TFW-HW4

R04943133 林岳徵

Code:

main.m

clc

clear all

close all

t = [0: 0.01: 10];

x = 0.2\*t + cos(2\*pi\*t) + 0.4\*cos(10\*pi\*t);

thr = 0.2;

y = hht(x, t, thr);

hht:

function y=hht(x\_org, t, thr)%by R04943133

%Step1

%initial

dt=t(2)-t(1);

x=x\_org;

%threshode n:1~30, k:1~30,

n=1; n\_upper=30;

k=1; k\_upper=30;

IMF\_matrix = zeros(n\_upper, length(t));

%Step2

for step8=1:n\_upper

for step7=1:k\_upper

clear lmax; %local maximum for x(i)

clear lmax\_sp; %sampling points of local maximum for x(i)

index=1;

for i=1:length(x)

if (i==1 || i==length(x))

lmax(index)=0;

lmax\_sp(index)=(i-1)\*dt;

index=index+1;

elseif( (x(i)>x(i-1)) && (x(i)>x(i+1)) )

lmax(index)=x(i);

lmax\_sp(index)=(i-1)\*dt;

index=index+1;

end

end

%boundary lmax(1) & lmax(max)

if(length(lmax)>=10)

lmax32=lmax(3)-lmax(2);

lmax\_sp32=lmax\_sp(3)-lmax\_sp(2);

lmax(1)=lmax(2)-lmax32/lmax\_sp32\*lmax\_sp(2);

lmax\_len=length(lmax);

lmax12=lmax(lmax\_len-1)-lmax(lmax\_len-2);

lmax\_sp12=lmax\_sp(lmax\_len-1)-lmax\_sp(lmax\_len-2);

lmax(lmax\_len)=lmax(lmax\_len-1)+lmax12/lmax\_sp12\*(10-lmax\_sp(lmax\_len-1));

end

%Step3

%local max curve for x(i) => lmaxc

clear lmaxc;

lmaxc=spline(lmax\_sp, lmax, t);

%Step4

clear lmin; %local minimum for x(i)

clear lmin\_sp; %sampling points of local minimum for x(i)

index=1;

for i=1:length(x)

if (i==1 || i==length(x))

lmin(index)=0;

lmin\_sp(index)=(i-1)\*dt;

index=index+1;

elseif( (x(i)<x(i+1)) && (x(i)<x(i-1)) )

lmin(index)=x(i);

lmin\_sp(index)=(i-1)\*dt;

index=index+1;

end

end

%boundary lmin(1) & lmin(max)

if(length(lmin)>=10)

lmin32=lmin(3)-lmin(2);

lmin\_sp32=lmin\_sp(3)-lmin\_sp(2);

lmin(1)=lmin(2)-lmin32/lmin\_sp32\*lmin\_sp(2);

lmin\_len=length(lmin);

lmin12=lmin(lmin\_len-1)-lmin(lmin\_len-2);

lmin\_sp12=lmin\_sp(lmin\_len-1)-lmin\_sp(lmin\_len-2);

lmin(lmin\_len)=lmin(lmin\_len-1)+lmin12/lmin\_sp12\*(10-lmin\_sp(lmin\_len-1));

end

%Step5

%local minimum curve for x(i) => lminc

clear lminc;

lminc=spline(lmin\_sp, lmin, t);

%Step6-1

for i=1:length(t)

z(i)=(lmaxc(i)+lminc(i))/2;

end

%x\_z=linspace(0,10,length(z));

%Step6-2

for i=1:length(t)

h(i)=x(i)-z(i);

end

%Step7

%Checking h(i)

fail=0;

clear lmax2; %local maximum for h(i)

clear lmax2\_sp; %sampling points of local maximum for h(i)

index=1;

for i=1:length(x)

if (i==1 || i==length(x))

lmax2(index)=0;

lmax2\_sp(index)=i;

index=index+1;

elseif( (h(i)>h(i+1)) && (h(i)>h(i-1)) )

lmax2(index)=h(i);

lmax2\_sp(index)=i;

