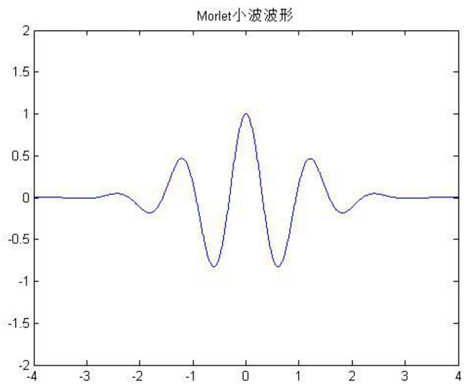
“小波”的意思就是小的波形。窗函数的条件：1.有限宽度（即紧支撑的）2.函数是振荡的。

常用的小波有 Haar 小波、 Daubechies 小波、Symlet 小波、双正交小波（biorNr.Nd）、Coiflet 小波、 Morlet 小波、Mexico 草帽小波和 Meyer 小波等等。

简单举两例：

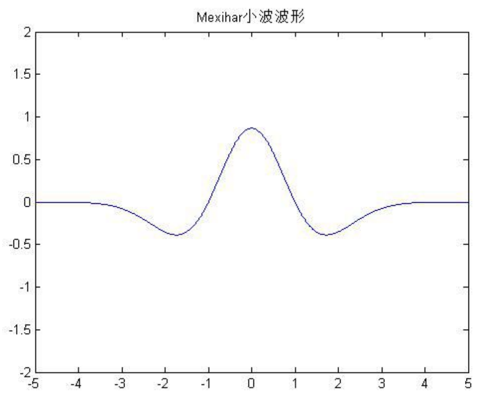
Morlet 小波定义为：

$$w(t) = e^{iat}.e^{-\frac{t^2}{2\sigma}}$$



墨西哥帽小波被定义为高斯函数的二阶微分：

$$w(t) = \frac{1}{\sqrt{2\pi}\sigma} e^{\frac{-t^2}{2\sigma^2}}$$



4、小波的重构与恢复

由小波系数计算原始信号值的小波重构过程可用如下公式计算：

$$x(t) = \frac{1}{c_{\psi}^{2}} \int_s \int_\tau \Psi_x^{\psi}(\tau, s) \frac{1}{s^2} \psi\left(\frac{t-\tau}{s}\right) d\tau\, ds$$

其中 C_psi 为与所用小波有关的常数。这个与小波重构过程有关的常数称为容许性常数。式 3.18 所示的小波重构条件称为容许性条件。

$$c_{\psi} = \left\{2\pi \int_{-\infty}^{\infty} \frac{|\hat{\psi}(\xi)|^2}{|\xi|} d\xi\right\}^{1/2} < \infty$$

