NBE-E4310 D - Biomedical Ultrasonics 2023

This document contains the solutions for the Execise 2 of the BMUS course. The student solutions should have been posted no later than 13:00 on October 23rd.

Please, note that the code might be different than yours and might lead to slightly different solutions. A differently solved problem can sometimes be correct. However, this example should serve as a reference on how the exercise solutions should look like.

TASK 1 (12 points)

Observe a rectangular region on xy plane with coordinates of the corners (-10,0), (+10,0), (-10,-10) and (+10,-10), the units being mm. Consider an array of 11 monopole sources oscillating radially, the sources positioned linearly at 0.5 mm spacing from (-2.5,0) to (+2.5,0). Since in the exercise you are calculating is a 2D simplification, assume that any sources are line sources unless else is assumed. Neglect any attenuation mechanisms, geometric or dissipation. The speed of sound is that in soft tissue. Use complex pressure in your studies leading to your solution. Assume continuous wave. Explain and justify in your own words each step of your calculation.

a. What needs to be the frequency to have destructive interference of the wave along the x axis? (2p)

In order to have destructive interference along the axis that contains the monopoles (x axis), we need the monopoles to be spaced in a way that the superposition of their respective waves results into 0. This happens when the monopoles are separated by $\lambda/2$. Considering that the spacing given in the exercise is 0.5 mm, we need to remember:

$$c = \lambda f \tag{1}$$

where c is the speed of sound in the medium of propoagation ($c_{tissue} = 1500m/s$) and f is the frequency that we are trying to find. Therefore, the frequency is 1.5 MHz.

b. For this frequency you calculated in *a)*, present the momentary pressure generated by the sound sources at t = 0 s. (4p)

Now, with the frequency found previously, we will generate the momentary pressure at $t = 0$ s, remembering from the slides of the course that:

$$p(r) = \hat{p}e^{i(wt-kr)} \tag{2}$$

where $\hat{p}$ is the pressure amplitude, w is the angular frequency that depends on f and k is the wave number that depends on the λ. From Equation 2 and considering the 11 monopoles necessary for this exercise, we develop the following code:

```
% Parameters
c = 1500;
f = 1.5e6;
T = 1/f;
t = 0:T/100:T;
x = c*t;
t = 0;
%x = 0;
lambda = c/f;
k = 2*pi / lambda;
p_hat_r = 1e6;
omega = 2*pi*f;
phi = pi()/2;

% Coordinates Task 1
fromx = -10e-3;
tox = 10e-3;
fromy = -10e-3;
toy = 10e-3;
```

```
[XX,YY] = meshgrid([fromx:lambda/8:tox]',[fromy:lambda/8:toy]');

%Pressure equation
P_right_tot = zeros(size(XX));

%Construct the wave (without attenuation):
figure(1)

for jj = -5:5
    offset = lambda/2;
    r = sqrt((XX+jj*offset).^2 + YY.^2);
    P_right = p_hat_r * exp(j*(omega*t-k*r));
    P_right_tot = P_right_tot + P_right;

    imagesc(real((P_right_tot)))
    axis image
    colorbar
    a = colorbar;
    a.Label.String = 'p(Pa)';
end

drawnow
```

Listing 1: Plotting the momentary pressure at t

The result of the code is shown in Figure 1.

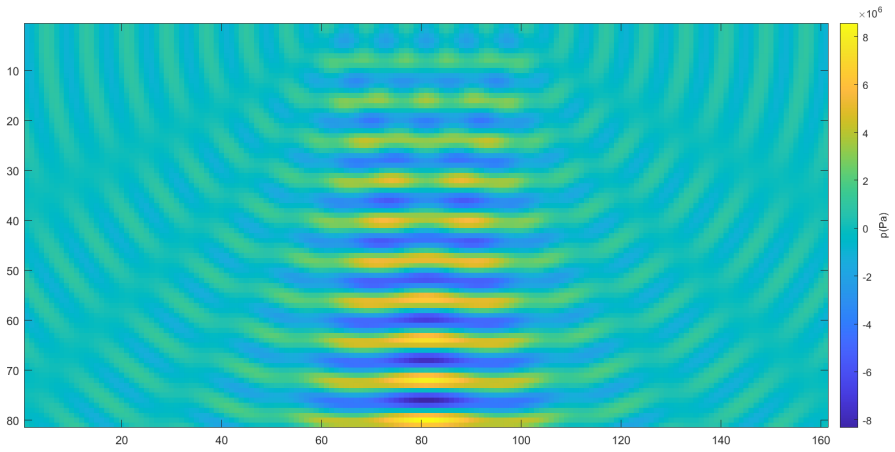

Figura 1: Momentary pressure at t=0 without attenuation.

