

A
THESIS
SUBMITTED AS PART OF THE APPLIED PHYSICS PROGRAM
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
BACHELOR OF SCIENCE IN PHYSICS

Modes of the Upper and Lower Bout of the Violin Top Plate

Julie DiNitto

DEPARTMENT OF PHYSICS
INDIANA UNIVERSITY

Date

Supervisor:

Signature
Rick Van Kooten

Co-signer:

Signature
Matthew Shephard

Abstract:

In determining the modes of the upper and lower bout of the violin, we were able to obtain a physical representation of the first ten modes of a circle-like representation and the first six modes of a stadium-like representation of the bouts of a violin¹. We concluded that the resonant frequencies were high because we were dealing with a plate of spruce wood that has a high speed of sound. The high speed of sound allows the plate to be set into motion quickly, which is ideal for a violin to sound. We then compare this frequency to the resonant frequency of the box of the violin.

Introduction:

The first violin known to date is the 1564 Andrea Amati violin named Charles IX. Andrea Amati was one of many famous Cremonese makers from Cremona, Italy. Other well known makers are Antonio Stradivarius and Guarneri. It is believed that the original Cremonese makers succeeded in their field of work because they understood how the violin functioned as a whole. Many understood mathematically and scientifically what was occurring and were able to adapt their violins to accommodate for experimentation. Through experimentation, the Cremonese were still able to produce beautiful sounding instruments. Unfortunately the true understanding of violins was lost because mathematics was considered witchcraft and was not passed on to the apprentices of the Cremonese makers. The apprentices rather copied their instructor's previous instruments. Today, we are still copying the Cremonese makers, but also have become very curious as to how they function.

¹ The bout of a violin is roughly the oval shaped part of the top and bottom of the violin plate.

Looking at the design reveals a very complex structure that we today cannot predict the outcome before the violin is finished. Within this complex structure many mathematical models and bases have been uncovered; such as repeating numbers, aveca pieces, catenaries, golden ratio, and many more. For our purpose there will be a brief discussion on repeating numbers and the construction of the aveca pieces which is the formation of pentagons. No one will ever know whether or not these mathematical concepts are how the violin was first designed, but as a guess, it still holds today.

One of these mathematical concepts in the layout of the violin top is repeating numbers. Throughout the violin we see repeating numbers. A repeating number is when you find a number two or more times in a given region. For example, the inside of the violin is 321 mm from end block to upper block and the string length is also 321 mm. This style is continuous throughout the whole violin. Where there is one measurement, there is the same one existing somewhere else. The famous Cremonese maker Antonio Stradivarius used repeating numbers everywhere on his instruments. One of Stradivarius' ways of determining repeating numbers was by using the golden ratio. The golden ratio is $(1 \pm \sqrt{5})/2$ which equals 1.618 and 0.618. This implies a ratio of 1 to 0.618. He began with 1 and repeatedly multiplied it by 1.618 until he arrived at a number that best suited a measurement on the violin. Another way of finding repeating numbers is taking an existing number and continually dividing the number by 2. One of the most popular numbers used is 40.125 mm. Here is how this number came about. The number 321 mm when halved returns the length of 160.5 mm which is the distance from the end block to where the conical hole is on the inside of the violin. The next important number is 40.125, which is the width between the upper eyes of the f holes.

In the design of the violin, the pentagram allows you to physically draw up the shape of the violin and determine where the nodes are throughout the instrument. The pentagram is believed to be how the violin was originally designed. Courtesy of Thomas G. Sparks an excellent representation is shown in Fig. 1.1²

² To see the lower bout separately see Appendix D.

