25-29 August 2024
inter.noise
NANTES FRANCE

Experiential learning with online vibro-acoustic demonstrators

Stephen Hambric[1]
Hambric Acoustics, LLC
Asheville, North Carolina, USA

ABSTRACT

After 25 years of delivering lecture-based learning at a university with mixed success I now supplement my short courses with online experiential learning demonstrators for visualizing and interacting with vibro-acoustic systems. The demonstrators use simple javascript coding and calls to the plotly free libraries. No special software is required (but stronger computers obviously perform faster). Learners can quickly adjust parameters like stiffnesses, masses, and damping using sliders and immediately visualize changes to vibration response and sound fields. Students may exercise demonstrators for one- and two-dimensional acoustic waves; vibrations and mode shapes of simple oscillators, beams, and plates; and sound radiated by and transmitted through panels. Some of the demonstrators are freely available at hambricacoustics.com (others are only available to my short course students).

1. INTRODUCTION

Most people learn by doing. Music students play pianos, guitars, and other instruments. Art students paint, make pottery, or blow glass. Engineering students, however, are forced to sit and listen to an hour (or more) of instruction from a professor on complex topics and somehow remember the important points without actually doing anything. Later, sometimes much later (well past the point where the human brain can strengthen an important memory) the student sits at their desk or computer and works on homework problems that usually confuse, confound, and frustrate. Subsequent lectures, without the student fully comprehending the earlier material, frustrate even more.

This is not a good way to learn. During my 25 years of teaching vibration and acoustics I always pondered how engineering students can learn more efficiently. What they need is some sort of instrument or medium that they can work with interactively and immediately visualize the behavior. This is how all sentient beings learn – by doing something and observing the reaction. How then to develop such an instrument for acoustics and vibration?

For many years I used commercial scientific software packages to develop interactive demonstrators, but they required expensive licenses. The software companies also frequently issue updates which 'broke' my codes. After years of sporadic internet searches and discussions with web programmers I finally found a suitable approach: simple .html and javascript codes [1] which call the math.js math library [2] and the plotly plotting library [3] for generating interactive web-based 2D and 3D plots. The codes are stored on a hosting platform and all calculations and graphical displays are done on the user's computer and displayed in their browser. No licensing fees are required, and if I am careful to use well established modern coding methods browser updates should not 'break' the demonstrators (they haven't so far anyway).

---

[1] hambricacoustics@gmail.com

In this paper I briefly discuss my general coding approaches and show and discuss screenshots of some of the demonstrators. Of course, the only way to experience them is to go online and try them yourselves.[2] Only the most basic demonstrators are freely available, however; the more complex ones are reserved for my short course students (who have access for a year after their course).

2. CODING

Each demonstrator is coded in an .html file and I use a common cascading style sheet (css) for page formatting. Several javascript functions to compute quantities like vibration and sound are declared toward the front of the file, followed by a main section which sets up the initial case and calls plotly to display the responses. The file ends with additional javascript functions which manipulate the plots (by changing parameters like geometry and material properties) and ends with .html commands to display the controls – slide bars, animation buttons, selection boxes, and others. The html controls link back to the javascript functions which adjust the responses and update the plots. Since the data are stored within the plot objects, plotly makes it simple and fast to simply update the data and refresh the plot.

I use javascript [1] for all the demonstrator calculations[3]. Javascript is not intended for complicated math and is trickier to write than python and other languages, particularly for arrays and matrices (which are quite different). The math.js [2] library has been invaluable, although it has its quirks. It can take a lot of time and trial and error to work out new code. I highly recommend using the 'developer tools' pane in Google Chrome to examine variables and catch errors and warnings. I also often use the javascript 'console' command to do simple writes of variables throughout the coding process to make sure I'm doing things properly. There are likely fancier debugging tools available, but my old-fashioned methods serve me well.

Plotly originated as a plotting library for python codes. The developers adapted it for javascript [3], which makes these demonstrators possible. I am grateful to this team! You can plot simple curves, surfaces, and even solids with clever visualization options. The online documentation is detailed and you can change just about any property you like to improve the look and feel of a plot, including allowing users to zoom in and out, pan, and hover over data to see actual values. Plotly also allows some additional interactivity with sliders and selections, but my demonstrators became too complicated with multiple plots to use that functionality. Instead, I use html-based inputs, select boxes, buttons, and others which control all of the plots.