c. For the field of task *b)*, present the pressure field, *i.e.* the envelope. (2 p)

Now, we just need to plot the absolute pressure from the previous code. Following from the previous code shown, we add:

```
figure(2)

for jj = -5:5
    offset = lambda/2;
    r = sqrt((XX+jj*offset).^2 + YY.^2);
    P_right = p_hat_r * exp(j*(omega*t-k*r));
    P_right_tot = P_right_tot + P_right;

    imagesc(abs((P_right_tot)))
    axis image
    colorbar
    a = colorbar;
    a.Label.String = 'p(Pa)';
end

drawnow
```

Listing 2: Plotting the absolute pressure at t

And the result of the code is shown in Figure 2.

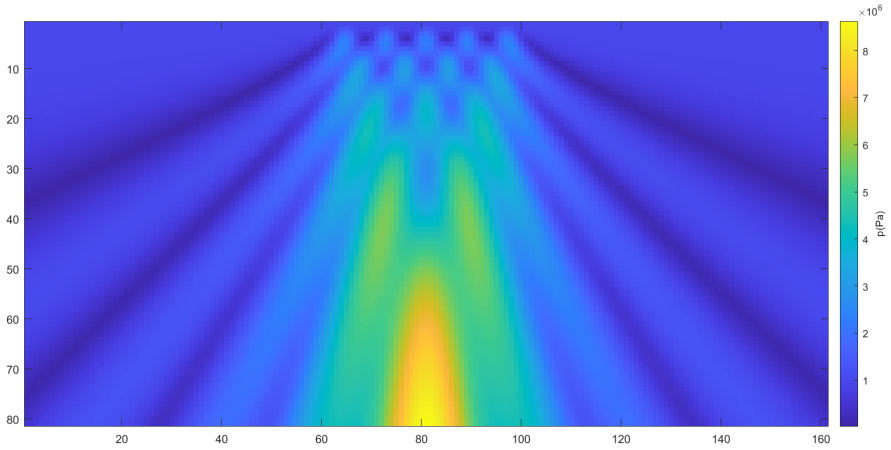

Figura 2: Pressure field, *i.e.* envelope, at t=0 without attenuation.

d. Recalculate the maps of *b)* and *c)* by considering geometric attenuation for your source (2 p)

In this exercise, we just need to add the attenuation factor in the expression of the momentary pressure. In this case, as it is not specified, we will go for the example of the planar pressure, that it is defined as:

$$p(r) = \hat{p}e^{i(wt-kr)}e^{-(\alpha * r)} \tag{3}$$

where in this case α was the attenuation coefficient. For this particular case, we have taken the attenuation coefficient of the kidney ($\alpha_{kidney}^{1.5MHz} = 150dB/m$). This value has been found following:

$$\alpha = af^b \tag{4}$$

where a and b are the coefficients found in *Slide 33 of the Lecture Slides 01,02.*. In the case of the kidney, both coefficients are simply 1.

From the Equation 3, we add to the code the following lines:

```
%Re-calculate the momentary pressure with the planar attenuation:
figure(3)

for jj = -5:5
    offset = lambda/2;
    r = sqrt((XX+jj*offset).^2 + YY.^2);
    alpha = 150; %kidney attenuation coefficient
    P_right = p_hat_r * exp(j*(-k*r)).*exp(-alpha.*r);
    P_right_tot = P_right_tot + P_right;

    imagesc(real((P_right_tot)))
    axis image
    colorbar
    a = colorbar;
    a.Label.String = 'p(Pa)';
end

% And exactly the same for the envelope
figure(4)

for jj = -5:5
    offset = lambda/2;
    r = sqrt((XX+jj*offset).^2 + YY.^2);
    alpha = 150; %kidney attenuation coefficient
    P_right = p_hat_r * exp(j*(-k*r)).*exp(-alpha.*r);
    P_right_tot = P_right_tot + P_right;
```

```
27
28      imagesc(abs((P_right_tot)))
29      axis image
30      colorbar
31      a = colorbar;
32      a.Label.String = 'p(Pa)';
33  end
34
35  drawnow
```

Listing 3: Plotting the momentary and absolute pressure at t

And the result of the code is shown in Figura 3 and 4.

