Square Plate Chladni Pattern Simulation

Introduction to Computer Simulations

Nadine Soliman - nhs284

May 7, 2017

1 Introduction

Ernst Chladni is a German physicist and musician, who heavily contributed to the field of acoustics. His most well-known experiment, however, is the 2D resonance experiment that now holds his name, Chladni's plates or Chladni's figures. In his experiment, Chladni excited a metal plate by running a violin bow along the side of the plate, and used sand particles to visualize the standing wave produced. As the metal began to vibrate, it formed standing waves on its surface, which drove the sand particles to form characteristic patterns. By observing the position of the sand particles, Chladni was able to determine the positions of the nodes and anti-nodes on the surface. And, thus this enabled him to visualize the standing waves on the surface of the plate, that would otherwise not be visible to the naked eye. When the plate was driven at different frequencies, different patterns would form. Further, using different materials and different plate shapes, lead to different patterns and different standing waves. Chladni theorized his studies and formulated equations to describe the motion of the vibrating surfaces, and understanding the spread of sound waves within solids.

This simulation aims to provide a visualization of the experiment, and replicate its results computationally. In this simulation, we will explore the results of the experiment with regards to the square plate, as it provides a wide range of distinct pattern.

2 Equations

$$k_x = \frac{m\pi}{l_x} \tag{1}$$

$$k_y = \frac{n\pi}{l_y} \tag{2}$$

$$w = v\sqrt{k_x^2 + k_y^2} \tag{3}$$

$$Z(x, y, t) = (Asin(xk_x)sin(yk_y) + Bsin(k_x y)sin(k_y x))sin(wt) \tag{4}$$

The equations above were used to simulate the vibrations of the 2D square plate. Equations 1 and 2 correspond to the x and y components of the spring constant, where m, n are positive integers, and l_x, l_y are the lengths of the plate along the x and y axes. In this case, as we are only dealing with a square plate, so we know that $l_x = l_y$. Additionally, as m, n could take up any positive values, each pair of integers results in the formation of a different standing wave, and a characteristic chladni pattern.

Equation 3 describes the angular velocity of oscillation of the square plate, where v is the velocity of propagation of the wave, and k_x and k_y correspond to the spring constants described above.

Lastly, equation 4 combines all the above, and describes the height of a point on the square plate given its x and y positions, at a given point in time, where A and B correspond to the relative amplitudes of the wave. Each pair of A and B would result in a different standing wave, as it would assign different weights to each component of the oscillation along the square plate.

3 Numerical Method

To produce the simulation, the following numerical method was used.

3.1 Displaying the Oscillating Square Plate

$$\begin{aligned} z(x_{t+\Delta t}, y_{t+\Delta t}, t + \Delta t) = z(x_t, y_t, t) + (Asin(k_x x_{t+\Delta t})sin(k_y y_{t+\Delta t}) + \\ Bsin(k_x y_{t+\Delta t})sin(k_y x_{t+\Delta t}))sin(w(t + \Delta t)) \end{aligned} \tag{5}$$

3.2 Theoretical Solution

A theoretical solution was determined and graphed, so that the resultant sand patterns could be compared to it, so that the data could be analyzed. To do so, we equation 4 mentioned above at a specific time value $t = \frac{\pi}{2}$ for convenience, and determined the nodal points, where the sand particles would accumulate. Solving for the position of the nodes, is equivalent to solving for the zeros of the equation by the definition of a node.

$$Asin(k_x x)sin(k_y y) + Bsin(k_x y)sin(k_y x)) = 0 \tag{6}$$

3.3 Moving the Sand Particles

To move the sand particles on the spatial grid, a vector field function $f = -\nabla Z$ was defined for each point on the grid, and the individuals velocities, v_x, v_y were determined accordingly.

$$v_x(t + \Delta t) = v_x(t) + f(x, y, t + \Delta t) \tag{7}$$

$$v_y(t + \Delta t) = v_y(t) + f(x, y, t + \Delta t) \tag{8}$$

$$x(t + \Delta t) = x(t) + v_x(t + \Delta t) \tag{9}$$

$$y(t + \Delta t) = y(t) + v_y(t + \Delta t) \tag{10}$$

To prevent all the sand particles from moving all towards a single point on the plate, simple collisions were accounted for where if two particles collide, they bounce off each other at a set velocity. For instance, if two particles i, j were to collide,

$$x_i(t + \Delta t) = x_i(t) + (x_i(t) - x_j(t))v_x; \tag{11}$$

$$y_i(t + \Delta t) = y_i(t) + (y_i(t) - y_j(t))v_y; \tag{12}$$

For the sake of simplicity, no momentum calculations were taken into account in handling the collisions.
The above numerical method, allowed the sand particles to move towards the nodal positions, and accumulate at the nodal points, following the theoretical patterns expected.

