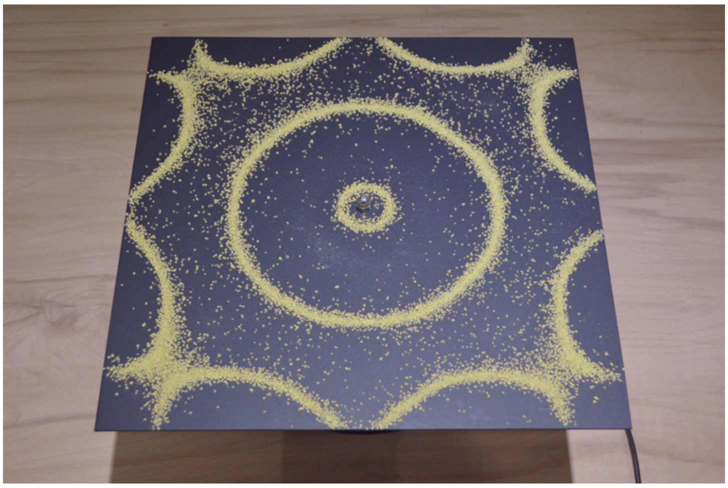
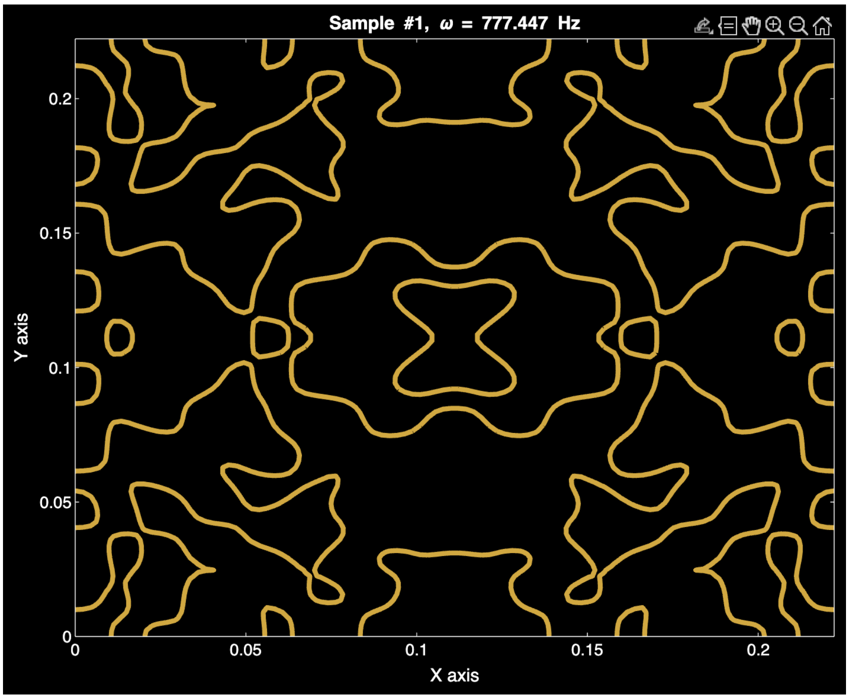
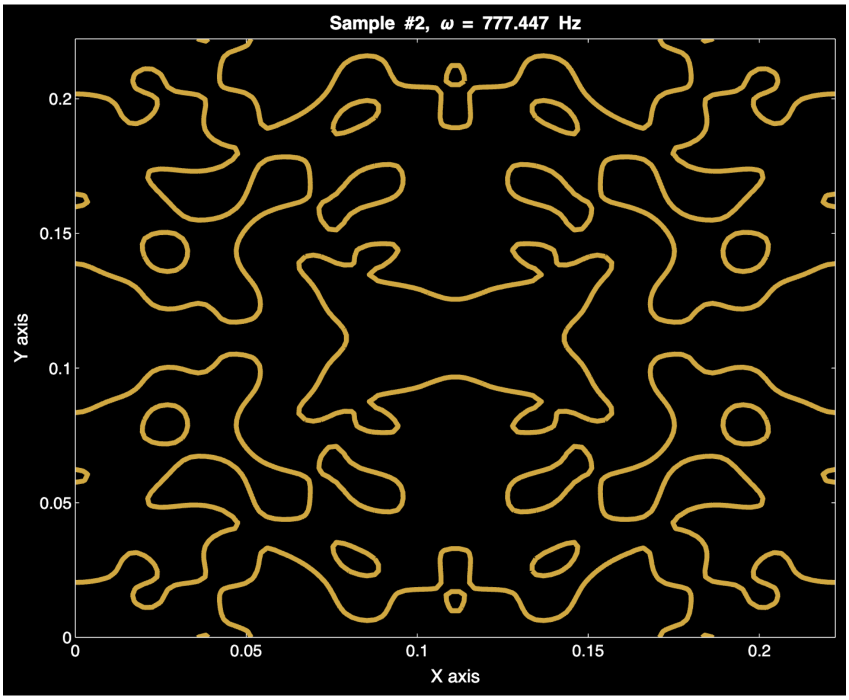
Nice Chladni Plate Benchmark