小波 psi(tau,s)如果满足式 3.18 所示的容许性条件，则能够利用式 3.17 完全恢复原始信号。

5、傅里叶变换与小波变换的比较

1）两者都是正交变换。正交变换（即把信号分解为相互正交的分量的加权和）可以减少向量之间的相关性，从而有利于信号的处理。

2）两种变换都是用来测量基本函数（即正交函数系）与信号本身的相似性。傅里叶变换的基本函数是正弦余弦函数、复指数函数等，小波变换的基本函数是小波函数。

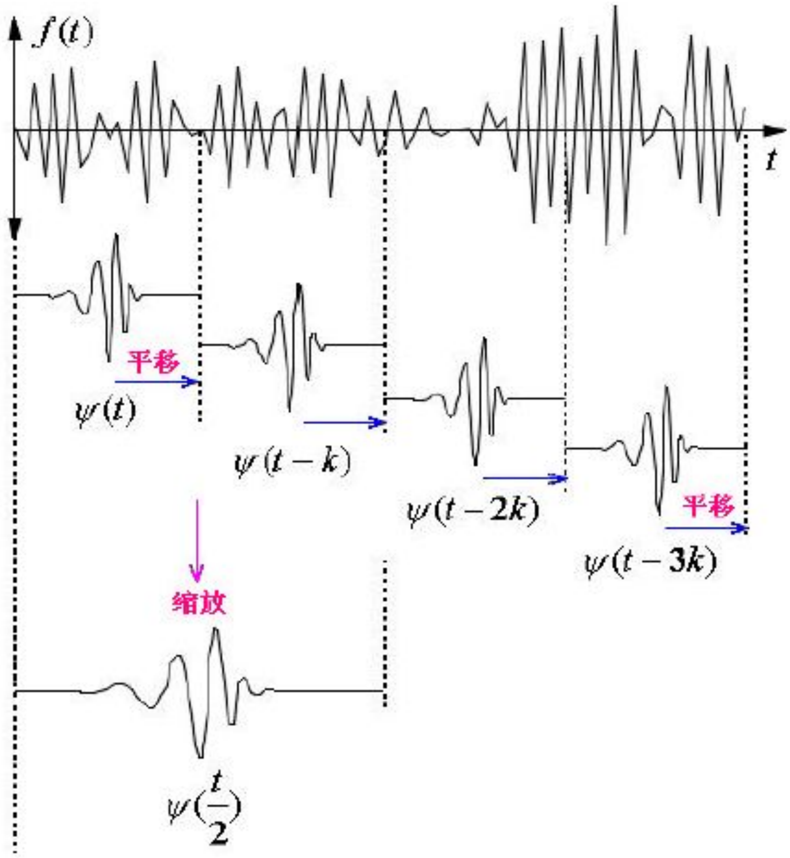
采用其他正交函数系还可以得到其他正交变换，如沃尔什变换、哈尔变换、K-L 变换等。

【具体内容】

一、一维连续小波变换（CWT）的实现

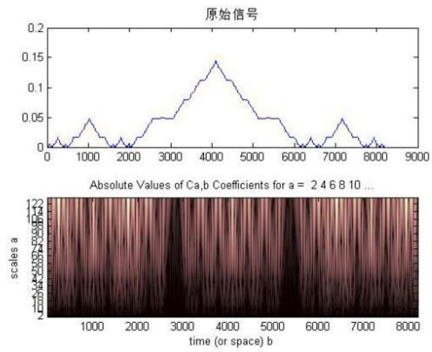
CWT 的变换过程

1. 把小波ψ(t)和原始信号 f(t)的开始部分进行比较
2. 计算系数 c 。该系数表示该部分信号与小波的近似程度。系数 c 的值越高表示信号与小波越相似，因此系数 c 可以反映这种波形的相关程度
3. 把小波向右移，距离为 k，得到的小波函数为ψ(t-k)，然后重复步骤 1 和 2。再把小波向右移，得到小波ψ(t-2k)，重复步骤 1 和 2。按上述步骤一直进行下去，直到信号 f(t)结束
4. 扩展小波ψ(t)，例如扩展一倍，得到的小波函数为ψ(t/2)
5. 重复步骤 1~4



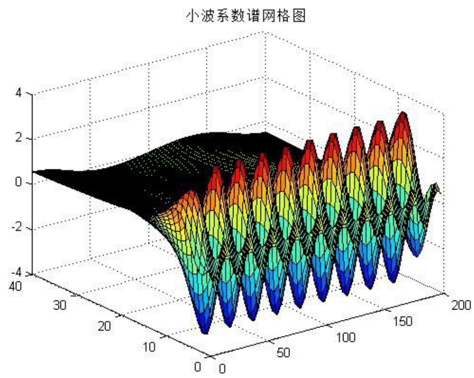
Matlab 中的实现：（注释是对原理的解释）
（注：下面的这些代码是根据小波变换的定义式，分部分计算的，比较原始，体现了前面说的平移和伸缩的原理， matlab 中有现成的小波变换语句）
程序代码如下：

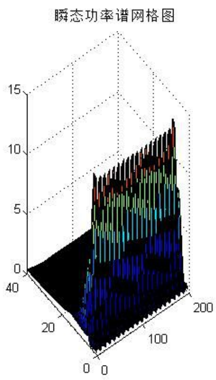
```
load vonkoch;
x=vonkoch;
subplot(211);
plot(x);
title('原始信号');
subplot(212);

f=cwt(x,[2:2:128],'coif3','plot')    ;
```