3. DEMONSTRATORS

During my short courses I pause for a few minutes after discussing each major topic for the students to interact with the demonstrators. I walk around the room and provide guidance to those who need it. I encourage the students to work with the demonstrators after class and during breaks and many of them do, leading to good questions for the group to discuss later.

The performance of the demonstrators depends on the strength of the student's computer. The simpler ones work well on any level of computer power, but the complex ones with contour plots which animate over frequency cycles, can be slow on older machines. Each demonstrator includes several paragraphs of explanation at the top of the page along with suggestions on how to use the controls. Below are screenshots and short descriptions of some of the demonstrators. Once again, I encourage you to visit the website and try them yourself as this is, of course, the best way to learn about them!

---

[2] https://www.hambricacoustics.com/demos/SAH_vibroacoustic_demos.html

[3] There are developmental versions of web-based python available, but to date they are much to slow to use for interactive demonstrators. I continue to monitor their progress, however, and may migrate to python when practical.

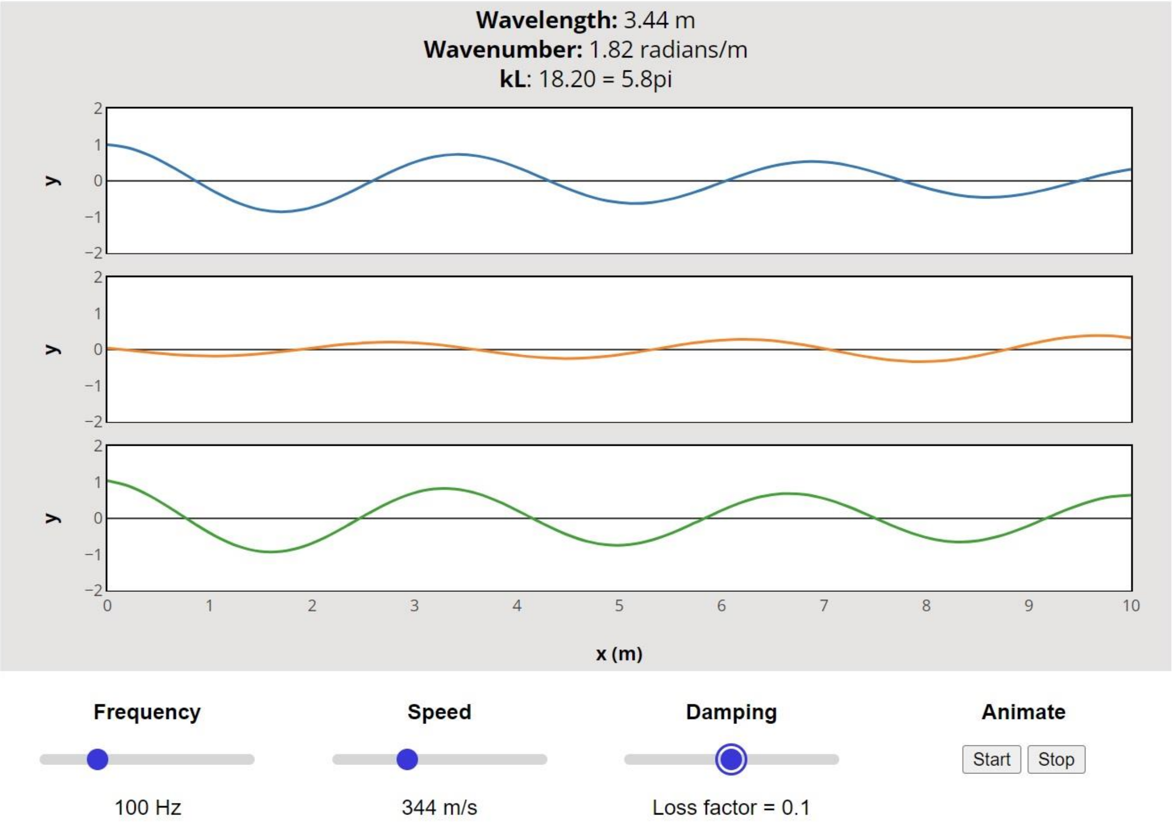
3.1. Wave Analysis

Figure 1 shows simple propagating and standing waves. Three waves are shown in the plots. Animate them by clicking on the 'start' button under the Animate controls on the bottom right. The top wave is the incident wave which starts on the far left at x=0 and propagates to the right. The middle wave is reflected from a rigid wall on the far right and propagates to the left, back toward the source. The bottom waveform is the sum of the two, which for the default low damping level looks like a standing wave. Although it looks like the animated wave is stationary, it is actually comprised of the underlying left and right traveling waves.