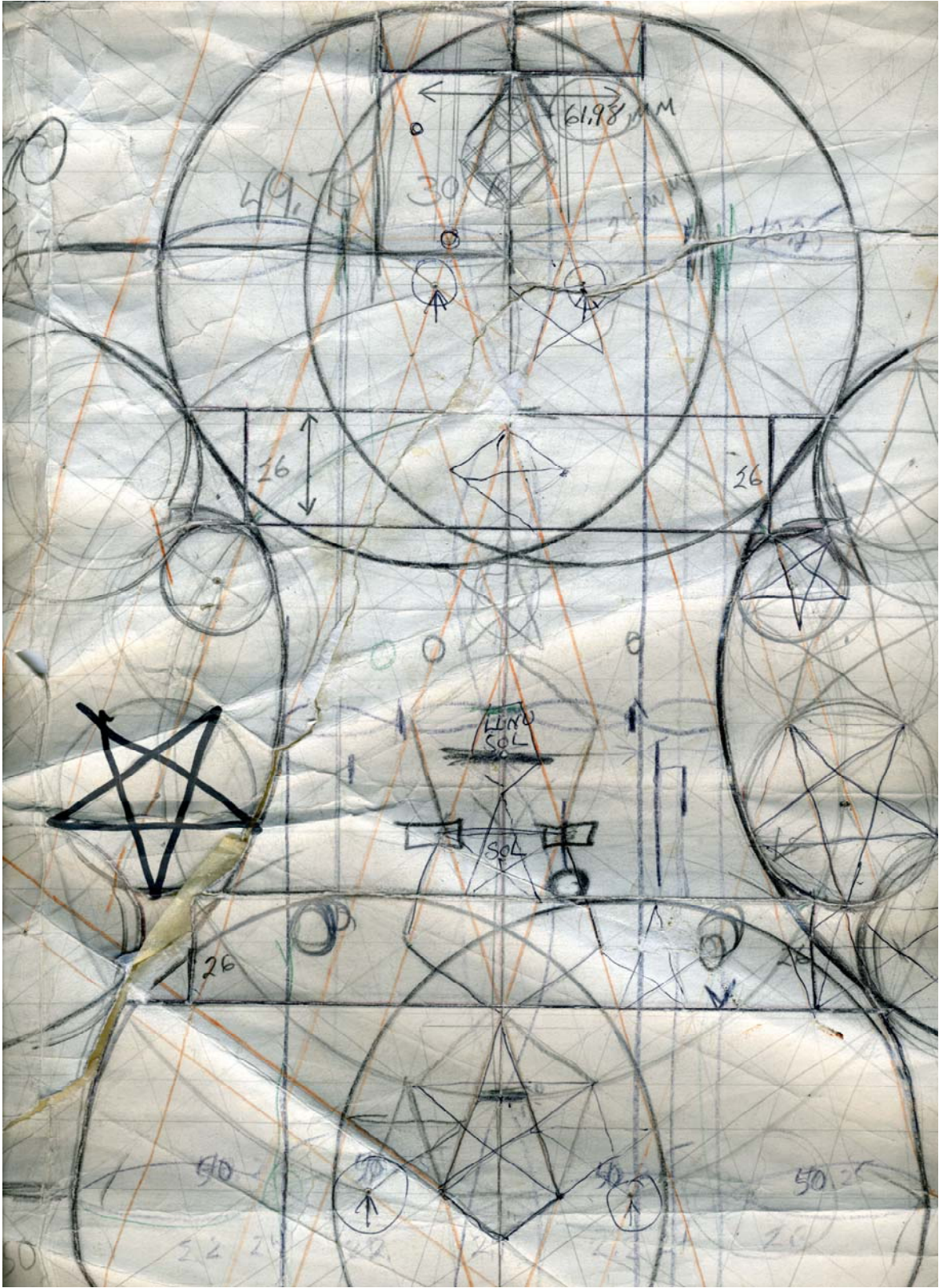


Figure 1.1 (not scaled to actual size)

In constructing the pentagon, you begin with two circles of diameter 321 mm (the length of the violin from end block to upper block) with distance 160.5 mm center to center of the circles. Next a pentagon is constructed on the inside of one of the circles. Within the pentagon a five point star can be formed and within that star is another pentagon in which another star is able to be formed. This process can continue for an infinite amount of times. If we extend the lines that create each star all the way throughout the circle, then a grid is formed within the circle. Within this new grid, one can see there are ten lines crossing at a single point. These points are nodal points. By taking these nodal points, we can determine the lower and upper bout using circles. We know that the largest part of the upper bout is 160.5 mm; therefore, by finding nodal areas on the grid, we can construct two circles a distance d apart between the center of the circles, totaling 160.5 mm. The same goes for the lower bout, which has a measurement of 200 mm between the largest part of the lower bout. This is as far as we need to proceed in the construction of the violin through the pentagram since we are going to now examine the modes of the violin within these two regions of the violin; the lower bout and upper bout. Before we get into this, an introduction into what is going on within the regions is necessary.

Data/Analysis:

When we pluck a string on a violin, we can see that we have set the string into motion; this is also referred to as a pizzicato. Consider the initial state where the string is resting at zero. The motion can be considered a fixed ended wave, because the bridge and the string nut are acting as nodal points.

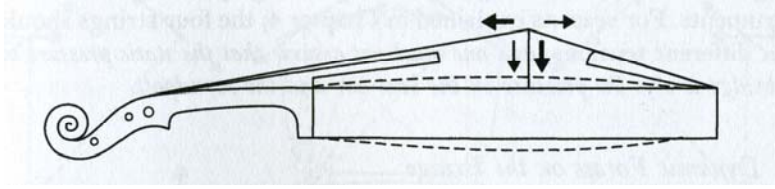


Figure 1.2 (Beasment, 14)

If we pluck the string we can see that the two forces from the fixed ends are acting on the string in opposing directions. See Fig. 1.3. The two forces eventually head in the same direction (#4 of figure 1.3), but the shape of the string is half positive and half negative according to how we defined the initial state. This process continues creating the complete opposite of when the string was released. At this point, we can consider the point at which the force directions cross; see # 7 in Fig. 1.3.

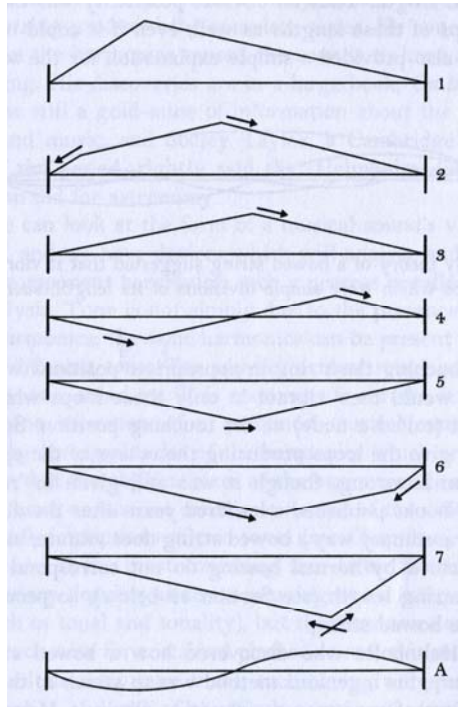


Figure 1.3 (Beasment, 12)

This process continues until the string is no longer in motion. Here is a picture of the one

dimensional wave equation and what is going on frame by frame³.

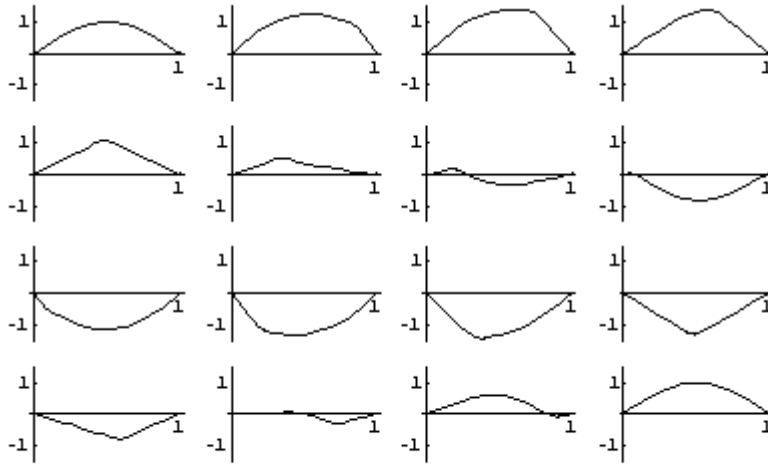


figure 1.4
one-dimensional wave as time elapses.

