```
% Starting-point-script for exercise 1 in the Simulation Methods in
% Ultrasound Imaging course (MEDT8007).
%
%
% author:  Marco M. Voormolen
% draft:   15 March 2008

% update:
%
% uses:
% sub of:
```

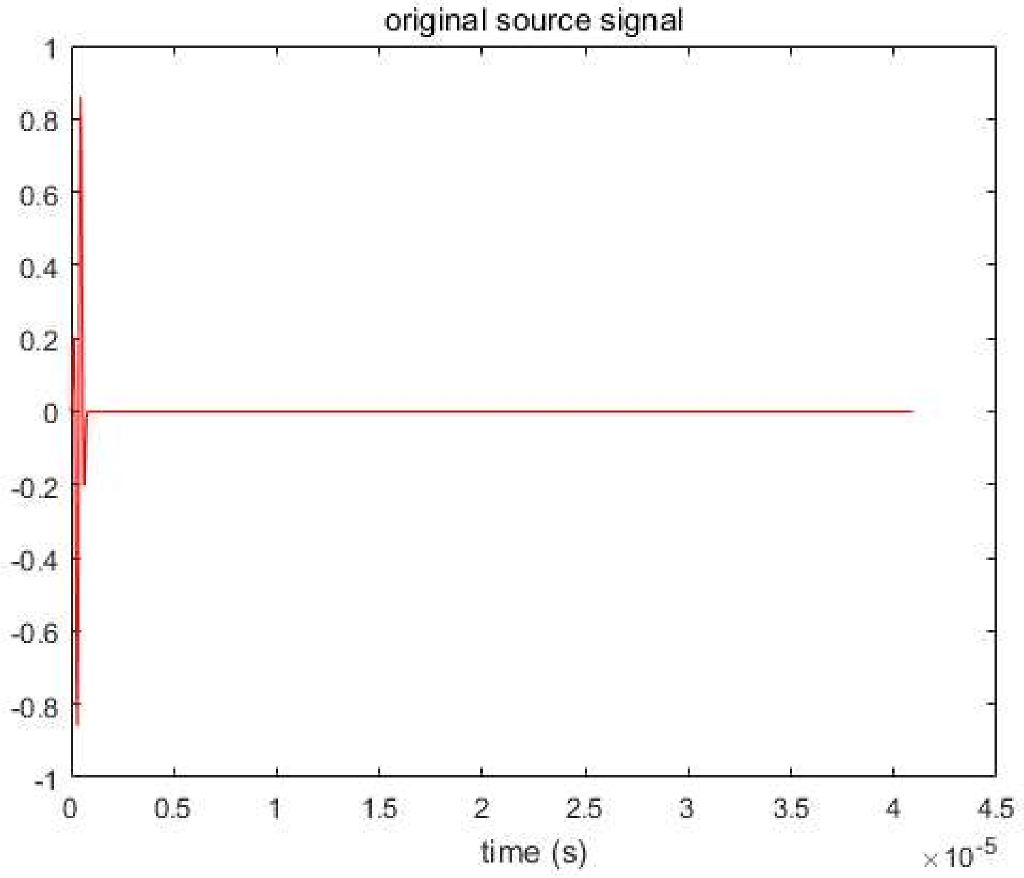
```
clear all
close all
fclose('all');


% Parameters (see figure 3.16 on page 156)
a=3e-3; % aperture radius in m
ds=0.4e-3; % spatial resolution in m
x_1=-a:ds:a; % x-coordinate of the (candidate) aperture point(s) in m
y_1=-a:ds:a; % y-coordinate of the (candidate) aperture point(s) in m
x_0=-30e-3:0.5e-3:30e-3; % x-coordinate of the observation point(s) in m
x_0=5e-3; % 5e-3
y_0=0e-3; % y-coordinate of the observation point(s) in m
z_0=50e-3; % z-coordinate of the observation point(s) in m
f_c=2.5e6; % center frequency of the transducer/aperture in Hz
f_Sample=10*f_c; % sample frequency in Hz
N_FFT=1024; % length of the FFT
t_s=0:1/f_Sample:0.8e-6; % non-zero duration of the excitation function in s
s=[sin(0.25*2*pi*f_c*t_s).^2.*sin(2*pi*f_c*t_s) zeros(1, N_FFT - length(t_s))]; % excitati
on function
c_0=1500; % speed of sound in m/s
df=f_Sample/N_FFT; % frequency resolution in Hz
f_c = 0:df:floor(N_FFT/2)*df; % 0.1e6:0.1e5:6e6;
f=f_c; % frequency (range) Hz


% Huygens' principle
R=zeros(length(x_1), length(y_1));
H=zeros(length(x_0), length(f)); % complex amplitude function
h_WB=waitbar(0);
tic
for m=1:length(x_1)
  for n=1:length(y_1)
    for q=1:length(x_0)
      if sqrt(x_1(m)^2 + y_1(n)^2)<=a % only accept points that lie within the radius of t
he aperture
        %<frequency independent variables>
                r =[x_0(q) - x_1(m) y_0-y_1(n) z_0];
                R(m,n) = sqrt(sum(r.*r));
        for i=1:length(f)
                  k(i) = 2*pi*f(i)/c_0;
          H(q, i)=H(q, i) + exp(-1i*k(i)*R(m,n))/R(m,n);
        end
      end
    end
```