Change the frequency or speed of the wave with the two sliders on the lower left and watch the waveforms lengthen or shrink. You can also track the wavelength, wavenumber, and dimensionless wavenumber kL in the title. kL is also expressed as a multiple of π, where π represents one half wave along the length L.

Change the damping within the wave with the damping slider. The incident wave amplitude decreases with distance from the origin (hover your mouse over the peaks of the waveform to track the decay). The reflected wave is also attenuated with distance from the right wall. The most interesting effect is on the summed wave. If you increase the damping substantially the summed wave looks like the propagating incident wave near the source. However, near the right wall the waveform appears to be standing again. This phenomenon is well known in acoustic enclosures, where the sound near a source appears to be propagating, called a 'direct field', and the sound far from a source and closer to a wall appears to be standing, or 'reverberant'. Change the frequency, wave speed, or damping to adjust the 'breakpoint' between the direct and reverberant regions of the summed waveform.

At very high frequencies, low sound speeds, and high damping the reverberant field is nearly completely suppressed, and the summed wave simply converges to the incident wave, decaying to nearly zero at the right wall (a free-field). Conversely, at very low frequencies, high sound speeds, and low damping the summed wave becomes completely reverberant.



Wavelength: 3.44 m
Wavenumber: 1.82 radians/m
kL: 18.20 = 5.8pi

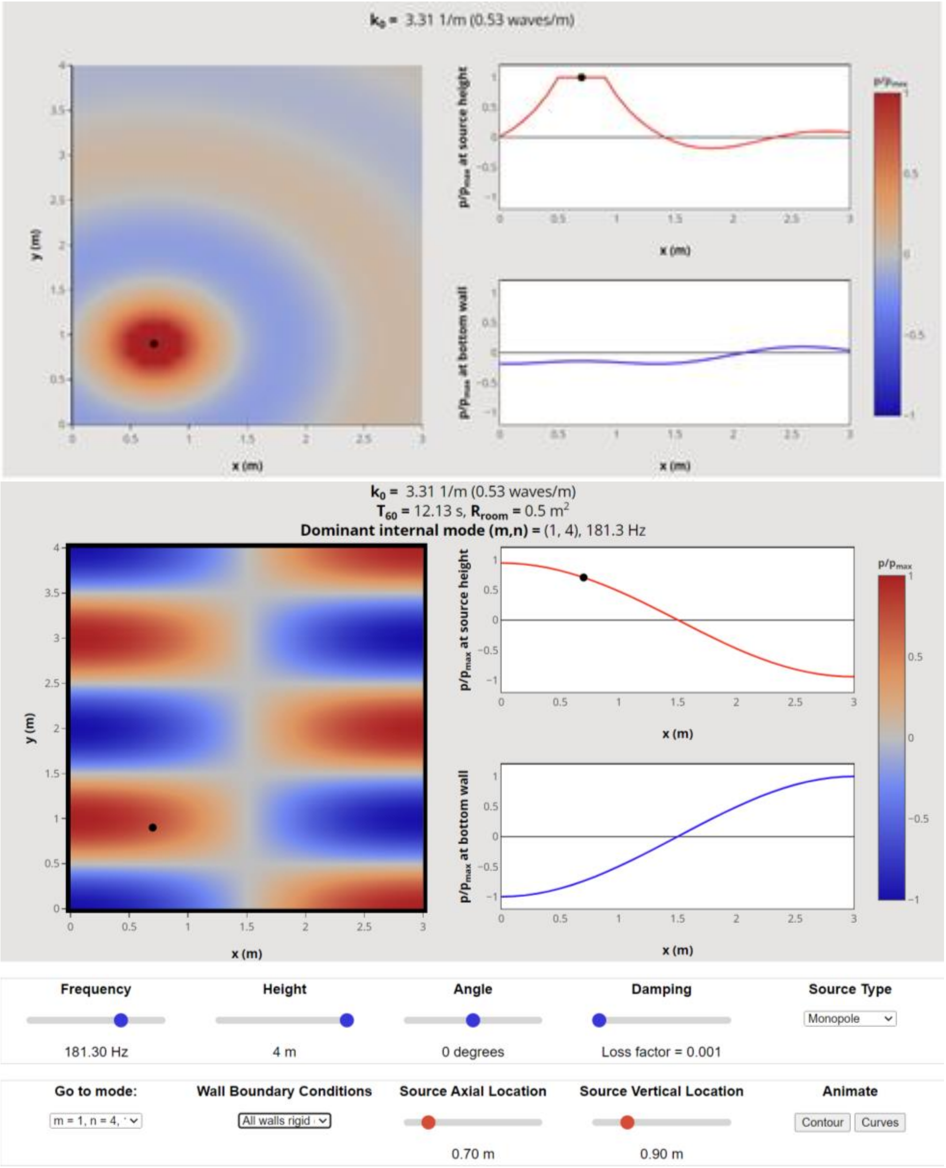
Frequency: 100 Hz
Speed: 344 m/s
Damping: Loss factor = 0.1