4 The program

The program was written and run in MATLAB using the following code.

```
%Chladni Pattern Simulation For a Square Plate-- Project 3
%Introduction to Computer Smulations
%Nadine Soliman

%%% variables for equations %%%

% constants in equation
m=7;
n=9;
A =1;
B = 1;

%length of the square plate
l=1;
```

```
%velocity of propagation of the wave
v=2;

%wave numbers
kx = m*pi/l;
ky = n*pi/l;

%angular velocity
w = v*sqrt(kx^2+ky^2);

% initialize x,y,z grid
x = -l:0.01:l;
y = -l:0.01:l;
z = zeros(size(x,2), size(y, 2));

% number of particles on the plate
N = 2500;

%%variables for plate with sand particles
%length of plate with sound x100 scale
L = 100;
kxL = m*pi/L;
kyL = n*pi/L;

%diameter of particles
diam = 0.75;

%drawing the gradient of the function
[uu,vv] = meshgrid(0:2:L,0:2:L);

Z = (A*sin(uu.*kxL).*sin(vv.*kyL)+B*sin(vv.*kxL).*sin(uu.*kyL));
[DX,DY] = gradient(Z,2,2);

% x,y coordinates of the sand particles
xx=zeros(1,N);
yy=zeros(1,N);

%space that could be occupied
space =L;

%initialize the positions of particles
for i=1:N
    xx(i) = rand*space;
    yy(i) = rand*space;
end
```

```
%axis limits setup
axisLimits2D = [-1 1 -1 1];
axisLimits3D = [0 1 0 1 -abs((A+B)) abs((A+B))];

for t = 0:0.1:10000
    %make the plate vibrate
    for i =1:size(x,2)
        for j=1:size(y,2)
            z(i,j) = z(i,j)+(A*sin(kx*x(i))*sin(ky*y(j)) + B*sin(kx*y(j))*sin(ky
            z(i,j) = z(i,j)/(abs(A)+abs(B));
        end
    end

    %move the sand particles according to the gradient
    for i=1:N
        dx = kxL*A*cos(xx(i)*kxL)*sin(yy(i)*kyL)+B*kyL*cos(xx(i)*kyL)*sin(yy(i)*
        dy = kyL*A*sin(xx(i)*kxL)*cos(yy(i)*kyL)+B*kxL*sin(xx(i)*kyL)*cos(yy(i)*

        %move according to the negative of the gradient
        x_to_go = xx(i)- dx;
        y_to_go = yy(i)- dy;

        %check for collisions
        for j=1:N
            if i~=j
                %if they hit make them bounce off eachother
                if ((x_to_go -xx(j))^2 + (y_to_go - yy(j))^2)<(diam^2)
                    x_to_go =xx(i)+ (x_to_go -xx(j))*diam*0.25;
                    y_to_go =yy(i)+(y_to_go - yy(j))*diam*0.25;
                    break
                end
            end
        end
        %move the particles
        xx(i) = x_to_go;
        yy(i) = y_to_go;
    end

    %plotting
    %vibrating square palte
    subplot(2,2,1);
    s = surf(x,y,z);
    axis(axisLimits3D);

    %contour map
    subplot(2,2,2);
```

```
    contourf(x,y,z);
    axis(axisLimits2D);

    %predicted solution
    toPlot = @(x,y) (A*sin(x.*kx).*sin(y.*ky) +B*sin(y.*kx).*sin(x.*ky));
    subplot(2,2,3);fimplicit(toPlot, [0 1 0 1]);

    %sand particles moving according to the vibrations
    subplot(2,2,4);
    %the flow field part
    quiver(uu,vv,DX.*sin(w*t),DY.*sin(w*t));
    hold on
    %the sand particles part
    scatter(xx, yy, 10);
    axis([0 L 0 L]);

    drawnow
    hold off
    drawnow
    axis manual
    axis equal
end
```

5 Results and Discussion

Below are some of the resulting patterns that were obtained when running the simulation.
Figures 1 and 2 below correspond to the mode of oscillation where $m = 9, n = 7, a = 1, B = 1$. As shown in the figures, the sand particles indeed follow the flow field and form the expected theoretical patterns.