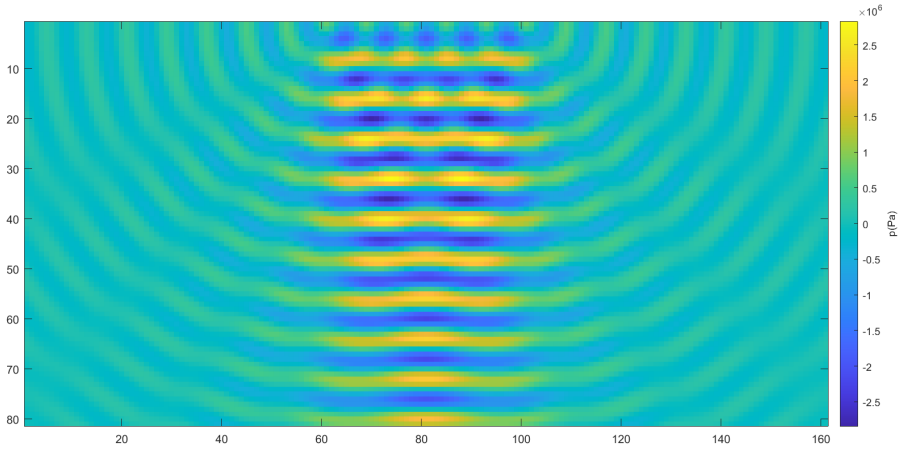

Figura 3: Momentary pressure at t=0 with planar attenuation.

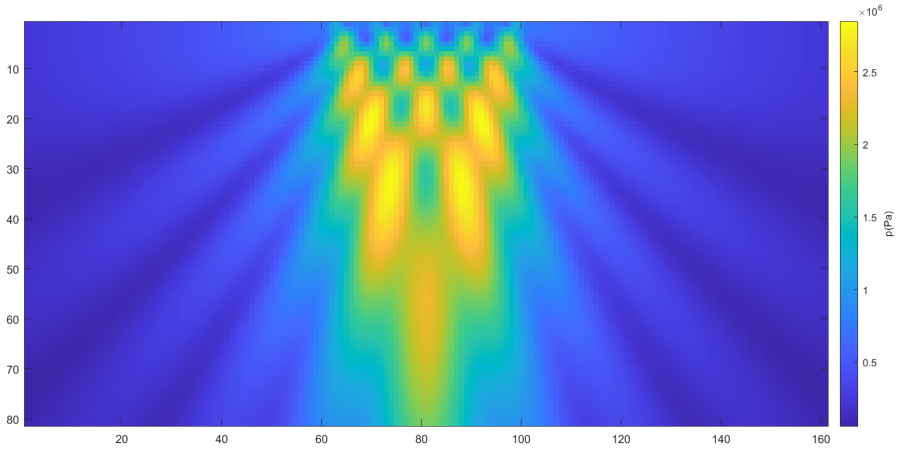

Figura 4: Pressure field, *i.e.* envelope, at t=0 with planar attenuation.

e. Recalculate the maps of *d)* by now considering the sources being point sources. (2p)

In this exercise, we just need to add the point source attenuation in the expression of the momentary pressure. The point source attenuation was a result from the previous exercise and the resulting momentary pressure follows:

$$p(r) = \frac{1}{\sqrt{r}}\hat{p}e^{i(wt-kr)}e^{-(\alpha * r)} \tag{5}$$

where in this case the factor $\frac{1}{\sqrt{r}}$ was added to show the energy loss due to the 3D expansion of the

pressure fronts.