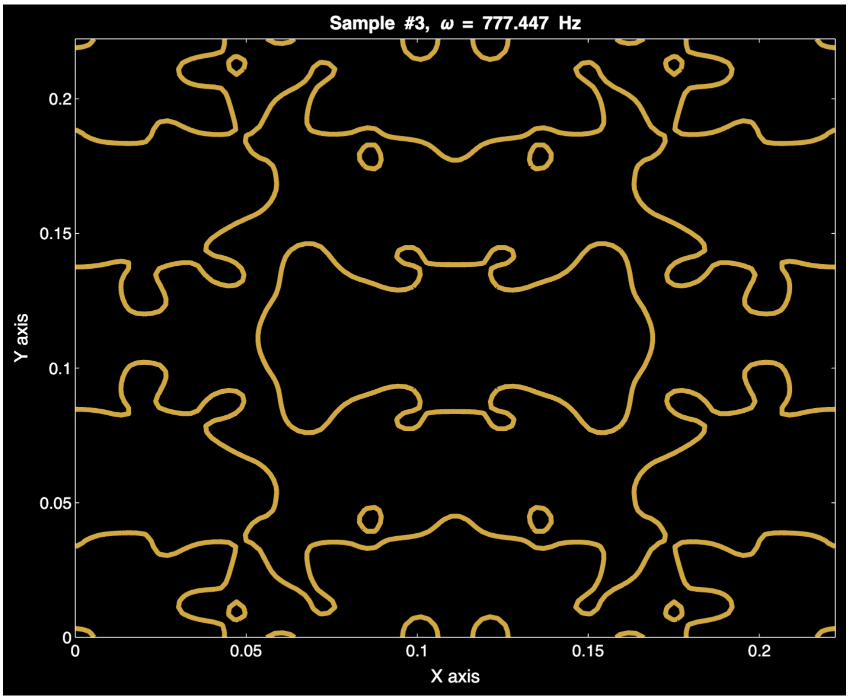
1 Introduction

Chladni plates produce beautiful patterns when excited by a vibrating source. These patterns result from 2D standing waves, visualized by pouring sand on the plate. This document presents the mathematical derivation and solution for these patterns using the 2D wave equation.









2 Mathematical Model

The 2D wave equation for the transverse displacement $u(x, y, t)$ is given by:

$$\frac{\partial^2 u}{\partial t^2} + \gamma \frac{\partial u}{\partial t} = \nu^2 \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) + S(x, y, t), \qquad (1)$$

where:

- γ: Damping constant,

- ν: Transverse wave speed,
- $S(x, y, t)$: Driving force.

The driving force $S(x, y, t)$ is initially modeled as:

$$S(x, y, t) = s_0 \, \delta\!\left(x - \tfrac{L}{2}\right) \delta\!\left(y - \tfrac{M}{2}\right) \cos(\omega t), \tag{2}$$

where δ is the Dirac delta function, s_0 is the source amplitude, and ω is the driving frequency.