Figure 1: Beginning of the Run

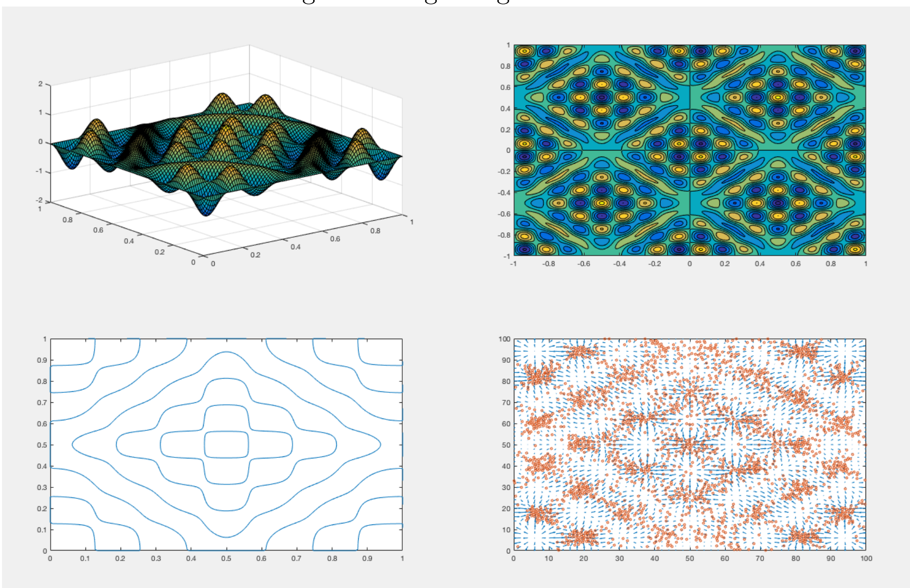
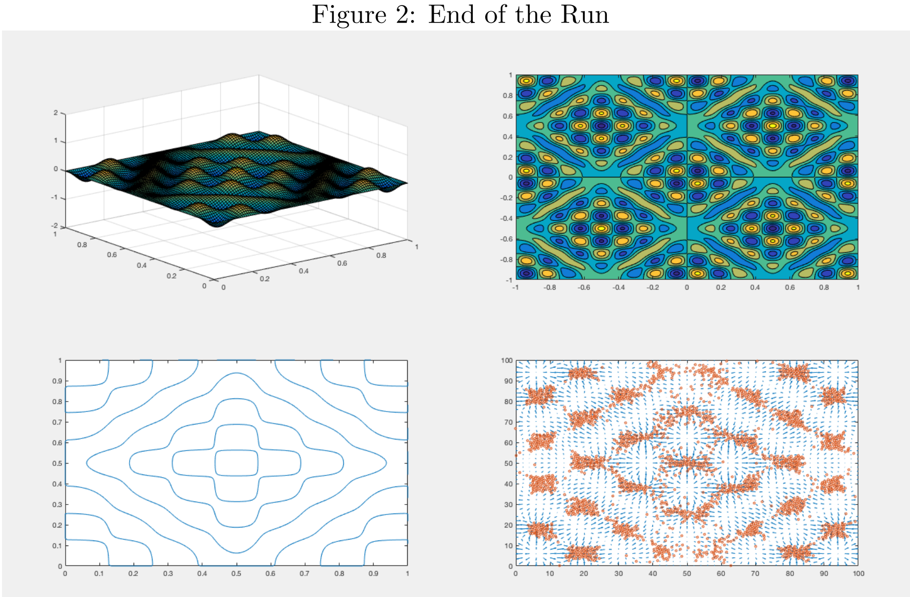


Figure 2: End of the Run



Figures 3, 4 and 5 below correspond to the mode of oscillation where $m =$

$9, n = 8, a = 1, B = 1$. Similarly, the sand particles also seem to follow the predicted pattern on the left, however, we note that some particles tend to collect at certain nodal points. This is due to the fact that the flow field is pushing the sand particles towards a singular point, that is the center of each node. And as the two sand particles can not be at the same position, they cluster together around the nodal centers. While implementing collisions in between the sand particles prevents them from collapsing to a single point, it does not prevent them from forming clusters instead.