Let's examine the two-dimensional wave equation of a circle, which will be applied to the violin top later. The two-dimensional wave equation is,

$$\begin{aligned}
 c^2(u_{rr} + 1/r u_r) &= u_{tt}, \quad 0 < r < \rho, \quad t > 0 \\
 u(\rho, t) &= 0, \quad |u(0, t)| < \infty, \quad t > 0 \\
 u(r, 0) &= f(r), \quad u_t(r, 0) = g(r), \quad 0 < r < \rho
 \end{aligned}
 \tag{2}$$

([1], 542)

These solutions of the wave equation for the circle are called Bessel functions. Bessel functions can be used to define a two-dimensional function that is fixed to zero on a closed boundary. The lowest order two-dimensional Bessel function can be used to model a drumhead with radius R . The boundary conditions are $f(R) = 0$. Let's consider the Bessel function as a drumhead. Depending on where the drumhead is struck, different motions or modes within the circle are excited. If the drum is struck directly in the center, one can see that a cosine-like wave is formed. Likewise, if the drumhead is

³ Mathematica was used to produce this image. See appendix A for the Mathematica code leading up to this figure

struck at $R/2$ then a sine-like wave is formed; see Fig. 2.1.

Each order Bessel function has an eigenvalue that corresponds to different situations of how the membrane oscillates within the circle. The next step is to determine these eigenvalues. With the help of Mathematica, the eigenvalues are α and representing the mode number is n .

```

a_n := a_n = BesselJZeros 0, c;
c=340.29;
rho=1;
f[r_] = r(r-1);
g[r_] = Sin[Pi*r];

k_n := k_n = a_n rho;

a_n := a_n = Integrate[r f[r] BesselJ[0, k_n * r], {r, 0, rho}] / BesselJ[1, a_n];
b_n := b_n = Integrate[r g[r] BesselJ[0, k_n * r], {r, 0, rho}] / (k_n BesselJ[1, a_n]);

Table[{a_n, b_n}, {n, 1, 10}, TableForm

```

1	2.40483	2.6661
2	5.52008	6.11982
3	8.65373	9.59393
4	11.7915	13.0727
5	14.9309	16.5531
6	18.0711	20.0344
7	21.2116	23.5162
8	24.3525	26.9983
9	27.4935	30.4806
10	30.6346	33.963

Table 2.1

Each of these eigenvalues corresponds to a different mode of oscillation. Here are the first four modes⁴.



figure 2.1
Modes of oscillation of a single circle bessel function.

With these eigenvalues, we can obtain the frequencies of the top plate upper bout of the

⁴ see appendix B for Mathematica code

violin if it is modeled by a simple circle.

Consider the simplest case where the upper bout is a single circle in the design (we know that from the pentagon that two circles make up the majority of the upper bout, but we will get to that later). From the design we know that the radius of this circle must be 80 mm. Because we have determined the first 10 eigenvalues we can determine the frequency of the single circle plate of the upper bout where the fundamental is found by,

$$f_1 = 5.898 (h/r_0^2) \sqrt{(E/\rho)} \quad (2)$$

where h is the height of the plate, r_0 is the radius of the circular plate, $E = 14,000$ MPa is Young's modulus for spruce, and ρ is the density at 440 kg/m^3 . We can also obtain the harmonics of the plate by,

$$f_n = (m_n/m_1) * f_1 \quad (3)$$

where m_1 is the first eigenvalue and f_1 is the fundamental frequency. With these numbers, the frequency is as follows for the single circle of radius 80 mm.

r = 0.080 m

n	m	f (Hz)
1	2.4	20793.25
2	5.52	47824.48
3	8.65	32583.63
4	11.79	28341.33
5	14.93	26331.07
6	18.07	25166.38
7	21.21	24406.47
8	24.35	23871.56
9	27.49	23474.6
10	30.63	23168.33

Table 2.2

Now graphing the frequencies,

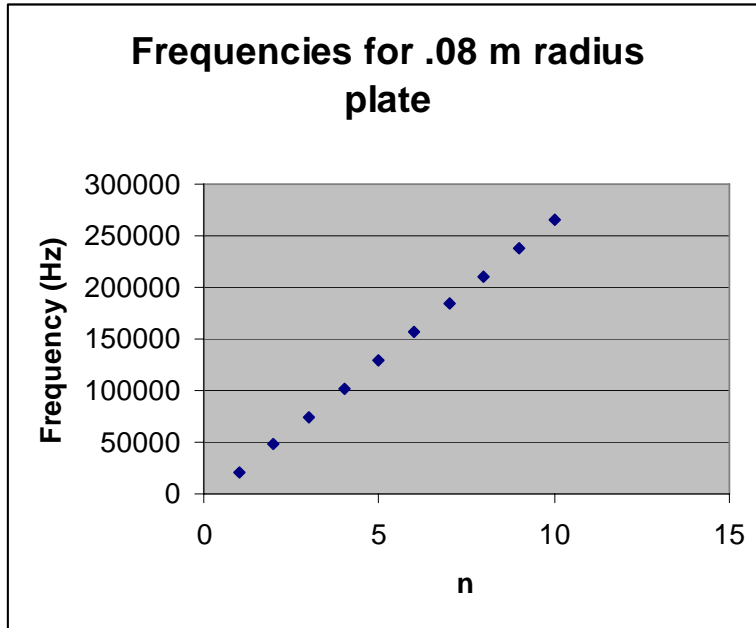


Figure 2.2

we can see that the frequencies are very close to sitting on a linear line, except if we take a closer look at the frequencies in a Bessel function, we can see that there are small non-linearities within the first few frequencies. Likewise, the frequencies for the lower bout of radius 100 mm is determined the same way.

n	m	f (Hz)
1	2.4	13307.68
2	5.52	30607.67
3	8.65	47963.11
4	11.79	65373.99
5	14.93	82784.88
6	18.07	100195.8
7	21.21	117606.6
8	24.35	135017.5
9	27.49	152428.4
10	30.63	169839.3

Table 2.3

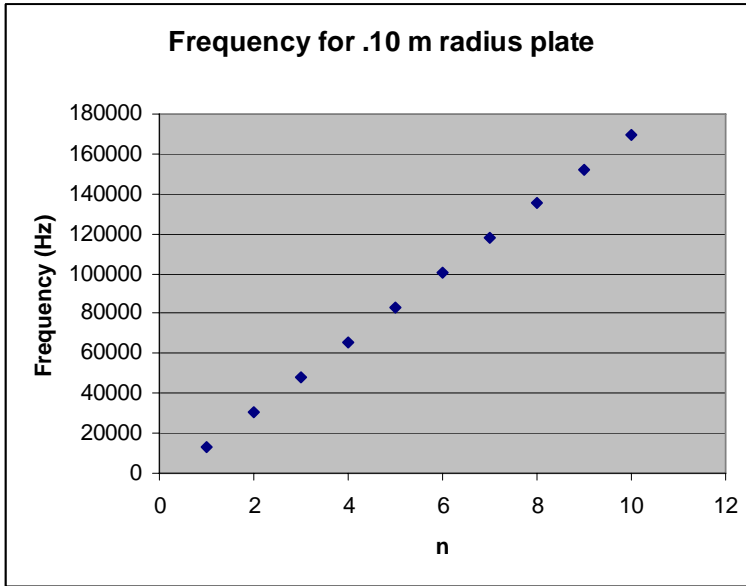


Figure 2.3

In order to better model the violin bouts, we need to move on to a stadium-like representation, where two circles of radius R are set next to each other a distance a apart.