From the Equation 5, we add to the code, adjusting the r vector in order to divide by it, in the following lines:

```
%Re-calculate the momentary pressure with the planar attenuation:
figure(5)

for jj = -5:5
    xuse = (XX-sources(j,1)).^2;
    yuse = (YY-sources(j,2)).^2;
    r = sqrt((xuse).^2 + (yuse).^2);

    P_right = p_hat_r.*exp(-alpha*r).*exp(j*(-k*r))/sqrt(r);
    P_right_tot = P_right_tot + P_right;

    imagesc(real((P_right_tot)))
    axis image
    colorbar
    a = colorbar;
    a.Label.String = 'p(Pa)';
end

% And exactly the same for the envelope

figure(6)
for jj = -5:5
    xuse = (XX-sources(j,1)).^2;
    yuse = (YY-sources(j,2)).^2;
    r = sqrt((xuse).^2 + (yuse).^2);

    P_right = p_hat_r.*exp(-alpha*r).*exp(j*(-k*r))/sqrt(r);
    P_right_tot = P_right_tot + P_right;

    imagesc(abs((P_right_tot)))
    axis image
    colorbar
    a = colorbar;
    a.Label.String = 'p(Pa)';
end

drawnow
```

Listing 4: Plotting the momentary and absolute pressure at t

And the result of the code is shown in Figure 5 and 6.

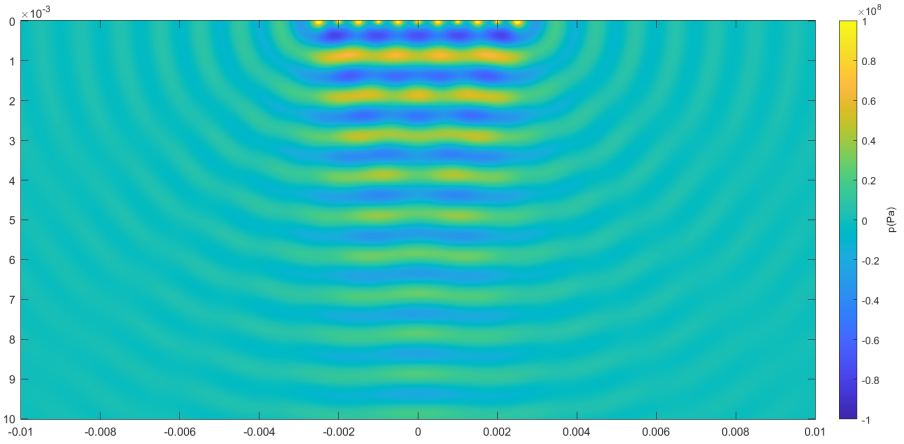

Figura 5: Momentary pressure at t=0 with point source attenuation.

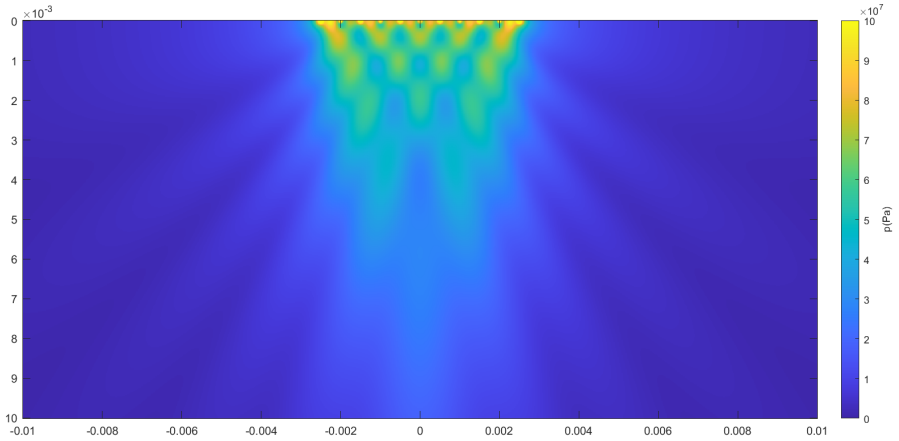

Figura 6: Pressure field, *i.e.* envelope, at t=0 with point source attenuation.