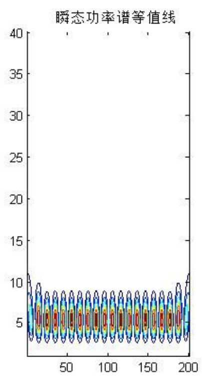



```
function Y=mexh(x) % 自定义的函数，另存为mexh.m
if abs(x)<=8
Y=exp(-x*x/2)*(1-x^2);
else
   Y=0;
end

for i=1:200
```

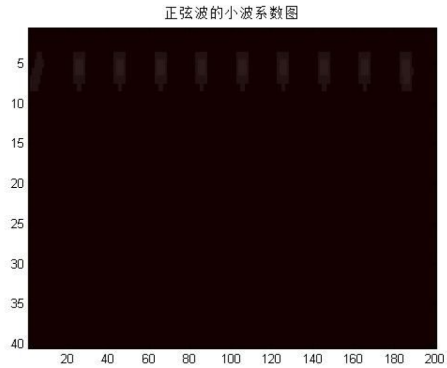
```
dat(i)=sin(2*pi*i*0.05);      %定义一个以i为自变量的正弦函数，作为待处理的信号
end
len=length(dat);
lna=40;                       % （尺度a）的长度
a=zeros(1,lna);               %矩阵a记录每个尺度对应的小波系数
wfab=zeros(lna,len);          %矩阵wfab为小波系数矩阵
mexhab=zeros(1,len);          % 离散化小波系数矩阵

for s=1:lna         %s 表示尺度
    for k=1:len      %从左到右扫描原信号
        mexhab(k)=mexh(k/s);  %k除以s(因为尺度要随t的正一倍变）后，代入mexh的
                              %小波定义式中，求出的psi值计入矩阵mexhab中
    end
    for t=1:len    % t 表示位移。以某一尺度从左到右扫完一遍后将这70个psi值分别代
入公式，并积分
      wfab(s,t)=(sum(mexhab.*dat))/sqrt(s);   %将积分用求和代替
      mexhab=[mexh(-1*t/s),mexhab(1:len-1)];  %mexhab修改第一项并右移
    end
end
figure(1);
plot(dat);
title('原始数据图');
figure(2);  %小波系数谱
image(wfab);
title('小波系数图');
surf(wfab);
title('小波系数谱网格图');

pwfab=wfab.*wfab;  %瞬态功率谱
figure(3);
subplot(1,2,1);
surf(pwfab);
title('瞬态功率谱网格图');

subplot(1,2,2);
contour(pwfab);
title('瞬态功率谱等值线');
```







;



二、一维离散小波变换(DWT)与重构

虽然离散化的连续小波变换（即小波级数）使得连续小波变换的运算可以用计算机来实现，但这还不是真正的离散变换。事实上，小波级数仅仅是 CWT 的采样形式。即便是考虑到信号的重构，小波级数所包含的信息也是高度冗余的。这些冗余的信息同样会占用巨大的计算时间和资源。而离散小波变换（DWT）则不仅提供了信号分析和重构所需的足够信息，其运算量也大为减少。

1、Mallat 算法

原理：

小波变换就是将 “ 原始信号 s ” 变换 成 “ 小波 系数 w ”，

w=[wa , wd] （ 近似系数 wa 与细节系数 wd)

则原始信号 s 可分解成小波近似 a 与小波细节 d 之和。

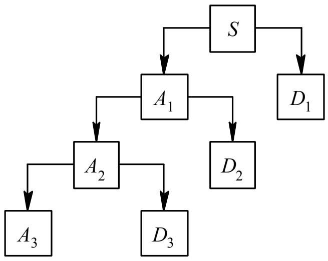
s = a+d

小波系数 w = [wa , wd] 的分量，乘以基函数，形成小波分解：

小波近似系数 wa ×基函数 A=近似分解 a ---平均

小波细节系数 wd ×基函数 D=细节分解 d---变化

执行离散小波变换的有效方法是使用滤波器，该方法是 Mallat 于 1988 年提出的，称为 Mallat 算法。这种方法实际上是一种信号分解的方法， 在数字信号处理中常称为双通道子带编码。