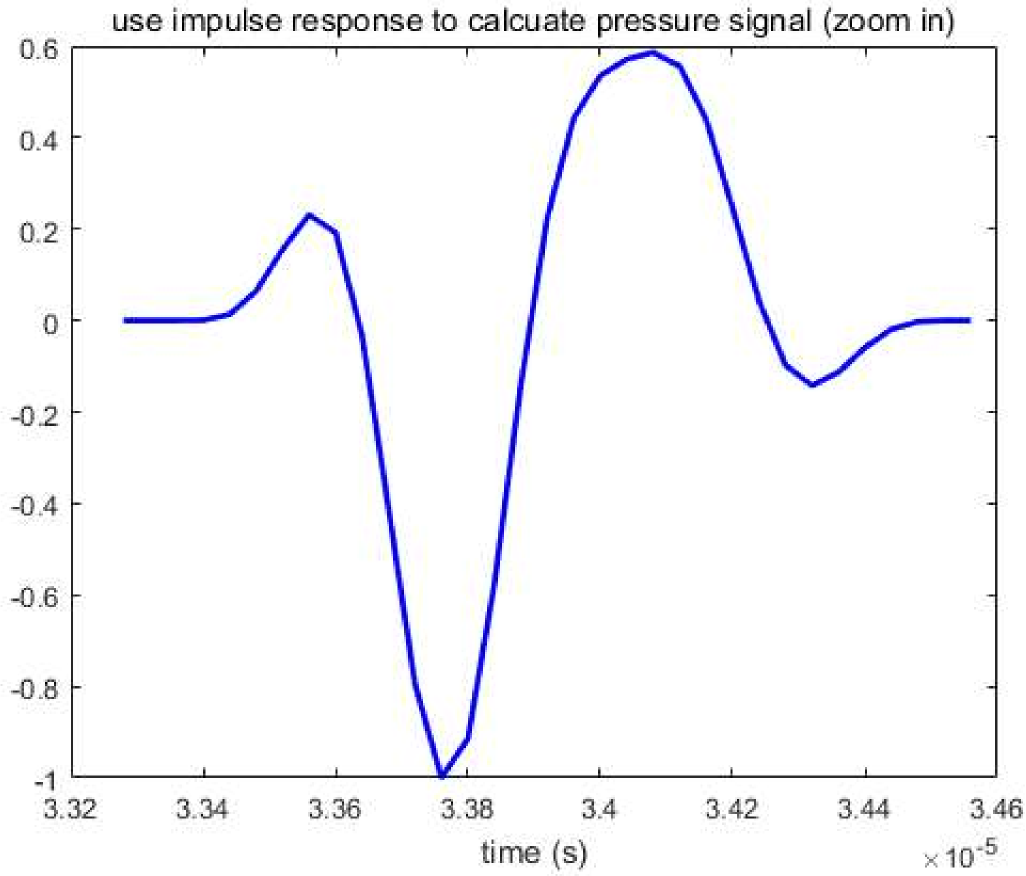
```
  end
  waitbar(m/length(x_1), h_WB);
end
t_1=toc;
close(h_WB)
n=find(R==0);
R(n)=NaN;

% figure
% %plot Normalized Green function frequency response
% plot(x_0, abs(H(:))/max(abs(H(:))),'b');
% xlabel("x axis");
% %xlabel("x axis (mm)");
% ylabel("Impulse_Response_Mag");
%
% Theta = x_0/z_0;
% D_theta = 2*besselj(1,k(1)*a*sin(Theta))./(k(1)*a*sin(Theta));
% hold on
% plot(x_0, D_theta, 'r');


%impulse response
X= fft(s);
Y_Phi = X(1:length(H(1,:))).*H(1,:);
Phi = ifftmmv(Y_Phi, N_FFT);
t=(0:length(Phi) - 1)/f_Sample; % time axes
n_b = floor(min(R(:))/c_0*f_Sample)-1;
n_e = ceil(max(R(:))/c_0*f_Sample)+1;
n=n_b + (length(t_s) + (n_e - n_b + 1) - 1) - 1;

figure;
plot(t,s,'r');
title("original source signal");
xlabel("time (s)");

figure;
plot(t(n_b:n), Phi(n_b:n)/max(abs(Phi(n_b:n))), 'b-', 'LineWidth', 2);
title("use impulse response to calcuate pressure signal (zoom in)");
xlabel("time (s)");

figure;
plot(t(n_b)+(0:length(Phi(n_b:end))-1)/f_Sample, real(Phi(n_b:end))/max(abs(Phi(n_b:end)))
);
title("use convolusion method to calcuate pressure signal");
xlabel("time (s)");

% convolusion method
h = ifft(H, N_FFT);
figure;
plot(real(h(n_b:n_e)/max(abs(h))));
title("impulse response of huygen's method h(n)");

Phi_2 = conv(s,h(n_b:n_e));
t=(0:length(Phi_2) - 1)/f_Sample; % time axes
figure;
plot(t(n_b)+(0:length(Phi_2)-1)/f_Sample, real(Phi_2)/max(abs(Phi_2)));
title("use convolusion method to calcuate pressure signal");
xlabel("time (s)");
% for q=1:length(x_0)
% %    plot(f_c, abs(H(q,:))/max(H(q,:)));
%     plot(f_c, log10(abs(H(q,:))));
```