See Figure 2.4.

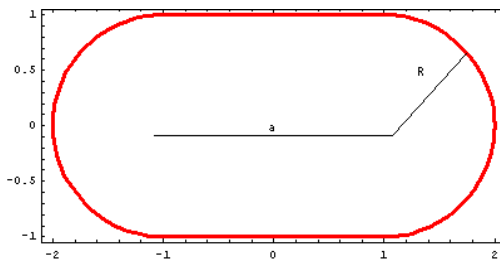


Figure 2.4

From the drawing of the pentagram, we get $R = 63$ mm and $a = 34$ mm for the upper bout and $R = 70$ mm and $a = 60$ mm for the lower bout. We can determine the frequency by setting the limit as $a \rightarrow 0$. This limit tells us that the value of R is just the radius of one of the circles. So for the upper bout, $R = 63$ mm and the lower bout $R = 70$ mm. But that is not all that has changed. The Eigen values of the solution to the wave equation are not

the same anymore, because we changed shape. Determining them is quite complex and the aid of Mathematica is needed⁵.

In the calculation of the eigenvalues for the stadium, k is the first scaled eigenvalue, $\pm o$ is the size of the eigenfrequency matrix, R is the radius of the circle, a is the distance between the two circles, and pp is the number of points for the numerical integration. Because the Eigen value is unitless, R and a are just ratios to each other.

```
fourierMatrix[k_, o_, {R_, a_}, pp_] :=
Module[{h[Delta] = N[ScriptL[{R, a}]/pp], rData, \[CurlyPhi]Data,
sData, l1, l2, l3},

With[{k = 1, l = 1, n = 1, R = 1, a = 0.859},
NIntegrate[Evaluate[
  Exp[-2 Pi I n s/ScriptL[{R, a}]] Exp[I l \[CurlyPhi][s, {R,
a}]] *
  BesselJ[l, k r[s, {R, a}]]],
{s, 0, Pi, Pi + 2, 2 Pi + 2, ScriptL[{R, a}]]] // Timing
{0.441659 Second, 2.73262 \[InvisibleSpace] -4.40427 \[ImaginaryI]}

det[k_?NumericQ, o_, {R_, a_}, pp_] :=
Det[fourierMatrix[k, o, {R, a}, pp]]

Timing[mat = Table[{k, det[k, 2, {1, 1}, 301]}, {k, 0.1, 6, 1/30}];]
{50.2989 Second, Null}

Show[Apply[Function[{reim, col},
(* real part in red; imaginary part in blue *)
ListPlot[{#[[1]], reim#[[2]]]& /@ mat,
PlotJoined -> True, PlotStyle -> col,
DisplayFunction -> Identity]],
{{Re, {Hue[0]}}, {Im, {Hue[0.8], Dashing[{0.01, 0.01]}}}], {1}],
DisplayFunction -> $DisplayFunction,
PlotRange -> {-1, 1}, Frame -> True, Axes -> False]
```

⁵ For full code from Mathematica see Appendix C

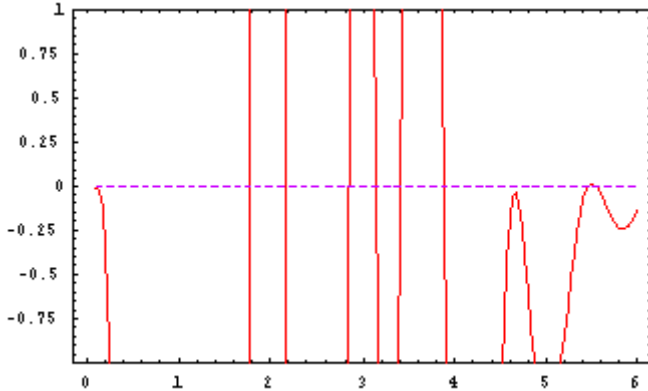


Figure 3.1

The crossings at zero are the eigenvalues that we are looking for. In order to find them we use the function `kRoots` from Mathematica.

```
startPairsN = {{1.7, 1.8}, {2.1, 2.2}, {2.8, 3.0}}
              {{1.7, 1.8}, {2.1, 2.2}, {2.8, 3.}}
kRoots = (k /. FindRoot[Re[det[k, 2, {1, 1}, 301]],
  Evaluate[{k, Sequence @@ #}],
  MaxIterations -> 30, AccuracyGoal -> 10])& /@ startPairsN
{1.77792, 2.1661, 2.84605}
```