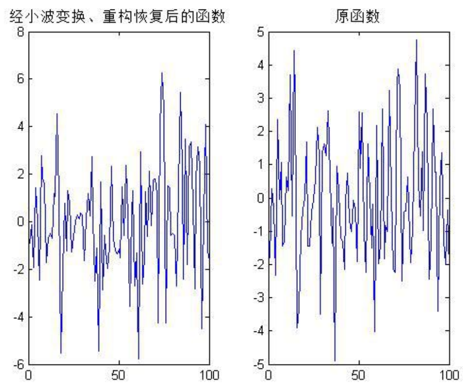


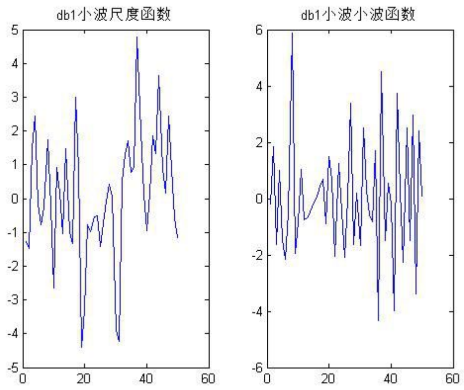
Matlab 中的实现

（注：下面的代码在高通、低通滤波的时候都用的是 conv 卷积来实现的，更原始一些，能体现这种分解方法的核心原理。matlab 中有直接计算计算小波变换及其反变换的语句：[cA,cD]=dwt(X,’小波名’)；X=idwt(cA,cD,’小波名’)）

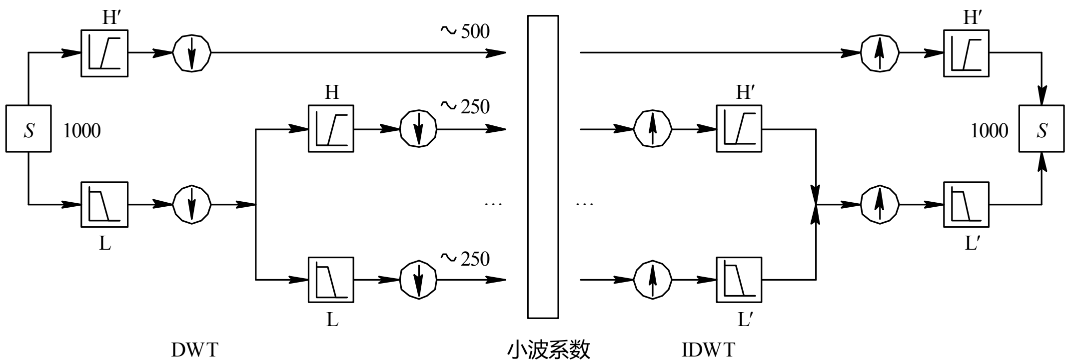
```
t=(0.01:0.01:1)*100;
x=0.5*sin(2*pi*15*t)+2*sin(2*pi*40*t);
y=x+2.0*randn(size(t));%加随机噪声
yo=y(1:2:99);%抽取信号 y 的所有奇数样点（下采样）yo
ye=y(2:2:100);%抽取信号 y 的所有偶数样点（下采样）ye
LoD=[0.7071,0.7071];%db1小波的分解滤波器,低通
HiD=[-0.7071,0.7071];%db1小波的分解滤波器,高通
Ly=conv(yo,LoD);%与db1低通滤波器做卷积（滤波）运算,得到y 信号的低频分量（近似成分）Ly
Hy=conv(ye,HiD);%与db1高通滤波器做卷积（滤波）运算,得到y 信号的高频分量（细节成分）Hy

[LoD, HiD, LoR, HiR] = wfilters( 'db1' );%要进行小波反变换（即重构）% LoD-分解低通滤波器；HiD-分解高通滤波器；
% LoR-重构低通滤波器；HiR-重构高通滤波器。

lly=conv(Ly(1:50),LoR);%y 信号的低频分量（近似成分）Ly与harr 低通滤波器做卷积（滤波）运算
hhy=conv(Hy(1:50),HiR);%y 信号的高频分量（细节成分）Hy与harr 高通滤波器做卷积（滤波）运算
yn=zeros(1,100);%先建一个数组，置零
yn(1:2:99)=lly(1:50);%对信号的所有奇数样点lly进行插值（上采样）得yn
yn(2:2:100)=hhy(1:50);%对信号的所有偶数样点hhy进行插值（上采样）得yn
figure(2)
subplot(121)
plot(t,yn);title('经小波变换、重构恢复后的函数')
subplot(122)
plot(t,y(1:100));title('原函数')
figure(1)
subplot(121);
plot(t(1:50),Ly(1:50));title('db1小波尺度函数')
```

```
subplot(122);
plot(t(1:50),Hy(1:50));title('db1小波小波函数')
```