Figure 1: Simple wave demonstrator.

3.2. Acoustics

Figure 2 shows a two-dimensional acoustic wave-field induced by a source, along with 'slices' of pressure along the plane of the source and the bottom of the plot (y=0). You can select plane wave, monopole, and dipole source types (the plane wave and dipole sources can be rotated with the 'angle' slider). You can also change the source location with either the sliders or by simply clicking on the contour plot with your mouse. A slider bar adjusts frequency which updates the contour and wave curves automatically, with the wavenumber updated in the title. You can animate the contours or the 2D wave slices.

The most interesting feature in the demonstrator is the ability to change the wall boundary conditions. The default is free (open space) as shown in the top figure. However, you can make the bottom and left walls rigid. As you move the source close to a wall, the classic pressure doubling effect is evident. You can also make all walls rigid to examine internal enclosure modes. The 'go to mode' selector allows you to jump to a particular enclosure mode shape, which in turn changes the frequency. Move the source around to see how well (or poorly) you excite particular modes. This is quite interesting when using dipole sources and by adjusting their orientation with respect to different mode shapes. Finally, add internal acoustic damping to attenuate the internal modes and converge to nearly free-field behavior.



Frequency: 181.30 Hz
Height: 4 m
Angle: 0 degrees
Damping: Loss factor = 0.001
Source Type: Monopole
Go to mode: m = 1, n = 4,
Wall Boundary Conditions: All walls rigid
Source Axial Location: 0.70 m
Source Vertical Location: 0.90 m
Animate