Figure 3: Beginning of the Run

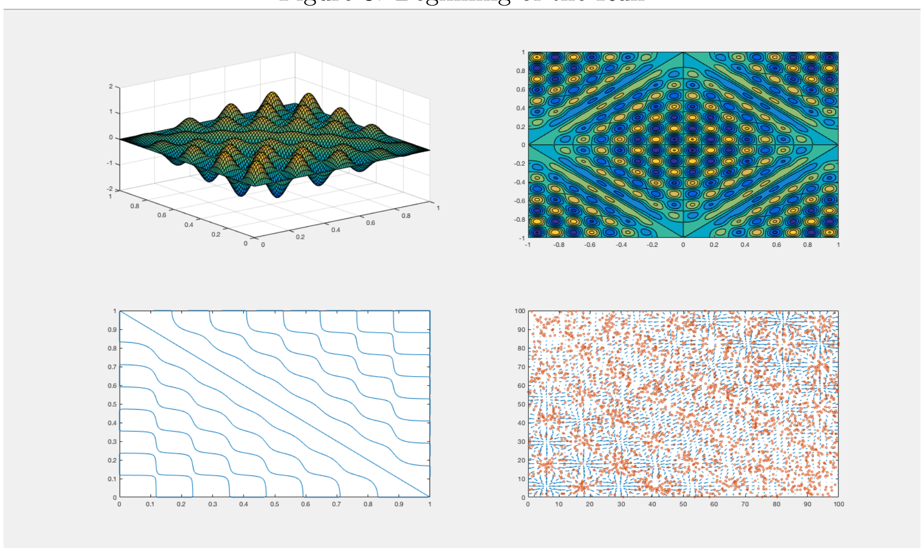


Figure 4: During the Run

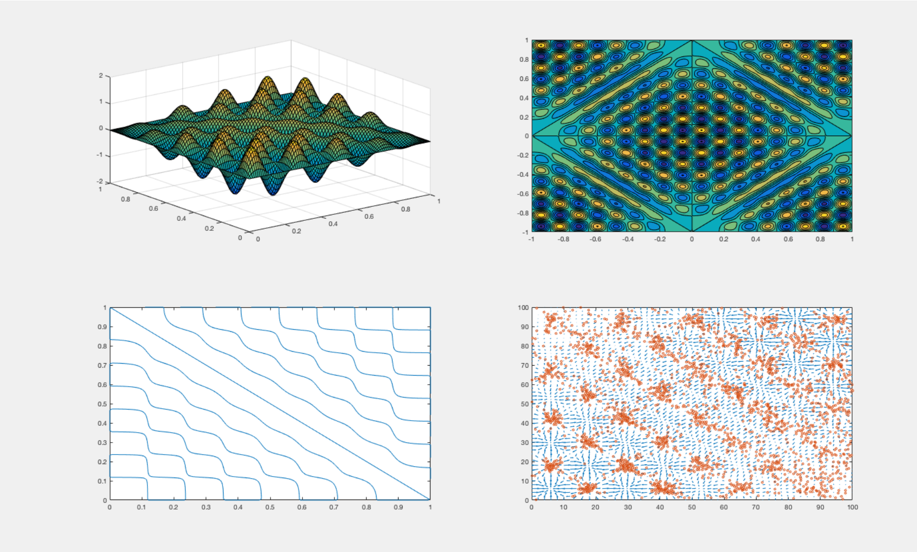
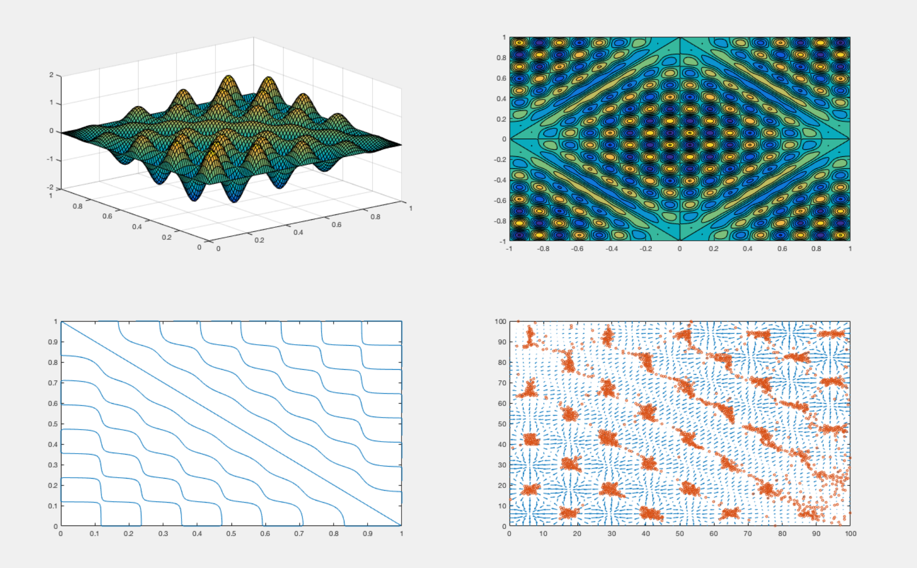