多层小波分解和重构示意图



2、多层小波分解和低频重构

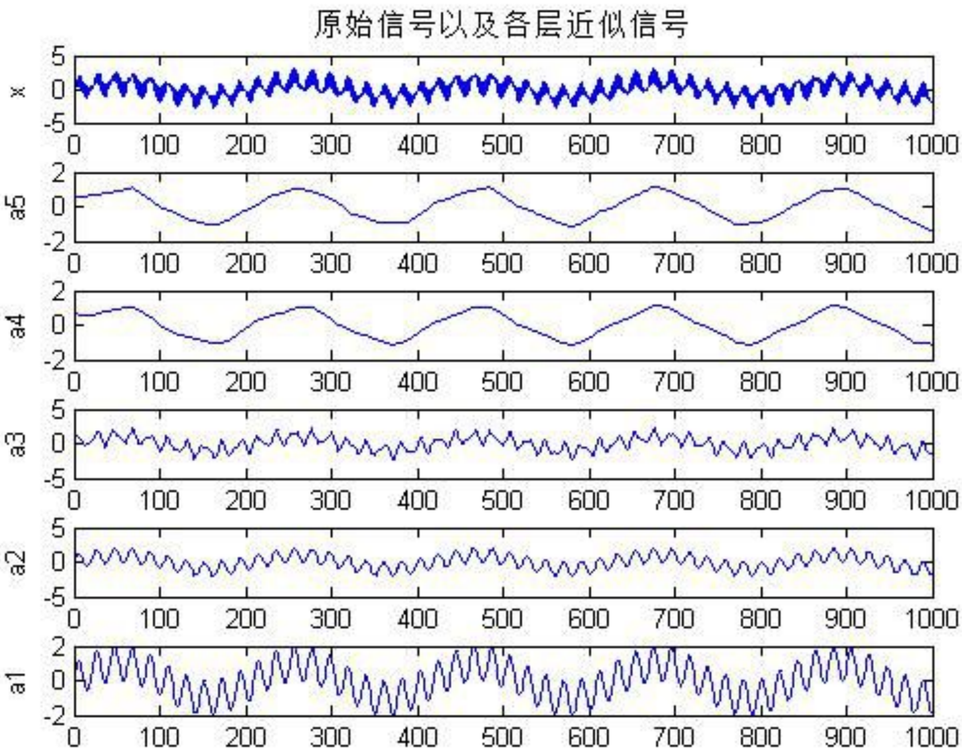
由于噪声的污染，从原始信号 x 中无法观察信号的发展趋势。通过进行多尺度的小波分解，只将小波分解的低频系数重构，则可以明显地看到原始信号的发展趋势。