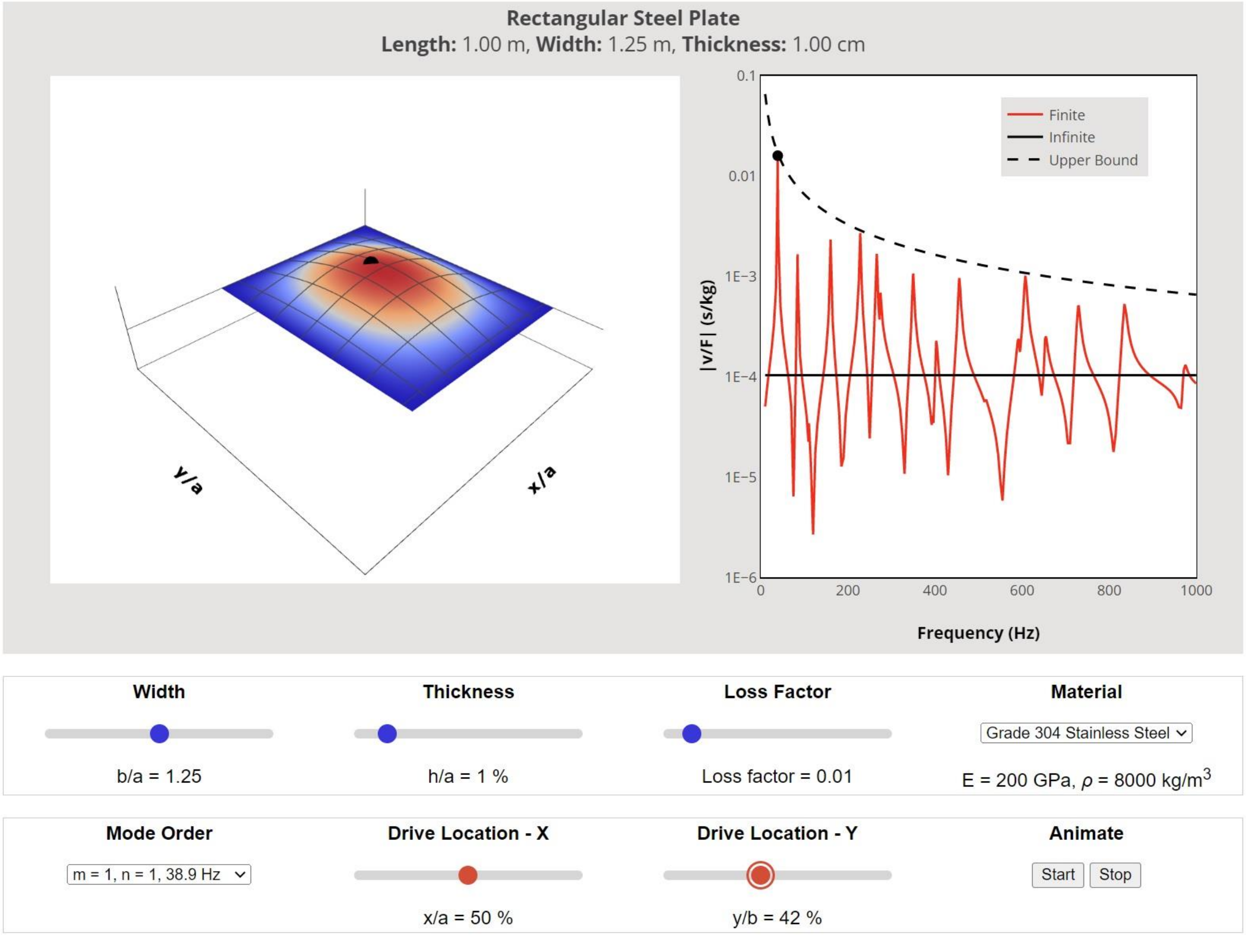
Figure 2: Two-dimensional acoustic waves – top - free field, bottom – rigid walls.

3.3. Vibration

Figure 3 shows the surface mobility of a flat rectangular plate. The controls are similar to those in the earlier demonstrators – you can move the source location (the black dot on the mode shape) with either the sliders or by clicking on the plate. The demonstrator includes the drive point mobility frequency response on the right, along with infinite plate response (the mean curve – see [4] for more on infinite structure theory) and an estimate of the upper bound response (based on the plate mass-damping parameter). You can visualize the effects of changing materials using the selection box. Modal frequencies shift, and the mean and peak mobilities change depending on the material properties. Increasing damping, of course, attenuates the peak (and upper bound) mobilities.

Selecting the mode order changes the mode shape on the left and also the frequency point on the mobility plot (the black circle). You can move the drive location to maximize or minimize a mode's response. Some modes 'disappear' from the mobility plot when you align the drive location at 'nodes', or nulls, of a mode shape. Shift the drive to a region of peak vibration - the 'anti-nodes' and watch the mobility peaks return.

