Journal of Applied Engineering Mathematics

Volume 12, December 2025

HARMONICS WITHIN A TONGUE DRUM

Damon Henderson and Teagan Matthews

Mechanical Engineering Department
Brigham Young University
Provo, Utah 84602

ABSTRACT

The sound an instrument makes is dependent upon the frequencies it vibrates at. To find these frequencies for a steel tongue drum, the tongue drum is modeled with the 2-D wave equation. The Neuman-Neuman and Neuman-Dirichlet boundary conditions are used, and a solution is found using separation of variables. A visualization of the solution is used to analyze the characteristics of the solution. The frequencies of vibrations are calculated using the fast Fourier transform and analyzed. The frequency spectrum is too noisy for there to be clear harmonics, suggesting that another approach is needed to accurately model the tongue drum.

NOMENCLATURE

$u(x, y, t)$ = position function for the surface of the tongue drum
$X(x)$ = x-coordinate dependent component of the position function of the tongue drum
$Y(y)$ = y-coordinate dependent component of the position function of the tongue drum
$T(t)$ = time dependent component of the position function of the tongue drum
α = wave speed
t = time
x = position along the length of the tongue drum
y = position along the width of the tongue drum
x_0 = x-coordinate where tongue drum is struck
y_0 = y-coordinate where tongue drum is struck
β = eigenvalue for T(t)
λ = eigenvalue for X(x)
ν = eigenvalue for Y(y)
L = length of the tongue drum
M = width of the tongue drum

INTRODUCTION

Instruments create musical notes by vibrating along the primary frequency associated with the note (e.g. A440's primary frequency is 440 Hz). However, the sound which the instrument makes is influenced by the other frequencies produced by the instrument. The difference in these frequencies causes instruments to have different sounds even when playing the same notes. The sound of many instruments can accurately be modeled using the 1D wave equation (e.g. guitar, trumpet, organ, and clarinet), which makes analysis of these instruments' sounds relatively easy.

In contrast, a tongue drum (Fig 1) is an instrument made of metal. A note is played by striking one of various metal "tongues" with a mallet. Because the tongue drum's width and length are close, the tongue drum cannot be accurately modeled using the 1D wave equation.

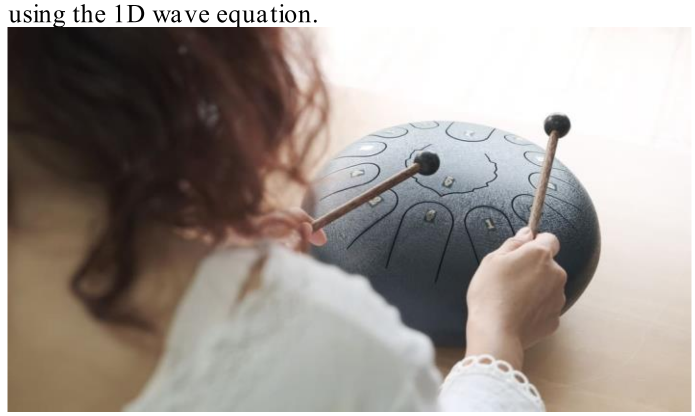


Figure 1. Woman playing a steel tongue drum. Photo credit: https://healing-sounds.com/

Our purpose is to model a metal tongue once it is struck and to analyze the characteristics of the vibration to better understand the sound of the tongue drum.

METHOD

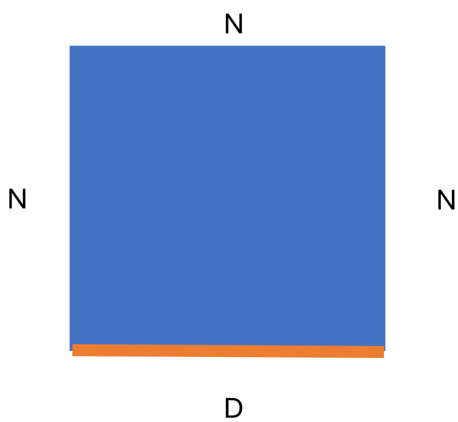
We use the 2-dimensional wave equation to model the tongue, where u is the transverse displacement of a point on the tongue from its equilibrium position and α is the wave speed.

$$\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}=\alpha^2\frac{\partial^2 u}{\partial t^2}$$

It should be noted that our model does not account for the stiffness of the material; a more accurate model would use a 4th-order equation.
We simplify the geometry of the tongue to a rectangle of length L and width M. We model the side connected to the rest of the drum as a fixed boundary (Dirichlet boundary condition). The three unconnected sides we model as free boundaries (Neumann boundary condition).

$$[u]_{x=0}=0, \left[\frac{\partial u}{\partial x}\right]_{x=L}=0$$

$$\left[\frac{\partial u}{\partial y}\right]_{y=0}=0, \left[\frac{\partial u}{\partial y}\right]_{y=M}=0$$



Our solution aims to analyze the effect of striking the tongue at t = 0. We approximate this event as a Dirac delta function applied at the location of the strike:

$$u(x,y,0)=\delta(x-x_0,y-y_0), \quad \frac{\partial u(x,y,0)}{\partial t}=0$$

To solve this partial differential equation, we apply separation of variables. We assume u to be a product of three independent functions

$$u=X(x)Y(y)T(t)$$

Applying the governing equation to this function yields

$$X''YT+XY''T=\alpha^2T''XY$$

Rearranging so that terms from similar functions are grouped gives us

$$\frac{X''}{X}+\frac{Y''}{Y}=\alpha^2\frac{T''}{T}$$

For this equation to be true, it must equal a constant, which we assume to be negative

$$\frac{X''}{X}+\frac{Y''}{Y}=\alpha^2\frac{T''}{T}=-\beta^2$$

We proceed to find appropriate eigenfunctions for X, assuming it equals a negative constant

$$\frac{X''}{X}=-\lambda^2$$

Rearranging so that all terms are on the left side of the equation gives

$$X''+\lambda^2X=0$$

As $u(x,y,t)$ has Dirichlet-Neuman boundary conditions along the x boundaries, it follows that X(x) has the same boundary conditions. Thus, the eigenfunctions for X(x) are

$$X_n=\sin[\lambda_n x],\ \lambda_n=\frac{(2n+1)\pi}{2L},\ n=0,1,2,\ldots$$

We do the same for Y

$$\frac{Y''}{Y}=-\nu^2$$

Rearranging so that all terms are on the left side of the equation gives

$$Y''+\nu^2Y=0$$

As $u(x,y,t)$ has Neuman-Neuman boundary conditions along the y boundaries, it follows that Y(y) has the same boundary conditions. Thus, the eigenfunctions for Y(y) are

$$Y_k=\cos(\nu_k y),\ \nu_k=\frac{k\pi}{M},\ k=0,1,2,\ldots$$

We find appropriate eigenfunctions for T

$$\alpha^2\frac{T''}{T}=-\beta^2$$

Rearranging so that all terms are on the left side of the equation gives

$$T''+\frac{\beta^2}{\alpha^2}T=0$$

Given that we assume that β^2 is negative, the general form of the solution for T(t) is

$$T=c_1\sin\left(\frac{\beta}{\alpha}t\right)+c_2\cos\left(\frac{\beta}{\alpha}t\right)$$

With the first derivative of T(t) being

$$T'=c_1\frac{\beta}{\alpha}\cos\left(\frac{\beta}{\alpha}t\right)-c_2\frac{\beta}{\alpha}\sin\left(\frac{\beta}{\alpha}t\right)$$

We find the particular solution by first applying the initial velocity

$$T'(0) = 0 = c_1\frac{\beta}{\alpha}\cos\left(\frac{\beta}{\alpha}0\right) - c_2\frac{\beta}{\alpha}\sin\left(\frac{\beta}{\alpha}0\right)$$

Applying trigonometric identities turns the equation into

$$0 = c_1\frac{\beta}{\alpha}\cdot 1 - 0$$

This results in

$$c_1 = 0$$

And our particular solution is of the form

$$T = c_2\cos\left(\frac{\beta}{\alpha}t\right)$$

Next, we solve for β. Recall that

$$\frac{X''}{X} + \frac{Y''}{Y} = -\beta^2$$

As we know that

$$\frac{X''}{X} = -\lambda^2$$

and

$$\frac{Y''}{Y} = -\nu^2$$

we find that

$$-\beta^2 = -\lambda^2 - \nu^2$$

Inserting our values for λ and ν and solving for β gives

$$\therefore \beta = \pm\sqrt{\lambda^2 + \nu^2}$$

Because cosine is symmetric about 0, the $\pm$ may be ignored. This gives our particular solution for T(t) as

$$T_{n,k} = a_{n,k}\cos\left(\frac{\sqrt{\lambda_n^2 + \nu_k^2}}{\alpha}t\right)$$

Hence, our solution for u is of the form

$$u(x,y,t) = \sum_{n=0}^{\infty}\sum_{k=0}^{\infty} a_{n,k}X_nY_kT_{n,k}$$

To find the constants $a_{n,k}$ we apply the initial condition

$$\delta(x - x_0, y - y_0) = \sum_{n=0}^{\infty}\sum_{k=0}^{\infty} a_{n,k}X_nY_kT_{n,k}$$

$$T_{n,k}(0) = \cos\left(\frac{\sqrt{\lambda_k^2 + \nu_n^2}}{\alpha}\cdot 0\right) = 1$$

$$\therefore \delta(x - x_0, y - y_0) = \sum_{n=0}^{\infty}\sum_{k=0}^{\infty} a_{n,k}X_nY_k$$

We now multiply both sides by X_pY_q, where p and q are arbitrary constants.

$$\therefore \delta(x - x_0, y - y_0)X_pY_q = X_pY_q\cdot\sum_{n=0}^{\infty}\sum_{k=0}^{\infty} a_{n,k}X_nY_k$$

Because of the orthogonality of eigenfunctions, the only term of the summation that will not equal zero is the term where $p = n$ and $q = k$, hence

$$\delta(x - x_0, y - y_0)X_nY_k = a_{n,k}X_n^2Y_k^2$$

Integrating twice,

$$\int_0^L\int_0^M \delta(x - x_0, y - y_0)X_nY_k dxdy = a_{n,k}|X_n|\cdot|Y_k|$$

$$X_n(x_0)Y_k(y_0) = a_{n,k}|X_n|\cdot|Y_k|$$

Which yields

$$a_{n,k} = \frac{2}{LM}\sin\left(\frac{(2n+1)\pi}{2L}x_0\right),\ k = 0$$

$$a_{n,k} = \frac{4}{LM}\sin\left(\frac{(2n+1)\pi}{2L}x_0\right)\cos\left(\frac{k\pi}{M}y_0\right),\ k = 1,2,\ldots$$

This gives us the complete solution for the motion of the tongue drum

$$u(x,y,t) = \sum_{n=0}^{\infty}\Bigg(\frac{2}{LM}\sin\left(\frac{(2n+1)\pi}{2L}x_0\right)\sin\left(\frac{(2n+1)\pi}{2L}x\right) * \cos\left(\frac{(2n+1)\pi}{2L\alpha}t\right) + \sum_{k=1}^{\infty}\frac{4}{LM}\sin\left(\frac{(2n+1)\pi}{2L}x_0\right)\cos\left(\frac{k\pi}{M}y_0\right)\sin\left(\frac{(2n+1)\pi}{2L}x\right) * \cos\left(\frac{k\pi}{M}\right)\cos\left(\frac{\sqrt{\left(\frac{(2n+1)\pi}{2L}\right)^2 + \left(\frac{k\pi}{M}\right)^2}}{\alpha}t\right)\Bigg)$$

To analyze the sound of the tongue, we model the solution, $\boldsymbol{u(x,y,t)}$, in MATLAB for a finite amount of n and k. To analyze the dominant frequencies of vibration, we take the numerical fast Fourier transform of the model in time. For this simulation we set L = 1, M = 1, $x_0 = \frac{L}{2}$, $y_0 = \frac{M}{2}$, and α = 1. The model was simulated for 20 seconds, with a time step of 0.00125 s. This gives a Nyquist frequency of 400 Hz and a frequency resolution of 0.033 Hz. As we are interested in the sound produced by the entire tongue, we take the fast Fourier transform of each point on the drum and sum the result for a global frequency spectrum (see Appendix for MATLAB code modeling the solution and taking the fast Fourier transform).

CONCLUSIONS

The results (Fig 2) show that there are many spikes along the frequency spectrum from around 10 Hz until around 275 Hz with no discernible pattern in harmonics development, that is the spikes are not integer multiples of the dominant frequency).

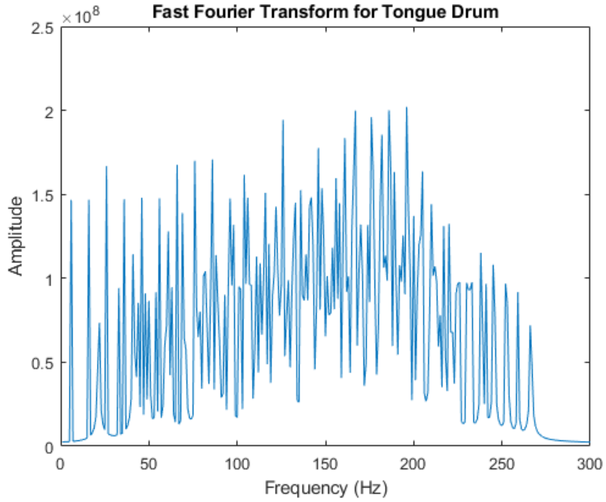


Fig 2: Frequency spectrum of a Tongue drum produced by a fast Fourier transform.

The large number of spikes indicates that there are many frequencies interacting in the tongue, due to waves reflecting at the boundaries of the tongue. The large number of frequencies suggest that there are many different notes interacting within a tongue which is not true. In addition, there is a positive drift in the baseline of the fast Fourier transform. This suggests that there are non-physical interactions impacting the frequency spectrum. Both of these results indicate that neglecting the stiffness of the tongue drum is an invalid assumption to make. Thus, to truly understand the unique sound of the tongue drum, a different analysis is needed.

APPENDIX

```
clc, clear, close all

% If the solution was not previously calculated and save calculate it
% This is included so that the solution can be saved, speeding up analysis
% of the fourier transform
savefile = true;
if not(isfile("2D_wave_equation_solution.mat"))

% Set up parameters
L = 1; % End of x domain and length of tongue drum
M = 1; % End of y domain and width of tongue drum
beta = L/2; % x coordinate of hammer strike
omega = M/2; % y coordinate of hammer strike
alpha = 1; % wave speed

% Set up domains for the solution
x = 0:0.01:L;
y = 0:0.01:M;
[X, Y] = meshgrid(x,y);
dt = 1/800;
t = 0:dt:20;
u = zeros(size(X,1), size(X,2), size(t,2));
% Calculate the solution for 20 n and 20 m terms
for k = 1:size(t,2)
  %u = zeros(size(X));
  for i = 1:20
    for j = 1:20
      n = i - 1;
      m = j - 1;
      a_n_m = 4 / (M*L) * cos(m*pi/M*omega) * sin((2*n+1) / (2*L) * pi * beta);
      x_term = sin((2*n+1) / (2*L) * pi * X);
      y_term = cos(m*pi*Y / M);
      t_term = cos(1/alpha * sqrt(pi^2 / (4*L^2)* (2*n+1)^2 + m^2*pi^2/M^2)*t(k));
      u(:, :, k) = u(:, :, k) + a_n_m .* x_term .* y_term .* t_term;
    end
  end
end

if savefile
  save 2D_wave_equation_solution.mat u x y t dt Y X;
end
% If the solution was preivously calculated and saved, load it
else
  load('2D_wave_equation_solution.mat');

end

%%5 ---- Code to create animation of solution and save it as a video file
saveVideo = false;
videoName = 'tongue_drum_vibration3D.mp4';

if saveVideo
  v = VideoWriter(videoName, 'MPEG-4');
  v.FrameRate = 30;
  open(v);
end

playVideo = false;
if playVideo
  figure;
  for k = 1:size(t,2)
    surf(X,Y,u(:,:,k));
    view(3);
    title(sprintf('Vibration at t = %0.4f seconds', t(k)));
    xlabel('x');
    ylabel('y');
    zlabel('u');
    zlim([-60, 60]);
    drawnow;

    if(saveVideo)
      frame = getframe(gcf);
      writeVideo(v, frame);
    end
```

```
            pause(0.01);
        end
    end

    if saveVideo
        close(v);
        disp(['Video  saved as: ', videoName]);
    end

    %%% ---- Code to perform FFT on solution and analyze it
    % We want only temporal frequencies so take fft only over
time at each x,y
    % piar
    f = zeros(size(u));
    for i = 1:length(x)
        for j = 1:length(y)
            f(i,j,:) = fft(u(i,j,:)); % take fft at each coordinate on the
string
        end
    end
    % Sum the FFT frequencies from each point to get the global
frequency
    % spectrum
    freq = sum( sum( abs( f(:,:,1:floor(length(t)/2)) ), 1), 2);
    % Indices for plotting
    ind=1:length(freq);
    figure(2);
    freq = squeeze(freq);
    % Plot up to 300 Hz, as there are no peaks in the frequency
spectrum past
    % that point
    plot(ind(1:300), freq(1:300));
    title("Fast Fourier Transform for Tongue Drum");
    xlabel("Frequency (Hz)");
    ylabel("Amplitude");
```