这是因为信号的发展趋势往往体现在信号的低频成分上，在小波变换中对应着最大尺度小波变换的低频系数。

用 sumsin 这个测试信号举例说明多层分解，低频系数重构：

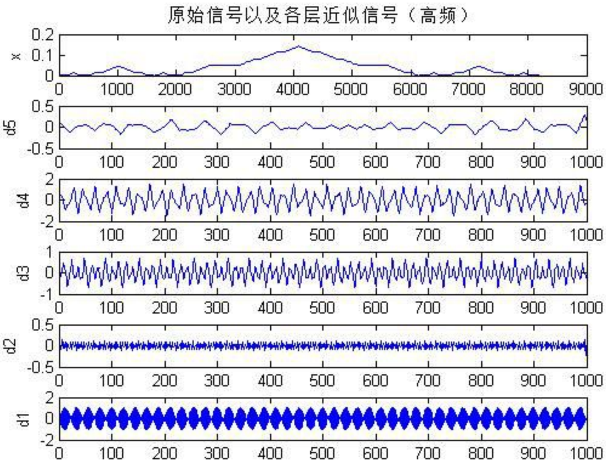
```
load sumsin;
x=sumsin;
figure;
subplot(611);
plot(x);
ylabel('x');
title('原始信号以及各层近似信号');

[c,l]=wavedec(x,5,'db3');      %使用 db3 小波进行 5 层分解
for i=1:5        %对分解的第 5 层到第 1 层的低频系数分别进行重构
    a=wrcoef('a',c,l,'db3',6-i);
      subplot(6,1,i+1);
```

```
        plot(a);
        ylabel(['a',num2str(6-i)]);
end
```



高频重构就不能很好体现原来的形态：

```
figure;
subplot(611)
plot(x);
ylabel('x')
title('原始信号以及各层近似信号（高频）');
for i=1:5
%对分解的第 5 层到第 1 层的高频系数进行重构
       d=wrcoef('d',c,l,'db3',6-i);
       subplot(6,1,i+1);
       plot(d);
       ylabel(['d',num2str(6-i)]);
end
```



三、用快速傅里叶变换（fft）实现连续小波变换（cwt）

$$CWT_x^{\psi}(\tau,s)=\Psi_x^{\psi}(\tau,s)=\frac{1}{\sqrt{|s|}}\int x(t)\psi^*\left(\frac{t-\tau}{s}\right)dt$$

按照圆周卷积定理，原周卷积和线性卷积的关系 L>=M+N-1

例如：x=(x1,x2,x3,x4),y=(y1,y2,y3),则两者卷积 x*y 结果应有 4+3-1=6 项

用 6 阶矩阵表示这个过程：

$$\begin{bmatrix} x1 & 0 & 0 & 0 & 0 & 0 \\ x2 & x1 & 0 & 0 & 0 & 0 \\ x3 & x2 & x1 & 0 & 0 & 0 \\ x4 & x3 & x2 & x1 & 0 & 0 \\ 0 & x4 & x3 & x2 & x1 & 0 \\ 0 & 0 & x4 & x3 & x2 & x1 \end{bmatrix} \cdot \begin{bmatrix} y1 \\ y2 \\ y3 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} z1 \\ z2 \\ z3 \\ z4 \\ z5 \\ z6 \end{bmatrix}$$

Matlab 中的实现：

```
t1=cputime;

for i=1:200
    dat(i)=sin(2*pi*i*0.05);
end
```

```
len=length(dat);
lna=70;             % a（尺度）的长度
a=zeros(1,lna);     % a 表示尺度
b=zeros(1,len);      % b 表示位移
wfab=zeros(lna,len);    %小波系数矩阵
mexhab=zeros(1,2*len-1);    %相当于例子中的z矩阵。由原周卷积和线性卷积的关系
                                          L>=M+N-1,此处M=len,N=len
data=[zeros(1,len-1),dat];  %相当于例子中的y矩阵
Ydata=fft(data,4*len);
for s=1:lna            %尺度s变换的循环
    for k=1:2*len-1     %位移k变换的循环
       mexhab(k)=mexh((k-len)/s);    %计算psi
    end

   temp=ifft( Ydata.*fft( mexhab,4*len ) ,4*len);
   wfab(s,:)=real(temp(2*len-1:3*len-2))/sqrt(s); %矩阵第s行：即当前尺度
                                             下每一个位移k对应的小波系数
end
figure(1);
plot(dat);
title('原始数据图');
figure(2);  %小波系数谱
image(wfab);
colormap(pink(128));
title('小波系数谱 ');
cputime-t1
```