TASK 2 (10 points)

Generate a *'dense'* array of monopole sources, where the source spacing must be of, at least, $< \lambda/8$. The sources are placed at a distance 20 mm from origin of your xy-coordinates, positioned symmetrically to x = 0, where y > 0, forming a circularly symmetric transducer with a 90 degrees opening angle. Consider a 2D simplification and a rectangular region on the xy plane with coordinates of the corners (-25,25), (25,25), (-25,-10) and (+25,-10). The frequency is 1 MHz and the medium is water. Neglect any attenuation mechanisms, geometric or dissipation. Assume continuous wave. Explain and justify in your own words each step of your calucaltion.

a. For the frequency given, present the momentary real pressure generated by the sound sources at t = 0 s. (4p)

Now, this task is very similar to Task 1. The only difference is that the point sources need to be arranged in a semi-circular array. In this case, the 'dense' point sources array will generate a field that is similar to the one produced by a high-intensity focused ultrasound (HIFU). Therefore, we just need to tweak the original code in a way that allows us to plot the points in a curved line with a certain aperture. To do that, we generate the following code:

```
f = 1e06;
c = 1540;
lambda = c/f;
p_hat_r = 1;
k = 2*pi/lambda;
t = 0;

%% Task 2a
% Parameters for the space and sources distribution
x = linspace(-25e-03, 25e-03, 1000);
y = linspace(-25e-03, 10e-03, 750);
r = 20e-03;
aperture = linspace(5*pi/4, 7*pi/4, 20);

% Space generation
[XX,YY] = meshgrid(x,y);

% Point sources generation
x_cartesian = r * cos(aperture);
y_cartesian = r * sin(aperture);
sources = [x_cartesian; y_cartesian];

% Making those point sources to generate the field
mom_p = zeros([length(y) length(x)]);
for j = 1:length(sources)
    xuse = (XX-sources(1,j)).^2;
    yuse = (YY-sources(2,j)).^2;
    d = sqrt(xuse + yuse);
```

```
    add_p = p_hat_r*exp(1j*(-k*d));
    mom_p = mom_p + add_p;
end

figure
imagesc(x,y,real(mom_p));
% set(gca, 'CLim', [0 10]);
colorbar
title("Momentary Pressure (Pa)")

drawnow
```

Listing 5: Plotting the momentary pressure at t

And from this code, we get the left image from Figure 7.

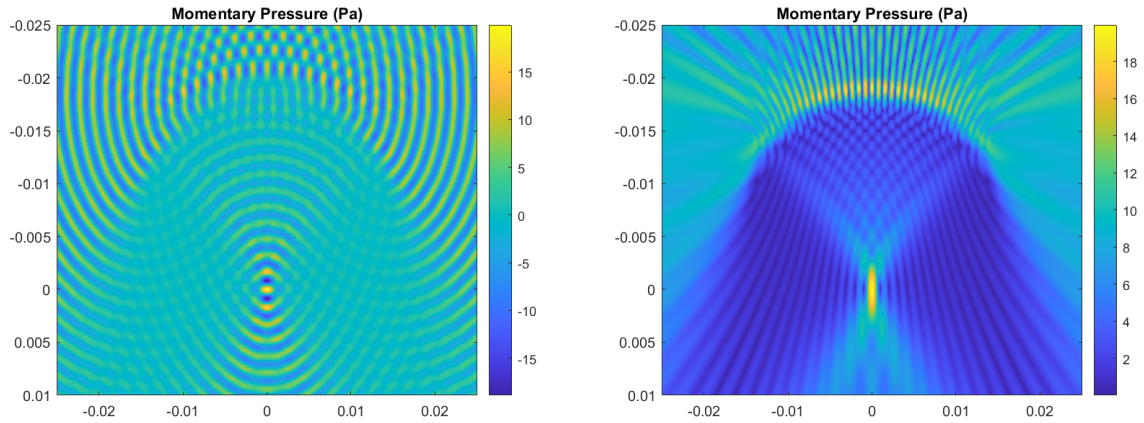

Figura 7: *(Left)*Momentary pressure and *(Right)* pressure field, *i.e.* envelope, at t=0 with a circular point source array.

b. For the field of task *a)*, present the pressure field, i.e. the envelope. (2 p)

Now, we just need to plot the absolute value of the pressure from the previous code instead of the real value. Therefore, we can add to the code the following:

```
figure
imagesc(x,y,abs(mom_p));
% set(gca, 'CLim', [0 10]);
colorbar
title("Momentary Pressure (Pa)")

drawnow
```

Listing 6: Plotting the momentary pressure at t

And from this code, we get the right image from Figure 7.