Figure 5: End of the Run



Figures 6, 7 and 8 below correspond to the mode of oscillation where $m = 3, n = 4, a = 1, B = 1$. As shown in the figures below, the pattern formed by the sand particles in this mode is less clear than the previous two patterns. After running the simulation with a wide range of values for m and n, we observed, that for smaller values of m and n, the resultant patterns formed by the sand particles tends to be less accurate. This is a result of the use of a flow field to drive the motion of the sand particles. When there are more nodes on the surface, the simulation yields more precise results, as there is less particles clustering together. However, with a fewer number of nodes, the clustering of the particles around the nodes increases and makes the patterns less accurate.

Figure 6: Beginning of the Run

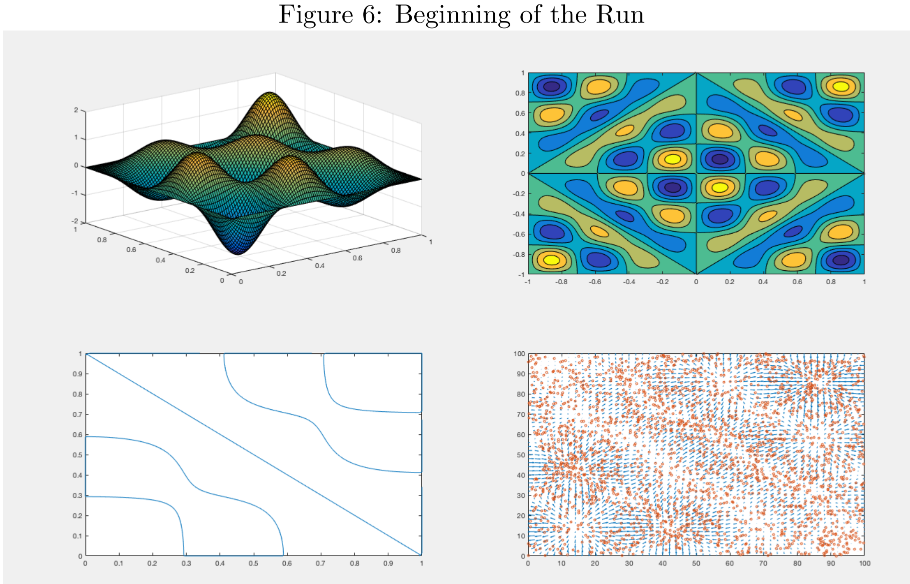


Figure 7: During the Run

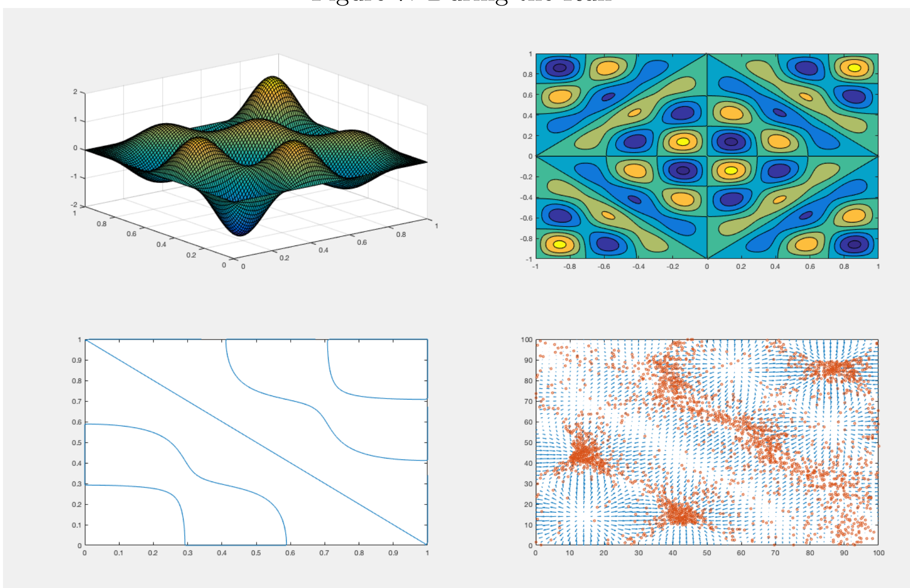


Figure 8: End of the Run