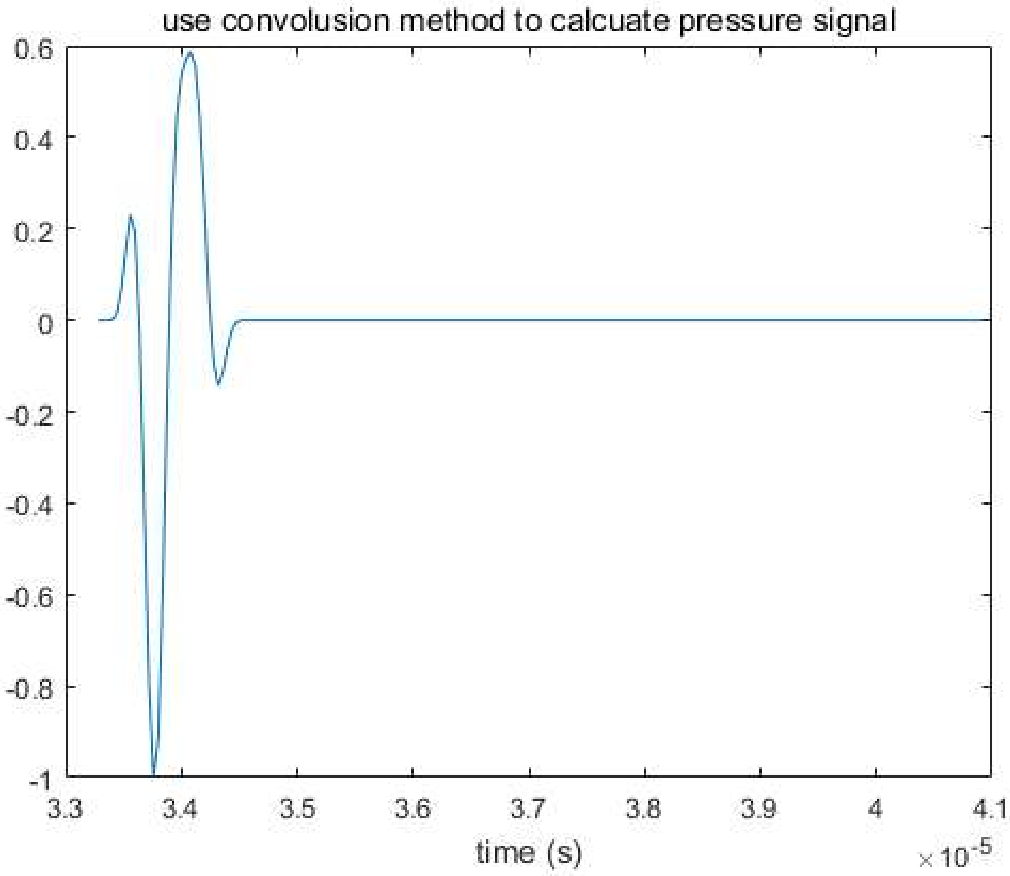
```
%     xlabel("frequency");
%     title_s = sprintf("position=%d", x_0(q));
%     title(title_s);
%     hold on;
%     pause;
% end

[h_sir, t_0] = sirmmv(a,x_0,y_0,z_0,f_Sample,c_0);
figure;
plot(h_sir);
title("use sirmmv to get the impulse response");

Phi_3 = conv(s,h_sir);
figure;
plot(t_0 + (0:length(Phi_3) - 1)/f_Sample, Phi_3/max(abs(Phi_3)), 'm-.', 'LineWidth', 2);
title("use sirmmv to calculate pressure signal");
xlabel("time (s)");


% field II
field_init(0);
set_field('c',c_0);
set_sampling(f_Sample);
N_div = 10; % number of lateral division
TxAperature=xdc_piston(a, 2*a/N_div);

xdc_excitation(TxAperature, s);
[Phi_4, t_field_b]=calc_hp(TxAperature, [x_0 y_0 z_0]);
figure;
plot(t_field_b + (0:length(Phi_4)-1)/f_Sample, Phi_4/max(Phi_4));
title("use field-ii to calculate pressure signal");
xlabel("time (s)");

% xvector=ones(N_FFT,1)*x_0;
% yvector=ones(N_FFT,1)*y_0;
% zvector=transpose(linspace(1e-3,z_0,N_FFT));
%
% positions=[xvector yvector zvector];
% [Phi_5, t_field_b_5]=calc_hp(TxAperature, positions);
% ft=figure;
% for i=1:N_FFT
%     clf(ft);
%     plot(t_field_b_5+(0:length(Phi_5(:,i))-1)/f_Sample, Phi_5(:,i)/max(Phi_5(:)));
%     ylim([min(Phi_5(:))/max(Phi_5(:))*1.2 1.2]);
%     stitle=sprintf("use field-ii to calculate pressure signal at z axis from 1mm to 50mm
, at %dmm",zvector(i)*1000);
%     title(stitle);
%     xlabel("time (s)");
%     pause(0.1);
% end

xvector=transpose(linspace(-2*a,2*a,N_FFT));
yvector=ones(N_FFT,1)*y_0;
zvector=ones(N_FFT,1)*z_0;

positions=[xvector yvector zvector];
[Phi_5, t_field_b_5]=calc_hp(TxAperature, positions);
% ft=figure;
% for i=1:N_FFT
%     clf(ft);
%     plot(t_field_b_5+(0:length(Phi_5(:,i))-1)/f_Sample, Phi_5(:,i)/max(Phi_5(:)));
```