c. Recalculate the maps of *a)* and *b)* by considering geometric attenuation for your source, *i.e.* now your sound source being a line source (2 p)

In this section, we are asked about the linear attenuation. In the previous exercise, we found that the linear attenuation for the pressure is expressed as:

$$p(r) = \frac{1}{r}\hat{p}e^{i(wt-kr)}e^{-(\alpha * r)} \tag{6}$$

This is expressed in the code as:

```
f = 1e06;
c = 1540;
lambda = c/f;
p_hat_r = 1;
k = 2*pi/lambda;
```

```
t = 0;

%% Task 2a
% Parameters for the space and sources distribution
x = linspace(-25e-03, 25e-03, 1000);
y = linspace(-25e-03, 10e-03, 750);
r = 20e-03;
aperture = linspace(5*pi/4, 7*pi/4, 20);

% Space generation
[XX,YY] = meshgrid(x,y);

% Point sources generation
x_cartesian = r * cos(aperture);
y_cartesian = r * sin(aperture);
sources = [x_cartesian; y_cartesian];

% Making those point sources to generate the field
alpha = 150;
mom_p = zeros([length(y) length(x)]);
for j = 1:length(sources)
    xuse = (XX-sources(1,j)).^2;
    yuse = (YY-sources(2,j)).^2;
    d = sqrt(xuse + yuse);
    att = p_hat_r.*exp(-alpha*d);
    att = att ./ d;
    add_p = att.*exp(1j*(-k*d)); % complex pressure (t=0)
    mom_p = mom_p + add_p;
end

% figure
% imagesc(x,y,real(mom_p));
% set(gca, 'CLim', [0 100]);
% colorbar
% title("Momentary Pressure (Pa)")

figure
imagesc(x,y,abs(mom_p));
set(gca, 'CLim', [0 1000]);
colorbar
title("Momentary Pressure (Pa)")

drawnow
```

Listing 7: Plotting the momentary pressure at t

And from this code, we get Figure 8.

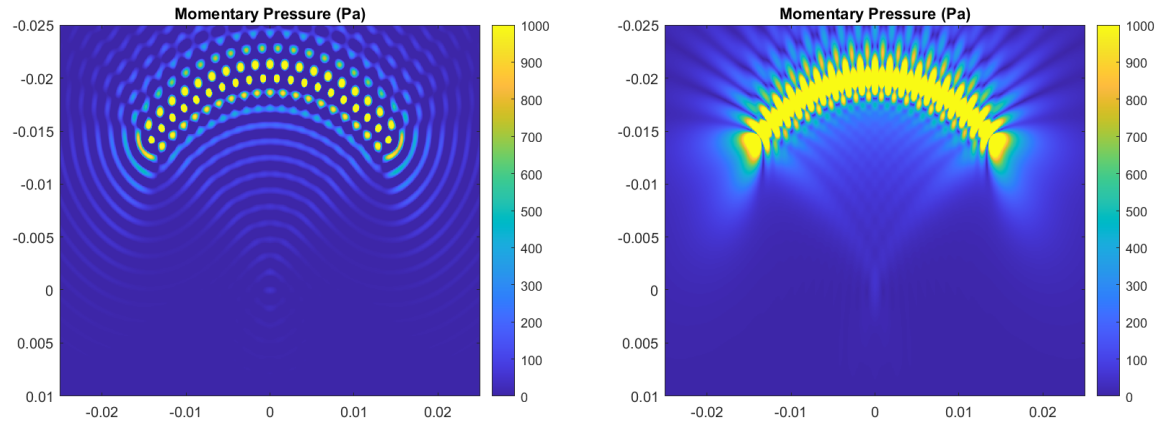

Figura 8: *(Left)*Momentary pressure and *(Right)* pressure field, *i.e.* envelope, at t=0 with a circular array considering linear source attenuation.

d. Recalculate the maps of *c)* by now considering the sources being point sources. (2p)