Finally, you can animate the selected mode shape. This required extra coding as the plotly animation approach did not work (for me anyway) for 3D surface contours. Instead, I used the javascript 'setTimeout' function and adjusted the wait time until the animations seemed smooth. This is not great programming and can be slow on older computers.



Rectangular Steel Plate
Length: 1.00 m, **Width:** 1.25 m, **Thickness:** 1.00 cm

Width: b/a = 1.25

Thickness: h/a = 1 %

Loss Factor: Loss factor = 0.01

Material: Grade 304 Stainless Steel; E = 200 GPa, ρ = 8000 kg/m^3

Mode Order: m = 1, n = 1, 38.9 Hz

Drive Location - X: x/a = 50 %

Drive Location - Y: y/b = 42 %

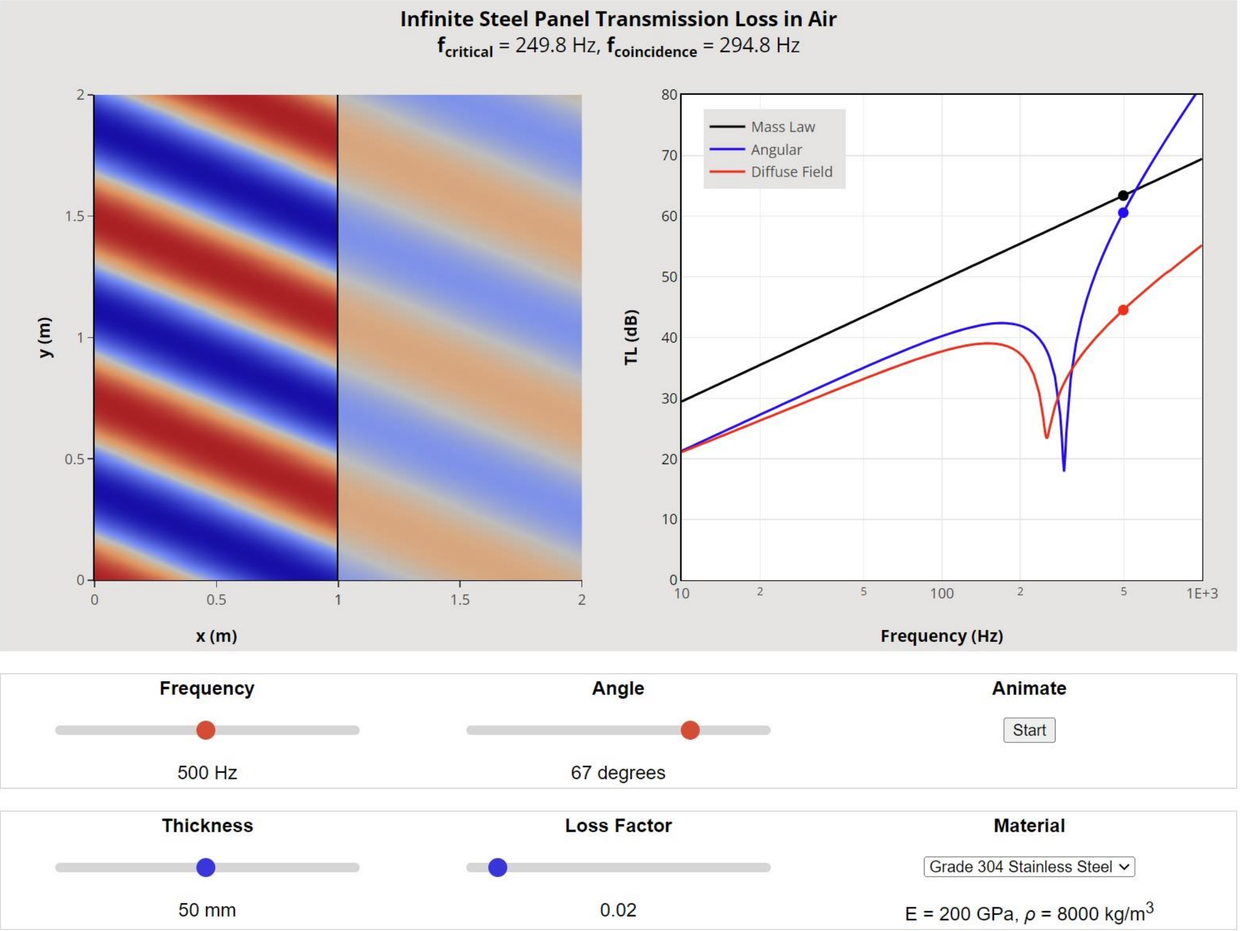
Animate

Figure 3: Rectangular plate vibration.