Now that we have found the first 3 roots, we can graph them to see what the function looks like.

```
fourierMatrices =
  Transpose[fourierMatrix[#, 2, {1, 1}, 301]]& /@ kRoots;

nontrivialHomogeneousSolutions =
  NullSpace[#, Tolerance -> 0.1][[-1]]& /@ fourierMatrices;
\[CapitalPsi][{r_, \[CurlyPhi]_}, k_, v_] := v.Table[BesselJ[1, k r]
Exp[I 1 \[CurlyPhi]],
  {1, -(Length[v] - 1)/2, (Length[v] - 1)/2}]
eigenfunctionPicture[k_, v_, {pp\[CurlyPhi]_, ppr_}, col_, opts___] :=
Module[{points, polys, boundary},

boundary = {Thickness[0.01], GrayLevel[0],
  Line[Table[1.006 r[s, {1, 1}] *
    {Cos[\[CurlyPhi][s, {1, 1}]], Sin[\[CurlyPhi][s, {1,
1}]]}, 0},
    {s, 0, 2Pi + 4, (2Pi + 4)/501}]]];

points = Table[\[Rho] = N[\[Alpha] r[s, {1, 1}]]; \[Phi] =
N[\[CurlyPhi][s, {1, 1}]];
  {\[Rho] Cos[\[Phi]], \[Rho] Sin[\[Phi]],
Re[\[CapitalPsi][{\[Alpha] \[Rho], \[Phi]}, k, v]}],
  {s, 0, \[ScriptL][{1, 1}], \[ScriptL][{1, 1}]/pp\[CurlyPhi]},
  {\[Alpha], 0, 1, 1/ppr}];

polys = Table[Polygon[#[[i, j]], #[[i, j + 1]], #[[i + 1, j + 1]],
```

```

#[[i + 1, j]]&[points]], {i, pp\[CurlyPhi]}, {j,
ppr}];
(* show graphics *)
Show[Graphics3D[{boundary, EdgeForm[], col, polys}], opts,
PlotRange -> All, Boxed -> False, BoxRatios -> {4, 2, 1}]
Show[GraphicsArray[
Table[eigenfunctionPicture[kRoots[[k]],
nontrivialHomogeneousSolutions[[k]], {201, 101},
SurfaceColor[Hue[k/4], Hue[k/4 + 0.4], 2.8],
DisplayFunction -> Identity], {k, 3}]]]

```

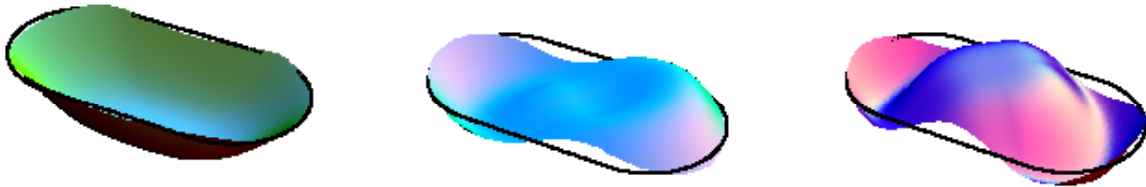


figure 3.2- First three modes of the oscillation of the upper bout.

We can use FindRoots again to determine the next three eigenvalues {3.13798, 3.40541, 3.88144} and their graphs are as follows.

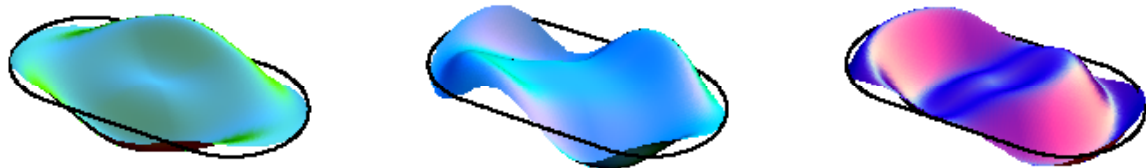


Figure 3.3- 4th-6th modes of the oscillation of the upper bout.

Now that we have determined the eigenvalues for the stadium-shape, we can calculate the frequency for a plate. For the upper bout with $R = 63$ mm, the frequencies are:

$r = 0.063$ m
 $d = 0.034$ m

N	k	f (Hz)
1	1.77792	33529.06
2	2.1661	40849.58
3	2.84605	53672.48
4	3.13798	59177.87

5	3.40541	64221.22
6	3.88144	73198.47

Table 3.1

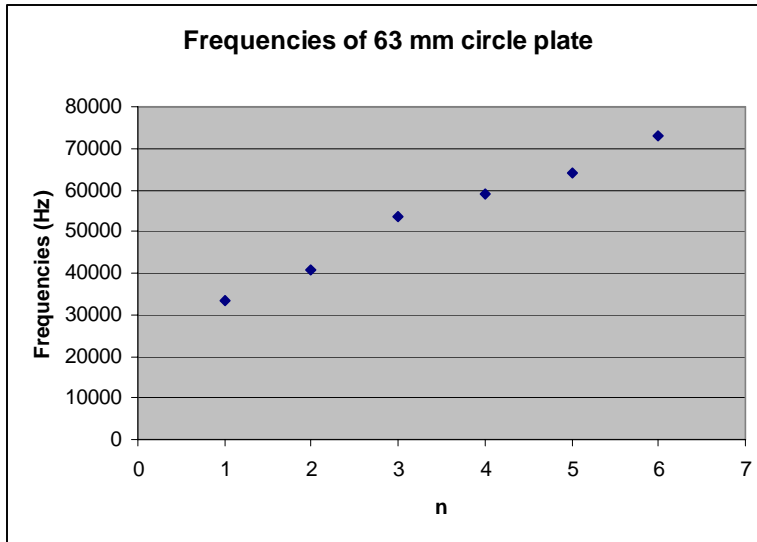


Figure 3.2

Likewise, the frequencies for the lower bout with $R = 70$ mm is as follows.

N	k	f (Hz)
1	1.77792	27158.54
2	2.1661	33088.16
3	2.84605	43474.71
4	3.13798	47934.07
5	3.40541	52019.19
6	3.88144	59290.76

Table 3.3

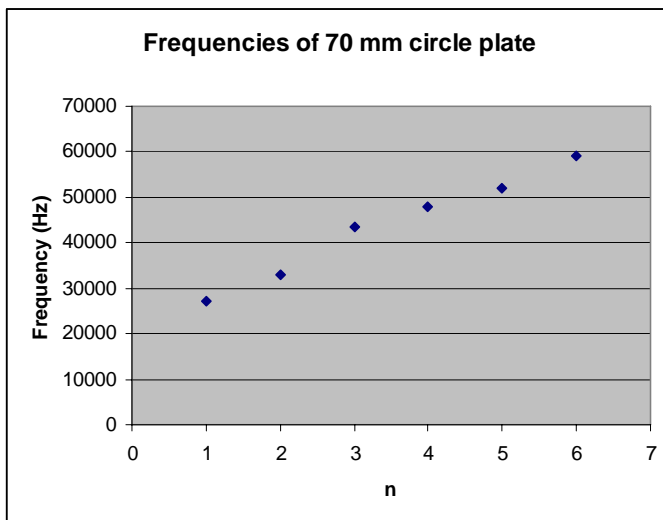


Figure 3.3

As you can see in all cases the frequencies are very high, and probably not in the range of human hearing. The reason for this is that each set of frequencies are for a small part of the instrument plate, not the actual air column in the violin. This shows that when the violin is played the violin is set into motion very quickly, because of the high speed of sound in the solid top plate. This is only a theoretical representation of the average speed of sound for spruce. Specific gravity plays a large role in changing these calculations. If the specific gravity lowers, then the speed of sound increases, because the actual weight of the wood has decreased. A low specific gravity is ideal in spruce for violins.