2.1 Boundary and Initial Conditions

- **Boundary Conditions (Neumann):**

$$\frac{\partial u}{\partial n} = 0 \quad \text{on all edges}, \tag{3}$$

  which signifies zero normal slope (a "free edge" condition).

- **Initial Conditions:**

$$u(x, y, 0) = 0, \qquad \frac{\partial u(x, y, 0)}{\partial t} = 0, \tag{4}$$

  meaning the plate starts from rest.

3 Solution Procedure

The solution involves the following steps:

3.1 Step 1: Eigenvalues and Eigenfunctions

For a rectangular plate of dimensions $L \times M$ and Neumann boundary conditions, the spatial eigenfunctions and eigenvalues satisfy:

$$\phi_{mn}(x, y) = \cos\!\left(\frac{m\pi x}{L}\right) \cos\!\left(\frac{n\pi y}{M}\right), \quad \lambda_{mn} = \nu^2\left[\left(\frac{m\pi}{L}\right)^2 + \left(\frac{n\pi}{M}\right)^2\right]. \tag{5}$$

3.2 Step 2: Fourier Transform

Apply the (spatial) Fourier transform in x and y to convert spatial derivatives into algebraic expressions.

3.3 Step 3: Laplace Transform

Use the Laplace transform in time to handle the time derivatives. In the Laplace domain, one obtains:

$$s^2\tilde{U} + \gamma s\tilde{U} + \lambda_{mn}\tilde{U} = \tilde{S}(s). \tag{6}$$

3.4 Step 4: Solve for $\tilde{U}$

$$\tilde{U}(s) = \frac{\tilde{S}(s)}{s^2 + \gamma s + \lambda_{mn}}. \tag{7}$$

3.5 Step 5: Inverse Laplace Transform

Take the inverse Laplace transform to recover the time-domain solution for each mode.

3.6 Step 6: Inverse Fourier Transform

Reconstruct the solution in physical space:

$$u(x,y,t) = \sum_{m,n} \frac{\phi_{mn}(x,y)\, \tilde{S}(s)}{\|\phi_{mn}\|^2 \left(s^2 + \gamma s + \lambda_{mn}\right)}. \tag{8}$$

4 Random Forcing Fields for Data-Driven Benchmarks

While a single delta-source at $(L/2, M/2)$ is common in classical derivations, certain data-driven or machine-learning applications require *many different* forcing fields as input. For instance, in *operator learning* (where we train a model to map an input function $S(x,y)$ to an output solution $u(x,y,t)$), a single δ-function forcing is insufficient to explore a broad range of boundary phenomena.

4.1 Definition of the Random Forcing

Instead of

$$S(x,y,t) = s_0\, \delta\!\left(x - \tfrac{L}{2}\right)\, \delta\!\left(y - \tfrac{M}{2}\right)\, \cos(\omega t),$$

we generate a more general, *random multi-mode* forcing of the form

$$S_k(x,y,t) \;=\; \sum_{m=1}^{M_{\max}} \sum_{n=1}^{N_{\max}} \alpha_{mn}^{(k)} \cos\!\left(\frac{m\pi x}{L}\right) \cos\!\left(\frac{n\pi y}{M}\right) f(t), \tag{9}$$

where:

- $\alpha_{mn}^{(k)}$ are *randomly sampled* coefficients for the k-th forcing realization,
- $M_{\max}, N_{\max}$ specify how many modes we include,
- $f(t)$ is some chosen time dependence (e.g., $\cos(\omega t)$ or a more complex function).

4.2 Motivation for Random Forcing

By introducing many different forcing patterns $S_k(x,y,t)$:

- We can generate *training datasets* for operator-learning or surrogate models, which learn a mapping $S(\cdot,\cdot) \mapsto u(\cdot,\cdot,t)$.
- This approach provides a broad set of input-output *examples*, better reflecting real-world scenarios where the driving force is not always localized at a single point.
- It allows us to explore how the plate responds to diverse spatial forcing fields, building a more generalizable model of the plate dynamics.