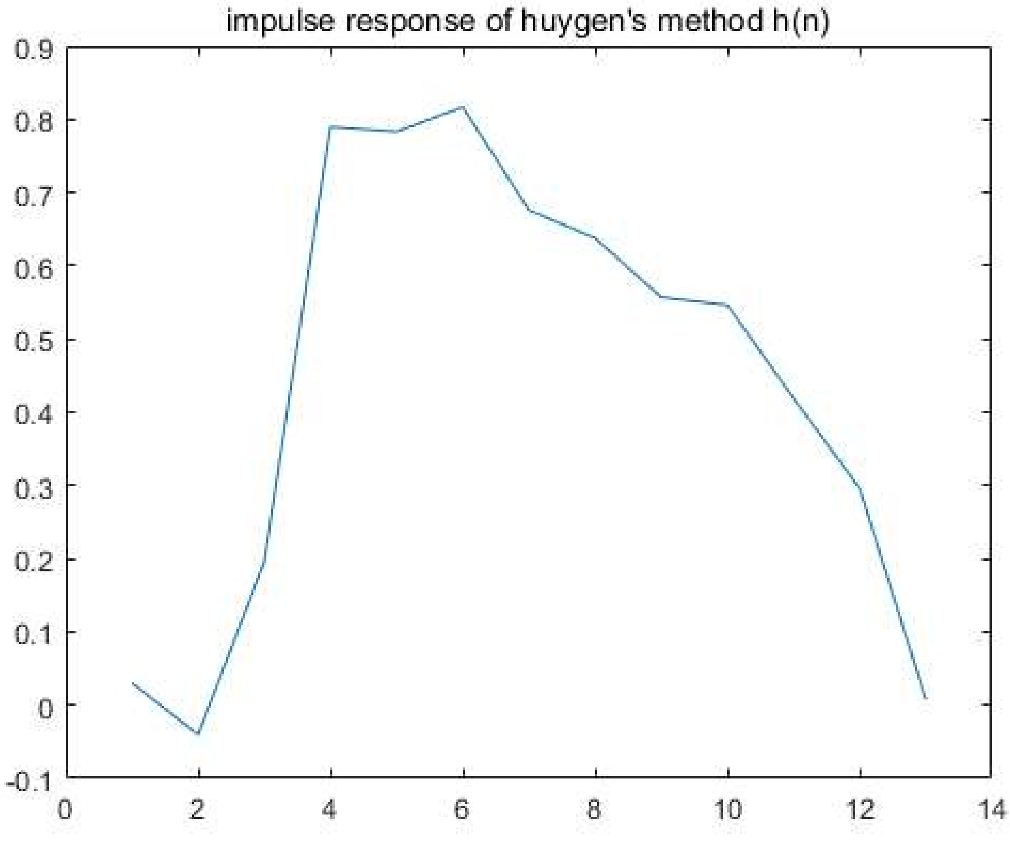
```
%     ylim([min(Phi_5(:))/max(Phi_5(:))*1.2 1.2]);
%     stitle=sprintf("use field-ii to calculate pressure signal at x axis from %dmm to %dm
m, at %dmm",-2*a*1000,2*a*1000,xvector(i)*1000);
%     title(stitle);
%     xlabel("time (s)");
%     pause(0.1);
% end

t_vector = t_field_b_5+(0:length(Phi_5(:,1))-1)/f_Sample;
figure;
for i=1:N_FFT
    plot3(xvector(i)*ones(1,128),t_vector(1:128),Phi_5(1:128,i));
    hold on;
end
xlabel("x axis in mm");
ylabel("t axis in s");
stitle=sprintf("use field-ii to calculate pressure signal at x axis from %dmm to %dmm",-2*
a*1000,2*a*1000);
title(stitle);
```