The frequencies for the air column of the box formed by the top and back plate of the violin are in the range of human hearing. If we call the direction pointing from the back plate to the top plate the z direction, we can calculate the frequencies in the air cavity in the z direction. First consider a box of height h . We know that,

$$h = \lambda/2 \quad (4)$$

where λ is the wavelength and

$$f = c/\lambda \quad (5)$$

where c is the speed of sound in air and f is the frequency. Therefore we can conclude that the frequency is,

$$f = c/(2h) \quad (6)$$

Now let us look at the instrument itself.

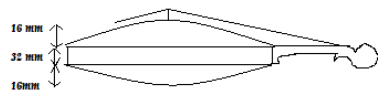


Figure:
The distance 16 mm corresponds to the arching on the top and back plate. The 32 mm represents the height of the rib structure.

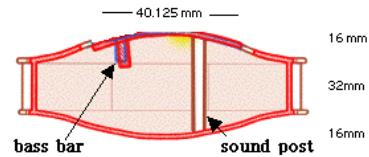


Figure
Cross section of the width of the violin.

Figure 4.1

We can see the shape of the top and bottom of the instrument look nothing like a box. If we look at the center of the violin top, we can determine an equation for the catenary.

The distance from f-hole to f-hole is 40.125 mm, with height 16 mm. Knowing the initial function form for a catenary as

$$y = (a/2) * (\text{Exp}[b*x/a] + \text{Exp}[-1*b*x/a]) - a \quad (7)$$

we can use Mathematica to obtain a and b respectively.

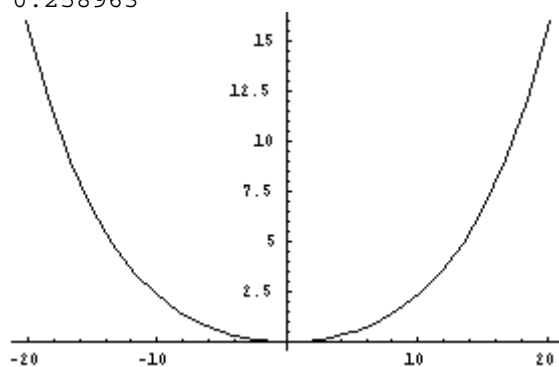
```
FindRoot[{(a/2)*(Exp[b*20.125/a]+Exp[-1*b*20.125/a])-a==16, (a/2)*(Exp[-1*b*20.125/a]+Exp[b*20.125/a])-a==16}, {a, 1}, {b, 1}]
```

```
{a->1.72558, b->0.258963}
```

```
FindRoot[{{1/2 a (e^(b 20.125/a) + e^(-b 20.125/a)) - a == 16,
1/2 a (e^(-b 20.125/a) + e^(b 20.125/a)) - a == 16}, {a, , b}]
```

```
a=1.72558
b=0.258963
```

```
Plot[(a/2)*(Exp[b*x/a]+Exp[-1*b*x/a])-a, {x, -20.125, 20.125}]
1.72558
0.258963
```



□Graphics□

We now have a physical representation of what the catenary of the violin top looks like. Now let's determine frequencies at a point within the violin. For the case where $x = 0$ and $x = 321$ mm, we can conclude that the height of the air column is 32 mm which is the height of the rib structure of a violin. This returns the resonant frequency 5317.1 Hz. Likewise, at the largest point the center (the point at $\{0,0\}$ on the graph) of the violin, the height is equal to 64 mm where 16 mm is the height of the canaries of both plates. This returns a frequency of 2658.1 Hz for the center of the violin. This frequency is note E7 on the musical scale. In addition, 5317 Hz is twice the frequency of E7. In a Stradivarius violin, the frequency of the air column is a D. So my measurements are close for a single point. Again this is only a theoretical model of what is occurring. Certain things have not been accounted for that would make the calculation change, such as the spruce top and maple back; where the 16 mm measurement contains 3.5 mm of spruce and 3.5 mm of maple. In addition, each instrument changes according to its quality.

We have looked mathematically at different eigenfunctions, which allowed us to determine the frequencies of these oscillations, by applying it to sections of a violin top. We can see the two-dimensional Bessel function frequencies have non-linearities to the one-dimensional frequencies. We saw that sections of the violin top produce high frequencies that allow for the top to be set into motion quickly to produce sound from the air column of the violin. The air column at two points on a fixed slice in the center of the violin produces a frequency corresponding to the note E7 or a harmonic of it.

References:

- [1] Abell, Martha, et al. Mathematica by Example, Second Edition Academic Press 1997. pp.537-553.
- [2] Beament, James. The Violin Explained: Components Mechanism and Sound. Oxford University Press 1997.
- [3] Cap, Ferdinand F. Mathematical Methods in Physics and Engineering with Mathematica. Chapman and Hall/CRC 2003. pp. 193-198.
- [4] Lamb, H. The Dynamical Theory of Sound, 2nd ed. London: Edward Arnold, 1925. Chapter V
- [5] Trott, Michael. The Mathematica Guide Book for Symbolics. Springer 2005 1st Edition.
- [6] “Vibrating Flat Things”, Accessed 5/5/07 <
<http://www.du.edu/~jcalvert/waves/membran.htm> >