%check whether local maximums > 0

if(fail==0 && lmax2(index)<0)

fail=1;

end

index=index+1;

end

end

%boundary lmax2(1) & lmax2(max)

if(length(lmax2)>=10)

lmax2\_32=lmax2(3)-lmax2(2);

lmax2\_sp32=lmax2\_sp(3)-lmax2\_sp(2);

lmax2(1)=lmax2(2)-lmax2\_32/lmax2\_sp32\*lmax2\_sp(2);

lmax2\_len=length(lmax2);

lmax2\_12=lmax2(lmax2\_len-1)-lmax2(lmax2\_len-2);

lmax2\_sp12=lmax2\_sp(lmax2\_len-1)-lmax2\_sp(lmax2\_len-2);

lmax2(lmax2\_len)=lmax2(lmax2\_len-1)+lmax2\_12/lmax2\_sp12\*(10-lmax2\_sp(lmax2\_len-1));

%check whether local maximums > 0

if(fail==0 && (lmax2(1)<0 || lmax2(lmax2\_len)<0))

fail=1;

end

end

%local maximums curve for h(i) => lmaxc2

clear lmaxc2;

lmaxc2=spline(lmax2\_sp, lmax2, t);

clear lmin2; %local minimum for h(i)

clear lmin2\_sp; %sampling points of local minimum for h(i)

index=1;

for i=1:length(x)

if (i==1 || i==length(x))

lmin2(index)=0;

lmin2\_sp(index)=i;

index=index+1;

elseif( (h(i)<h(i+1)) && (h(i)<h(i-1)) )

lmin2(index)=h(i);

lmin2\_sp(index)=i;

%check whether local minimums < 0

if(fail==0 && lmin2(index)>0)

fail=1;

end

index=index+1;

end

end

%boundary lmin2(1) & lmin2(max)

if(length(lmin)>=10)

lmin2\_32=lmin2(3)-lmin2(2);

lmin2\_sp32=lmin2\_sp(3)-lmin2\_sp(2);

lmin2(1)=lmin2(2)-lmin2\_32/lmin2\_sp32\*lmin2\_sp(2);

lmin2\_len=length(lmin2);

lmin2\_12=lmin2(lmin2\_len-1)-lmin2(lmin2\_len-2);

lmin2\_sp12=lmin2\_sp(lmin2\_len-1)-lmin2\_sp(lmin2\_len-2);

lmin2(lmin2\_len)=lmin2(lmin2\_len-1)+lmin2\_12/lmin2\_sp12\*(10-lmin2\_sp(lmin2\_len-1));

%check whether local minimums < 0

if(fail==0 && (lmin2(1)>0 || lmin2(lmin2\_len)>0))

fail=1;

end

end

%local minimums curve for h(i) => lminc2

clear lminc2;

lminc2=spline(lmax2\_sp, lmax2, t);

%check whether means < threshold

for i=1:length(t)

mean(i)=(lmaxc2(i)+lminc2(i))/2;

if( fail==0 && abs(mean(i))>thr )

fail=1;

break;

end

end

%check fail status

IMF\_matrix(n,:)=h;

if(fail==0)

k=1;

break;

else

x=h;

k=k+1;

end

%Step7 end

end

%Step8

c\_sum=zeros(1,length(t));

for j=1:length(t)

for i=1:n

c\_sum(j)=IMF\_matrix(i,j)+c\_sum(j);

end

end

x0=x\_org-c\_sum;

%local maximum

entry1=0;

for i=2:length(x0)-1

if( (x0(i)>x0(i+1)) && (x0(i)>x0(i-1)) )

entry1=entry1+1;

end

end

%local minimum

entry2=0;

for i=2:length(x0)-1

if( (x0(i)<x0(i+1)) && (x0(i)<x0(i-1)) )

entry2=entry2+1;

end

end

%the number of non-boundary extremes can be no more than 3

if( entry1+entry2 <= 3)

break;

else

n=n+1;

x=x0;

end

%Step8 end

end

%Show each IMF of x(t)

for i=1:n

figure;

plot(t,IMF\_matrix(i,:));

title('IMF','Fontsize',12);

xlabel('t (Sec)','Fontsize',12) ;

ylabel(['IMF' num2str(i) ],'Fontsize',12);

axis([-inf,inf,-2.3,2.3]);

end

y=x0;

  