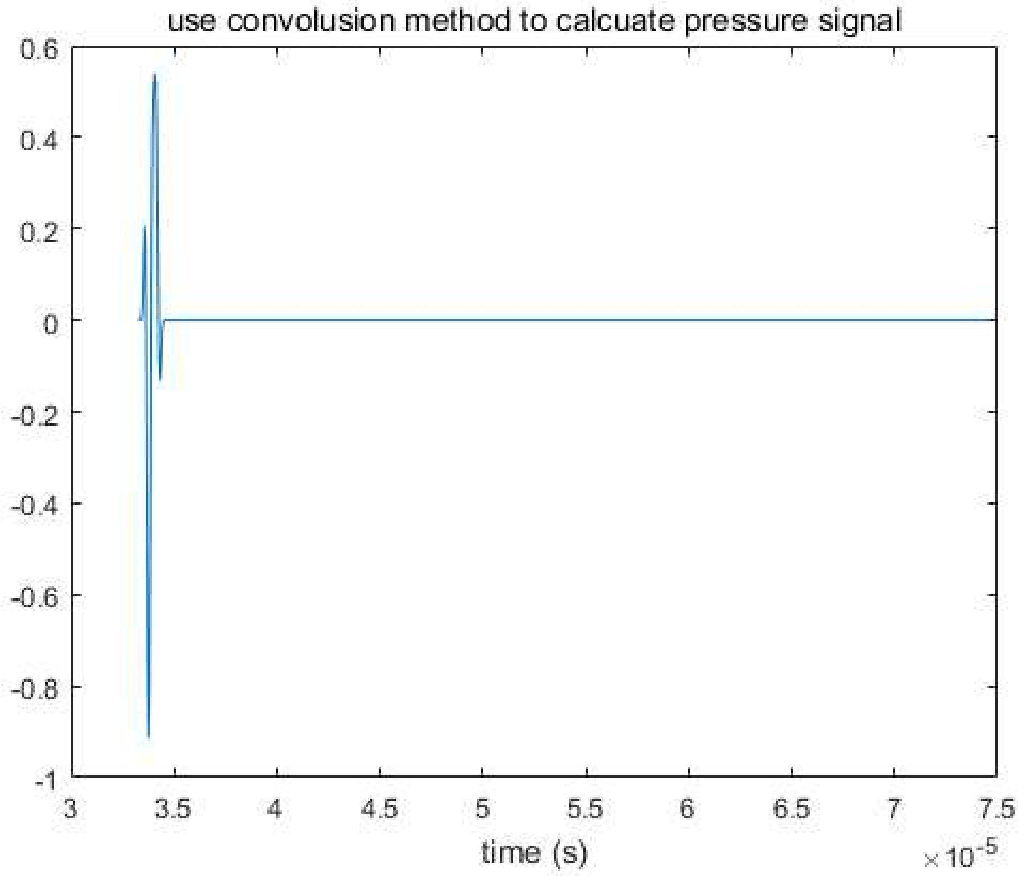
```
      *-------------------------------------------------------------*
      *                                                             *
      *                       F I E L D   I I                       *
      *                                                             *
      *               Simulator for ultrasound systems              *
      *                                                             *
      *              Copyright by Joergen Arendt Jensen             *
      *     Version 3.24, May 12, 2014 (Matlab 8.20 version)        *
      *                    Web-site: field-ii.dk                    *
      *                                                             *
      *      This is citationware. Note the terms and conditions    *
      *      for use on the web-site at:                            *
      *                 field-ii.dk/?copyright.html                 *
      *   It is illegal to use this program, if the rules in the    *
      *   copyright statement is not followed.                      *
      *-------------------------------------------------------------*
Warning:  Remember to set all pulses in apertures for the new sampling frequency
```

