```matlab
%%
% Author: Robert Nuster
% December-2017
% modified by Franz Taffner
% January-2018
%%

close all
clear all

fs = 30; % fontsize value

%% Definitions

N = 1e5-1;          % datapoint value

cs = 1500;         % speed of sound (m/s)
a = 10e-6;         % radius of spherical source (m)
z = 10e-3;         % source-detection distance (m)

F = 200;            % fluence [J/m]
mua = 20000;        % absobtion coefficient [m^-1]
gamma = 0.11;       % grueneisen Parameter

p0 = F * mua * gamma;      % initial pressure rise (Pa)

tmax = 2 * z/cs;
t = linspace(0,tmax,N);    % time vector
dt = t(2)-t(1);
freq = 0:1/max(t):(N-1)/max(t); % corresponding frequency vector
freq = freq-max(freq)/2;      % defining positive and negativ frequencies


%% Simulate signal of sphere with delta peak excitation

% analytical equation for a spherical source detected by a point like
% sensor
pP = 0.5*(z-cs*t)/z.*(cs*t>(z-a) & cs*t<(z+a));

figure(1)

plot(t*cs*1e3, pP);
title('pressure signal: \delta peak excitation');
xlabel('distance in mm','fontsize',fs);
ylabel('N.A.','fontsize',fs);
set(gca,'fontsize',fs);
xlim([z-5*a z+5*a]*1e3);
grid on

%% Simulate gaussian temporal profil of excitation laser pulse

tp = 10e-9;
t0 = tmax/2;

sigma = tp./(2*sqrt(2*log(2)));
LP = exp(-(t-t0).^2/(2*sigma^2));
tshift = t-tmax/2;

figure(2)

plot(tshift,LP);
title('Temporal profil of laser pulse');
xlabel('time in ns','fontsize',fs);
ylabel('N.A.','fontsize',fs);
set(gca,'fontsize',fs);
xlim([-2*tp 2*tp]);
grid on

```

```matlab

%% Simulate signal of sphere with finite excitation pulse duration

SigSphere = convn(pP,LP,'same');  % convolution of the timedomain pressure
                                  % signal with the time domain temporal
                                  % laser pulseprofile

figure(3)

plot(t*cs*1e3,pP,t*cs*1e3,SigSphere);
title('Ideal measured signal');
legend('\delta excit','finite pulse excit.');
xlabel('distance in mm','fontsize',fs);
ylabel('N.A.','fontsize',fs);
set(gca,'fontsize',fs);
xlim([z-3*a z+3*a]*1e3);
grid on

%% Spectrum of the spherical signal

FSigC=(fftshift(fft((SigSphere))));
FSig=abs(FSigC);
FSig=FSig./max(FSig);

figure(4)

plot(freq/1e6,abs(FSig));
title('Frequency spectrum of ideal Signal');
xlabel('f in MHz','fontsize',fs);
ylabel('N.A.','fontsize',fs);
set(gca,'fontsize',fs);
xlim([-200 200]);
grid on

%% Calculate Sensor Transfer function assumed as gaussian function

fc = 50e6;           % sensor center frequency (Hz)
BW = 0.7*fc;         % sensor bandwidth (Hz)
sigf = BW./(2*sqrt(2*log(2)));
AS = exp(-(abs(freq)-fc).^2/(2*sigf^2)); % amplitude spectrum of sensor
Fract = 100;
AS(AS<max(AS)/Fract) = max(AS)/Fract;

phifilt = imag(hilbert(log(AS)));   % phase of sensor response function
CF = AS.*(cos(phifilt) - 1i*sin(phifilt)); % complex fuction of
                                           % causal filter

MS_Spec = (FSigC).*CF;    % multiplication of the signal frequency spectrum
                        % with the transducer transferfunction

figure(5)

plot(freq/1e6,FSig./max(FSig),...
    freq/1e6,AS,freq/1e6,...
    abs(MS_Spec)./max(abs(MS_Spec)), 'linewidth', 5);
title('Frequency domain representation');
legend('Pressure wave spectrum','Transducer transferfunction',...
    'Resulting signal spectrum');
xlabel('f in MHz','fontsize',fs);
ylabel('N.A.','fontsize',fs);
set(gca,'fontsize',fs);
xlim([-200 200])
grid on

%% Measured signal in the time domain

MeasSig=real((ifft(ifftshift(MS_Spec))));

```

```
figure(6)

plot(cs*t*1e3,p0*SigSphere,...
     cs*t*1e3,p0*MeasSig,'linewidth', 5);

title({'Comparison of point detector temporal signal with';...
    'spherical detector signal'});
legend('Pointlike detector','Spherical detector');
xlabel('c_st in mm','fontsize',fs);
ylabel('p in Pa','fontsize',fs);
set(gca,'fontsize',fs);
xlim([z-10*a z+50*a]*1e3);
grid on
```