In this section, we are asked about the point source attenuation, given by Equation 5. This is expressed in the code as:

```
f = 1e06;
c = 1540;
lambda = c/f;
p_hat_r = 1;
k = 2*pi/lambda;
t = 0;

%% Task 2a
% Parameters for the space and sources distribution
x = linspace(-25e-03, 25e-03, 1000);
y = linspace(-25e-03, 10e-03, 750);
r = 20e-03;
aperture = linspace(5*pi/4, 7*pi/4, 20);

% Space generation
[XX,YY] = meshgrid(x,y);

% Point sources generation
x_cartesian = r * cos(aperture);
y_cartesian = r * sin(aperture);
sources = [x_cartesian; y_cartesian];

% Making those point sources to generate the field
alpha = 150;
mom_p = zeros([length(y) length(x)]);
for j = 1:length(sources)
    xuse = (XX-sources(1,j)).^2;
    yuse = (YY-sources(2,j)).^2;
    d = sqrt(xuse + yuse);
    att = p_hat_r.*exp(-alpha*d);
    att = att ./ sqrt(d);
    add_p = att.*exp(1j*(-k*d)); % complex pressure (t=0)
    mom_p = mom_p + add_p;
end

% figure
% imagesc(x,y,real(mom_p));
% set(gca, 'CLim', [0 100]);
% colorbar
% title("Momentary Pressure (Pa)")

figure
imagesc(x,y,abs(mom_p));
set(gca, 'CLim', [0 100]);
colorbar
title("Momentary Pressure (Pa)")

drawnow
```

Listing 8: Plotting the momentary pressure at t

And from this code, we get the Figure 9.

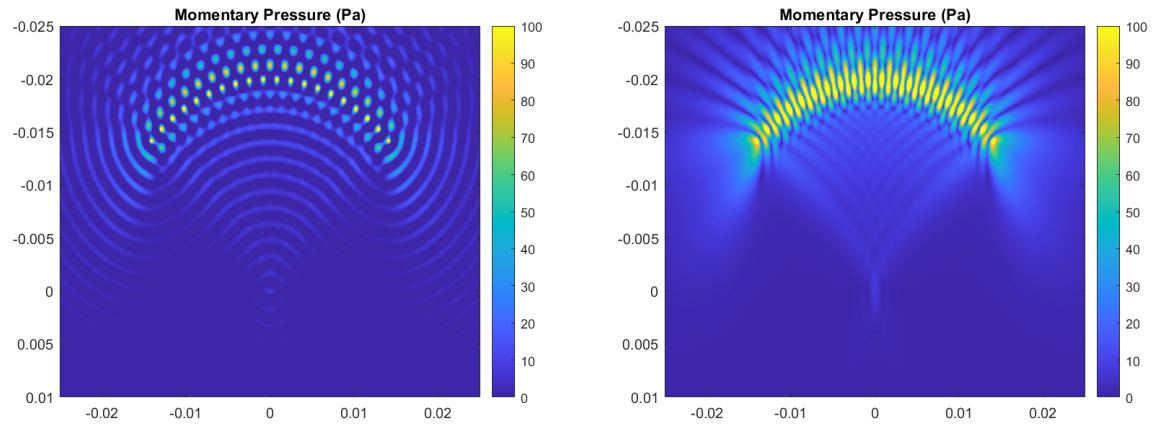

Figura 9: *(Left)*Momentary pressure and *(Right)* pressure field, *i.e.* envelope, at t=0 with a circular array considering point source attenuation.

TASK 3 (10 points)

On the lecture slides you were presented an animation (please, see the *.gif* file attached) of particles being within a space and oscillating back-and-forth parallel to the direction of the sound propagation in a linear harmonic fashion, forming a planar longitudinal wave traveling from left to right along the x-axis. Give the molecules within that space random rest positions, the space in the 2D simplification being an area rather than a volume. The corners of the space on the xy-plane are at coordinates (-10,5), (10,5), (-10,-5) and (10,-5). Assume the wavelength to be 2 mm and the medium to be water.

Recreate the animation of the black spots according to the instructions above. Your animation does not need to be exactly the same as the animation above (text, arrows or red objects are not expected), but still it needs to be physically correct. You may want to tune the amplitude and molecule density to see a visible compression and rarefaction and clear wave propagation of a continuous wave (10p)