3.4. Vibro-Acoustics

The demonstrator in Figure 4 shows how a flat infinite panel transmits incident sound. This is the classic transmission loss (TL) calculation which is used often in the noise control community (for more information see [5]). In the left plot I show an incident plane wave on the left and the transmitted wave on the right. An infinite panel between the two (at x=1) partially blocks the incident sound. How much sound is blocked depends on the panel thickness, mass, and damping. In general, the heavier and more highly damped the panel, the less sound is transmitted. Adjust the frequency slider to observe the effects on the incident and transmitted waves. You can also animate them.

Various TL curves are shown in the plot on the right, including the TL at the current angle (which you can adjust with a slider) and the diffuse field TL integrated over angles of incidence between 0 and 90 degrees. The TL curve with angular dependence has a clear dip at the 'pass-band' frequency, called coincidence. Adjust the damping to affect the depth of the coincidence dips. Lower structural damping means more transmitted sound and lower TL. The diffuse field TL responds quite differently to damping – with all TL levels decreasing with increasing damping at and above coincidence – this is because the values above coincidence are the integrations over all the coincidence dips at various angles. Finally, I include the mass-law TL, which is the TL computed at 0 angle of incidence (sound waves normal to the panel). This is the simple one-dimensional case where the mass of the panel is just a lumped parameter added to the fluid impedances. There is no coincidence dip like the ones in the other two curves. Mass law TL increases at a rate of 6 dB/octave.



Infinite Steel Panel Transmission Loss in Air

$f_{critical}$ = 249.8 Hz, $f_{coincidence}$ = 294.8 Hz

Frequency: 500 Hz

Angle: 67 degrees

Animate

Thickness: 50 mm

Loss Factor: 0.02

Material: Grade 304 Stainless Steel

E = 200 GPa, ρ = 8000 kg/m^3

Figure 4: Infinite panel sound power transmission loss.

4. SUMMARY AND CONCLUSIONS

After 25 years of traditional lecture-based teaching of vibration and acoustics I have shifted to an experiential learning approach using online vibro-acoustic demonstrators. I coded them with simple html and javascript calling the math.js and plotly libraries. You can access them at https://www.hambricacoustics.com/demos/SAH_vibroacoustic_demos.html Only the most basic demonstrators are freely available; the more complex ones are reserved for my short course students (who have access for a year after their course).

I have found that my short course students learn vibro-acoustic materials much more quickly with the interactive demonstrators. The few portions of my short course material that I have not yet developed demonstrators for are now always the hardest for me to teach. After a course, I return home to develop new demonstrators to address the topics students had trouble with. As computers continue to improve, and parallel CPUs and powerful GPUs become more commonplace I hope to expand and improve the demonstrators to solve more complex and larger problems.

ACKNOWLEDGEMENTS

I am grateful to the many sponsors and students of my short courses whose support and feedback has allowed me to develop and improve these demonstrators.

REFERENCES

1. W3 schools online reference for HTML, CSS style sheets, and javascript: https://www.w3schools.com/html/
2. Math.js online reference: https://mathjs.org/
3. Plotly online reference: https://plotly.com/javascript/
4. Hambric, S.A., "To infinite and beyond – the amazing uses of infinite structure mobility theory," *Proceedings of Inter-Noise 2019*, Madrid, Spain. https://www.hambricacoustics.com/In19_inf_panel.pdf
5. Hambric, S.A., "Tutorial on infinite panel sound transmission loss simulations," Proceedings of Inter-Noise 2015, San Francisco, CA, USA. https://www.hambricacoustics.com/SAH_tutorial_panel_TL_IN15_rev1.pdf