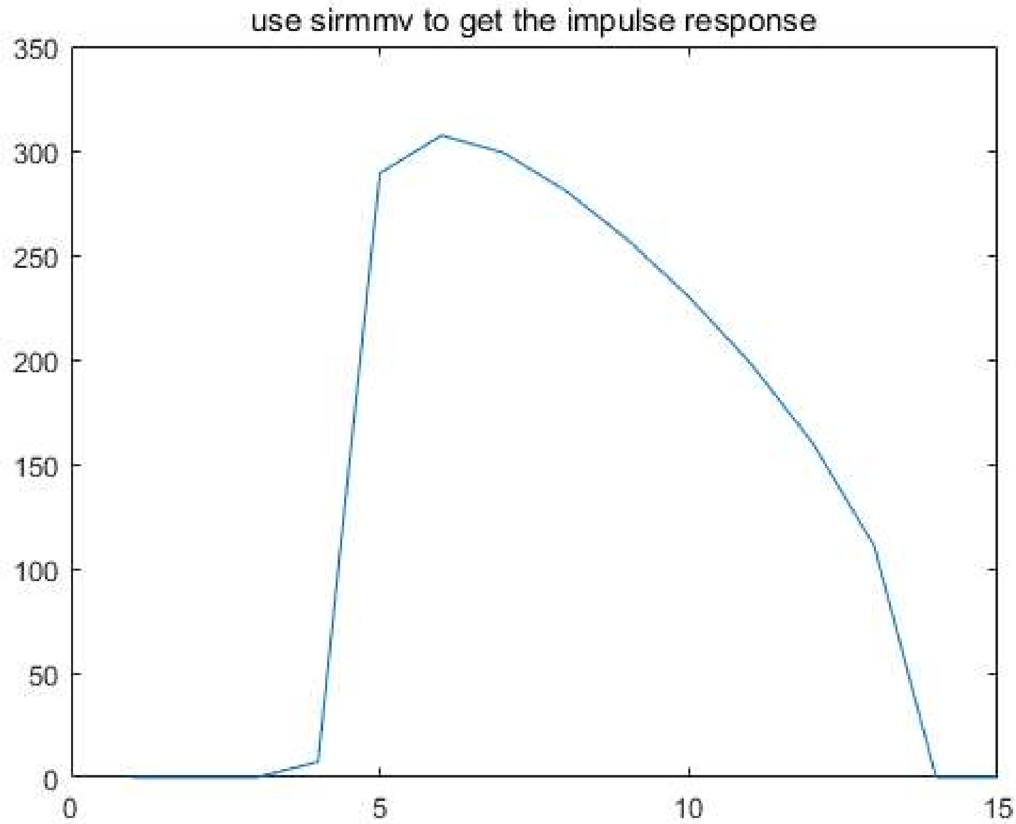

original source signal
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
time (s)
$\times 10^{-5}$

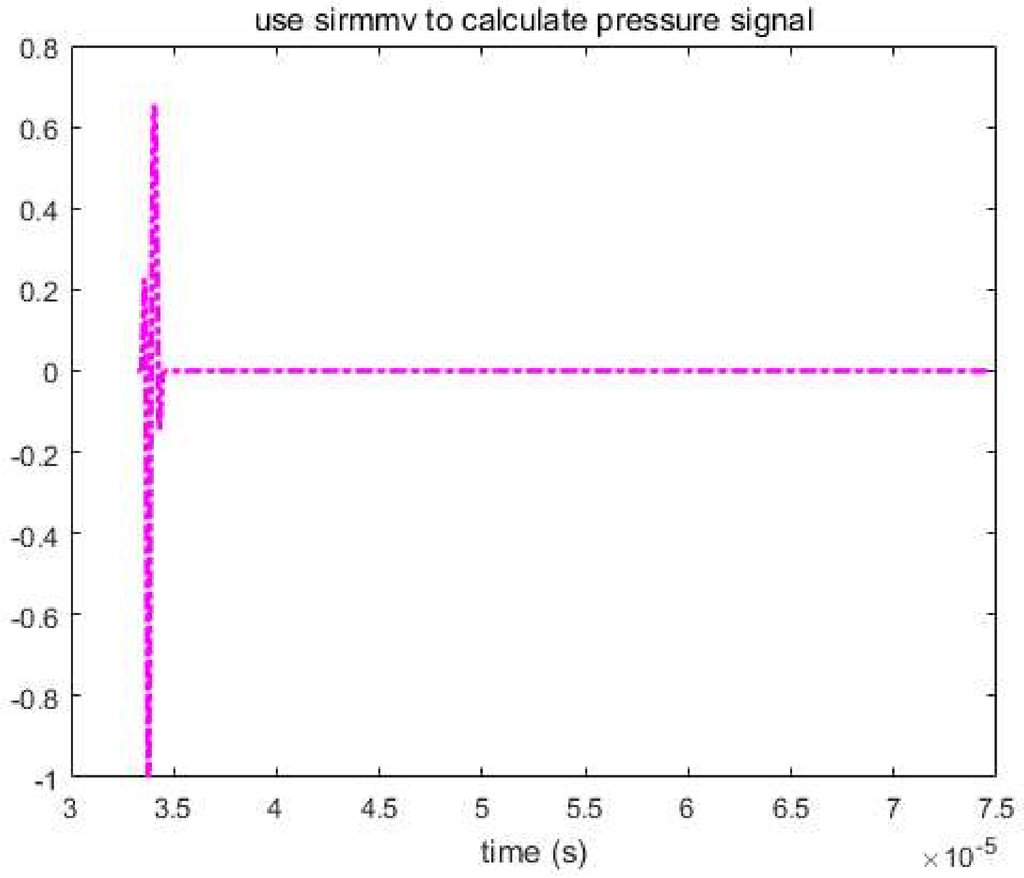

use impulse response to calcuate pressure signal (zoom in)
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
3.32
3.34
3.36
3.38
3.4
3.42
3.44
3.46
time (s)
$\times 10^{-5}$

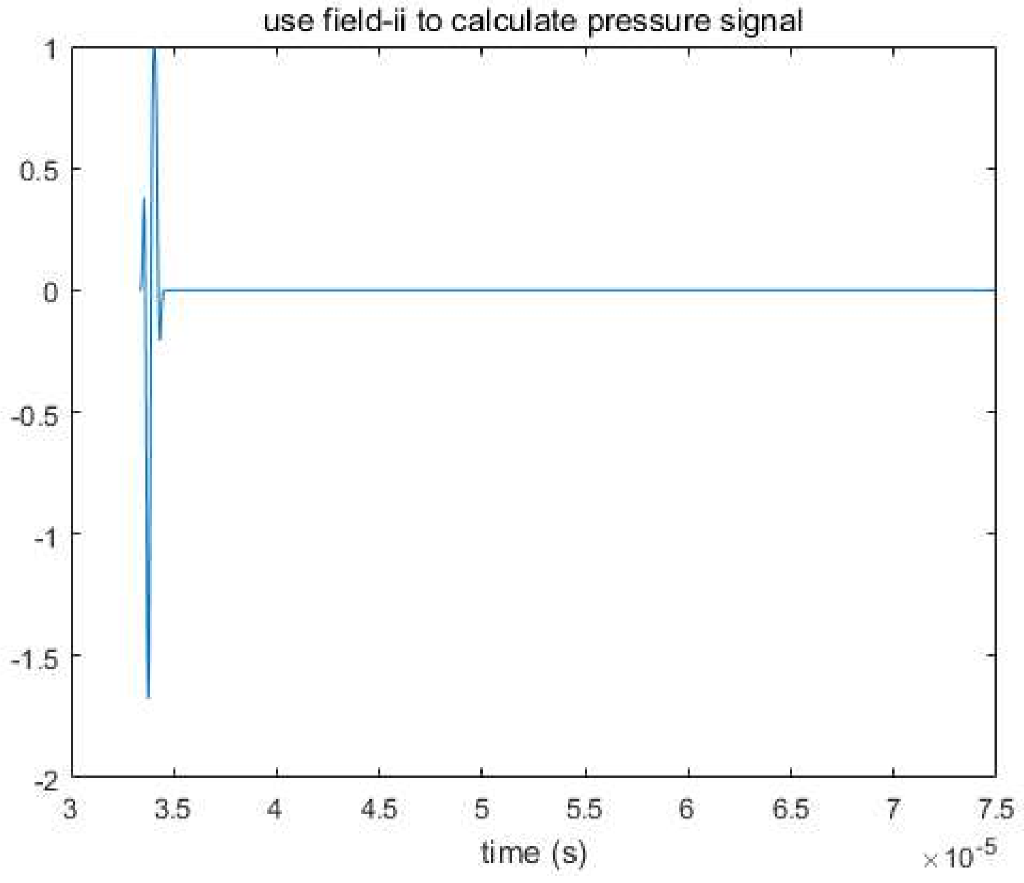

use convolusion method to calcuate pressure signal
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
3.3
3.4
3.5
3.6
3.7
3.8
3.9
4
4.1
time (s)
×10-5

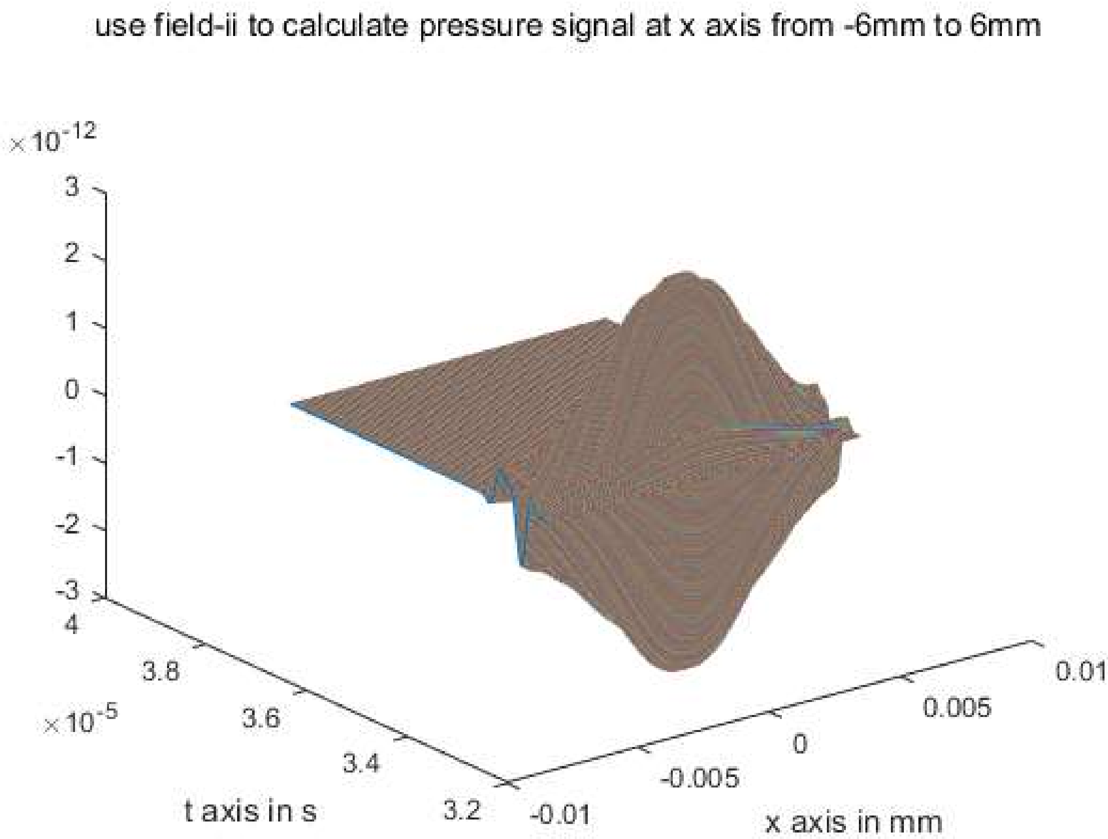

impulse response of huygen's method h(n)
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
-0.1
0
2
4
6
8
10
12
14


use convolusion method to calcuate pressure signal
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
3
3.5
4
4.5
5
5.5
6
6.5
7
7.5
time (s)
×10^-5


use sirmmv to get the impulse response
350
300
250
200
150
100
50
0
0
5
10
15


use sirmmv to calculate pressure signal
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
3
3.5
4
4.5
5
5.5
6
6.5
7
7.5
time (s)
$\times 10^{-5}$


use field-ii to calculate pressure signal
1
0.5
0
-0.5
-1
-1.5
-2
3
3.5
4
4.5
5
5.5
6
6.5
7
7.5
time (s)
$\times 10^{-5}$


use field-ii to calculate pressure signal at x axis from -6mm to 6mm
×10^-12
3
2
1
0
-1
-2
-3
4
3.8
×10^-5
3.6
3.4
3.2
t axis in s
-0.01
-0.005
0
0.005
0.01
x axis in mm