First, we need to create the parameters for the wave and the space where the particles will be distributed. To do so, we use the following code:

```
% Parameters for the space and sources distribution
lambda = 2e-03;
c = 1500;
fig = c / lambda;
k = 2*pi/lambda;
x = linspace(-10e-03, 10e-03, 1000);
y = linspace(-5e-03, 5e-03, 500);
x_disp = max(x) - min(x);
y_disp = max(y) - min(y);
rng(646294);

% Space generation
[XX,YY] = meshgrid(x,y);

% Generate the displaced points
points = zeros([2 1000]);
for i = 1:length(points(1,:))
    points(1,i) = rand(1) * x_disp - (x_disp/2);
    points(2,i) = rand(1) * y_disp - (y_disp/2);
end

pointsstart = points;
fig = figure('Position',[100 100 500 200]);
set(fig, 'Visible', 'on');
hold on
for i = 1:length(points(1,:))
    plot(points(1,i), points(2,i), '.', color='black')
end
axis([min(x) max(x) min(y) max(y)]);

drawnow
```

Listing 9: Plotting the particle distribution that will have a perturbation.

And from this code, we get the Figure 10.

Then, we need to give the wave front parameters and add the displacement equation to give the motion to each dispersed particle, which follows:

$$\xi = \hat{\xi} e^{i(wt-kr)} \tag{7}$$

where ξ is the particle displacement. Therefore, now we just need to add a temporal vector values to get the animation running. To do so, we use the following code (although many variations from it would work anyway):

```
% Wave front parameters
T = 1/f;
cycles = 5;
tot = T*cycles;
distr = 20*cycles;
```

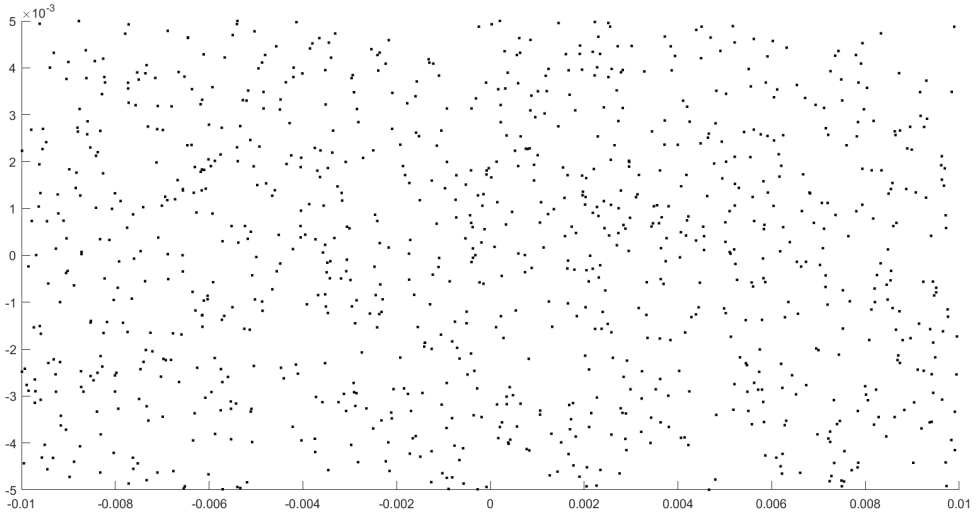

Figura 10: Particle field, with the wavefronts moving from left to right.

```
t = linspace(0, tot, distr);
disp_hat = 0.5e-03;

% Time display
for j = 1:length(t)
    for i = 1:length(points(1,:))
        points(1,i) = pointsstart(1,i) + disp_hat*cos(w*t(j)-k*pointsstart(1,i));
    end
    clf(fig); % clear the figure
    hold on
    % draw new positions
    for i = 1:length(points(1,:))
        % if (i == 15)
        %     plot(points(1,i), points(2,i), '.', color='red')
        % else
        plot(points(1,i), points(2,i), '.', color='black')
        % end
    end
    axis([min(x) max(x) min(y) max(y)]);

    drawnow % update fig

% And now we make the gif (there are a thousand different ways of doing this part):
    frame = getframe();
    im = frame2im(frame);
    [imind,cm] = rgb2ind(im,256);
    outfile = 'wavefronts.gif';
    if j==1
        imwrite(imind,cm,outfile,'gif','DelayTime',0.1,'loopcount',inf);
    else
        imwrite(imind,cm,outfile,'gif','DelayTime',0.1,'writemode','append');
    end
end
```

Listing 10: Plotting the momentary pressure at t

From the previous code, the gif presented in the file *wavefronts.gif* (attached to the solution .zip folder) is obtained.