Appendix A:

```
f[x_]=Sin[π x];
g[x_]=3x+1;
```

$$a_1 = 2 \int_0^1 f(x) dx$$

$$a_n = 2 \int_0^1 f(x) \cos(n\pi x) dx$$

$$b_n = \frac{2 \int_0^1 g(x) \sin(n\pi x) dx}{1 + (-1)^n}$$

$$b_n = \frac{2 \int_0^1 (3x+1) \sin(n\pi x) dx}{n\pi}$$

$$\frac{2n\pi - 8n\pi \cos(n\pi) - 6 \sin(n\pi)}{n^3\pi^3}$$

TableForm[bn, 10]

n	b_n	Value
1	$\frac{10}{\pi^2}$	1.01321
2	$-\frac{3}{2\pi^2}$	-0.151982
3	$\frac{10}{9\pi^2}$	0.112579
4	$-\frac{3}{8\pi^2}$	-0.0379954
5	$\frac{2}{5\pi^2}$	0.0405285
6	$-\frac{1}{6\pi^2}$	-0.0168869
7	$\frac{10}{49\pi^2}$	0.0206778
8	$-\frac{3}{32\pi^2}$	-0.00949886
9	$\frac{10}{81\pi^2}$	0.0125088
10	$-\frac{3}{50\pi^2}$	-0.00607927

```
Clear[u, uapprox]
```

```
u[n_]=Sin[n π t] Cos[n π x]
```

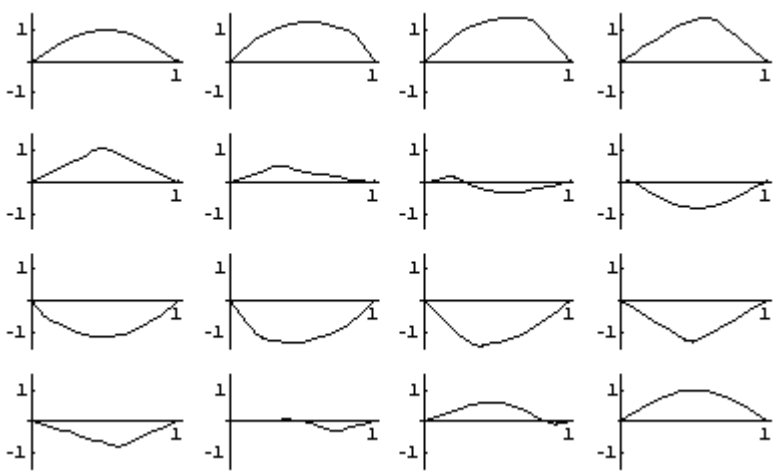
```
uapprox[k_]:=uapprox[k]=uapprox[k-1]+u[k];uapprox[0]=Cos[π t] Sin[π x];
uapprox[10]
```

$\cos p$
 $3 \sin 2$
 $10 \sin 3$
 $3 \sin 4$
 $2 \sin 5$
 $10 \sin 6$
 $3 \sin 8$
 $10 \sin 9$
 $3 \sin 10$

```

someg =
TablePlot[...], 1]
PlotRange[...], 1]
ticks@...
2
5
toshow = Partition[...], 4]
Show[GraphicsArray[toshow]

```




```

"2.404825557695773`", "239798.38289841285`"}, • {"2",
"5.5200781102863115`", "550437.3488062547`"}, • {"3",
"8.653727912911013`", "862910.8056998781`"}, • {"4",
"11.791534439014281`", "1.1757987523535285`*^6"}, • {"5",
"14.930917708487787`", "1.488843924760714`*^6"}, • {"6",
"18.071063967910924`", "1.8019651790922217`*^6"}, • {"7",
"21.21163662987926`", "2.115128952366725`*^6"}, • {"8",
"24.352471530749305`", "2.428318874924432`*^6"}, • {"9",
"27.493479132040253`", "2.7415260183909596`*^6"}, • {"10",
"30.634606468431976`", "3.0547451013029288`*^6"} •
RowSpacings->1, • ColumnSpacings->3, • RowAlignments->Baseline, •
ColumnAlignments->{Left}}, • Function[ BoxForm`e$, • TableForm[
BoxForm`e$]]\ )

```

Bessel Code

```

sValues[n_, {R_, a_}] := Table[s, {s, 0, ℓ[{R, a}], ℓ[{R, a}]/n}] ••
φValues[n_, {R_, a_}] := φ[#, {R, a}] & /@ sValues[n, {R, a}] ••
rValues[n_, {R_, a_}] := r[#, {R, a}] & /@ sValues[n, {R, a}]

TrapezoidalIntegrateC =
Compile[{{data, _Complex, 1}, {h, _Real}},
Module[{sum = 0. + 0. I},
Do[sum = sum + data[[i]], {i, Length[data]}];
h (sum - (First[data] + Last[data])/2.)]];

fourierMatrix[k_, o_, {R_, a_}, pp_] :=
Module[{h[Delta] = N[\[ScriptL][{R, a}]/pp], rData, \[CurlyPhi]Data,
sData, l1, l2, l3},

{rData, \[CurlyPhi]Data, sData} =
Developer`ToPackedArray[N[#pp, {1, 1}]] & /@
{rValues, \[CurlyPhi]Values, sValues};

Do[l1[m] = Exp[N[I] m \[CurlyPhi]Data], {m, -o, o}];
Do[l2[n] = Exp[N[-2Pi I n sData/\[ScriptL][{R, a}]]], {n, -o, o};
(* use stable recursion toward m=0 *)
l3[-o] = Developer`ToPackedArray[BesselJ[-o, k rData]];
l3[-o + 1] = Developer`ToPackedArray[BesselJ[-o + 1, k rData]];
(* use recursion formulas for BesselJ *)
Do[l3[m] = 2. (m - 1.)/(k rData) l3[m - 1] - l3[m - 2],
{m, -o + 2, 0}];
l3[o] = Developer`ToPackedArray[BesselJ[o, k rData]];
l3[o - 1] = Developer`ToPackedArray[BesselJ[o - 1, k rData]];
(* use recursion formulas *)
Do[l3[m] = 2. (m + 1.)/(k rData) l3[m + 1] - l3[m + 2],
{m, o - 2, 1, -1}];
(* the table of integrals *)
Table[TrapezoidalIntegrateC[l1[m] l2[n] l3[m], h[Delta]],
{m, -o, o}, {n, -o, o}]];

With[{k = 1, l = 1, n = 1, R = 1, a = 0.859},
NIntegrate[Evaluate[
Exp[-2 Pi I n s/\[ScriptL][{R, a}]] Exp[I l \[CurlyPhi][s, {R,
a}]] *
BesselJ[l, k r[s, {R, a}]]],
{s, 0, Pi, Pi + 2, 2 Pi + 2, \[ScriptL][{R, a}]]] // Timing

{0.441659 Second, 2.73262 \[InvisibleSpace] -4.40427 \[ImaginaryI]}

det[k_?NumericQ, o_, {R_, a_}, pp_] :=
Det[fourierMatrix[k, o, {R, a}, pp]]

Timing[mat = Table[{k, det[k, 2, {1, 1}, 301]}, {k, 0.1, 6, 1/30}];]

```



```

{50.2989 Second,Null}

Show[Apply[Function[{reim, col},
  (* real part in red; imaginary part in blue *)
  ListPlot[#[[1]], reim#[[2]]]& /@ mat,
    PlotJoined -> True, PlotStyle -> col,
    DisplayFunction -> Identity]],
  {{Re, {Hue[0]}}, {Im, {Hue[0.8], Dashing[{0.01, 0.01]}}}}, {1}],
  DisplayFunction -> $DisplayFunction,
  PlotRange -> {-1, 1}, Frame -> True, Axes -> False]

startPairsN = {{1.7, 1.8}, {2.1, 2.2}, {2.8, 3.0}}
  {{1.7, 1.8}, {2.1, 2.2}, {2.8, 3.}}

kRoots = (k /. FindRoot[Re[det[k, 2, {1, 1}, 301]],
  Evaluate[{k, Sequence @@ #}],
    MaxIterations -> 30, AccuracyGoal -> 10])& /@ startPairsN

  {1.77792, 2.1661, 2.84605}

fourierMatrices =
  Transpose[fourierMatrix[#, 2, {1, 1}, 301]]& /@ kRoots;

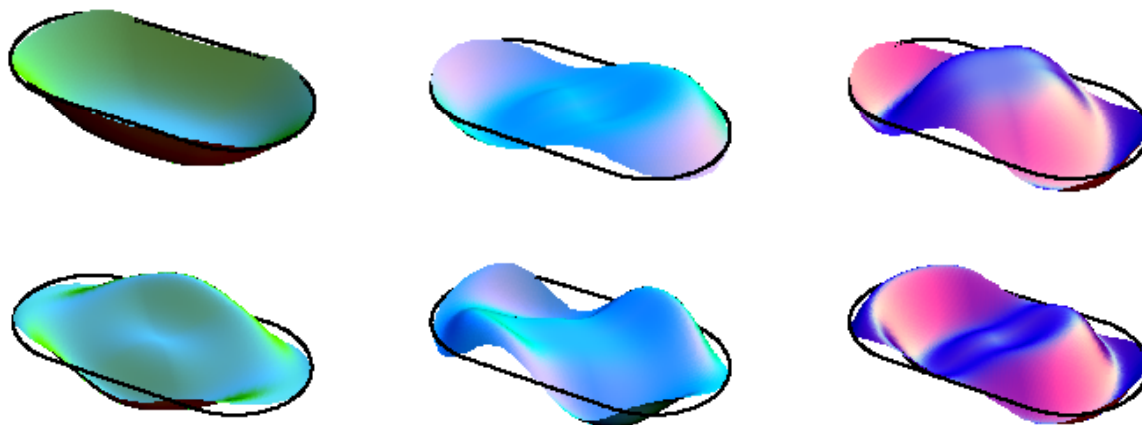
nontrivialHomogeneousSolutions =
  NullSpace[#, Tolerance -> 0.1][[-1]]& /@ fourierMatrices;
\[CapitalPsi][{r_, \[CurlyPhi]_}, k_, v_] := v.Table[BesselJ[1, k r]
Exp[I l \[CurlyPhi]],
  {1, -(Length[v] - 1)/2, (Length[v] - 1)/2}]
eigenfunctionPicture[k_, v_, {pp\[CurlyPhi]_, ppr_}, col_, opts___] :=
Module[{points, polys, boundary},

boundary = {Thickness[0.01], GrayLevel[0],
  Line[Table[1.006 r[s, {1, 1}]*
    {Cos[\[CurlyPhi][s, {1, 1}]], Sin[\[CurlyPhi][s, {1,
1}]]], 0},
    {s, 0, 2Pi + 4, (2Pi + 4)/501}]]];

points = Table[\[Rho] = N[\[Alpha] r[s, {1, 1}]]; \[Phi] =
N[\[CurlyPhi][s, {1, 1}]];
  {\[Rho] Cos[\[Phi]], \[Rho] Sin[\[Phi]],
Re[\[CapitalPsi][{\[Alpha] \[Rho], \[Phi]}, k, v]]},
  {s, 0, \[ScriptL][{1, 1}], \[ScriptL][{1, 1}]/pp\[CurlyPhi]},
{\[Alpha], 0, 1, 1/ppr}];

polys = Table[Polygon[{{#[[i, j]], #[[i, j + 1]], #[[i + 1, j + 1]],
  #[[i + 1, j]]}&[points]], {i, pp\[CurlyPhi]}, {j,
ppr}];
(* show graphics *)
Show[Graphics3D[{boundary, EdgeForm[], col, polys}], opts,
  PlotRange -> All, Boxed -> False, BoxRatios -> {4, 2, 1}]]
Show[GraphicsArray[
Table[eigenfunctionPicture[kRoots[[k]],
  nontrivialHomogeneousSolutions[[k]], {201, 101},
  SurfaceColor[Hue[k/4], Hue[k/4 + 0.4], 2.8],
  DisplayFunction -> Identity], {k, 3}]]]

```



```
[Lambda] = {1.777917044847115`, 2.1660952637765187`, 2.846053213908279`,
3.137975964274985`, 3.405413536789092`, 3.8814401597332533` }
```

```
c = 7977.24;
```

```
Table[{ \ [Lambda], c*\ [Lambda]/.063}]// TableForm
```

```
\!\(\)*
```

```
TagBox[GridBox[{
  {"1.777917044847115`", "2.1660952637765187`",
"2.846053213908279`"},
  {"225124.935981527`", "274277.1711429936`",
"360375.3895256775`"}
}],
Table[{ \ [Lambda], c*\ [Lambda]/.070}]// TableForm
```

```
TagBox[GridBox[{
  {"1.777917044847115`", "2.1660952637765187`",
"2.846053213908279`"},
  {"202612.44238337426`", "246849.45402869422`",
"324337.8505731097`"}
]
```

```
TagBox[GridBox[{• {"3.137975964274985`", "3.405413536789092`",
"3.8814401597332533`"}, • {"397339.4822421108`",
"431203.1917811971`", "491479.04285445233`"}
]
```

Appendix C:

