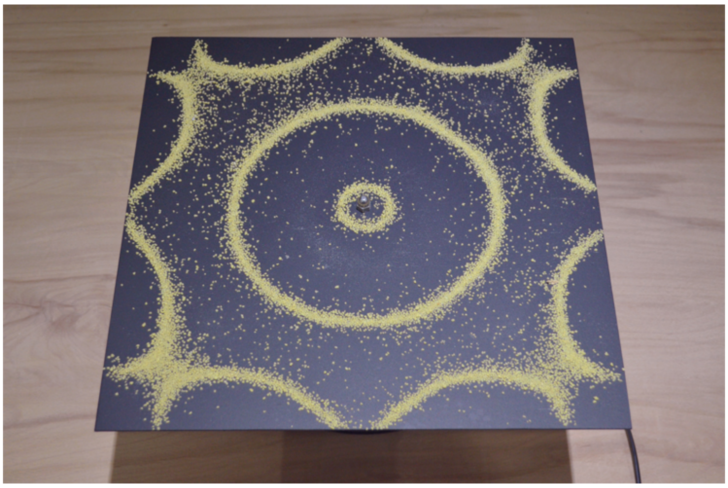
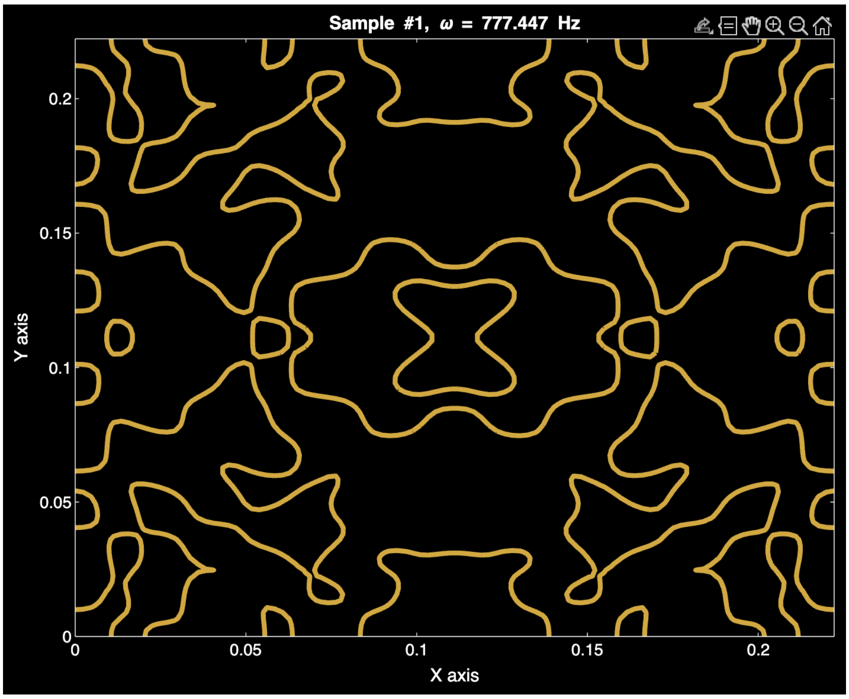
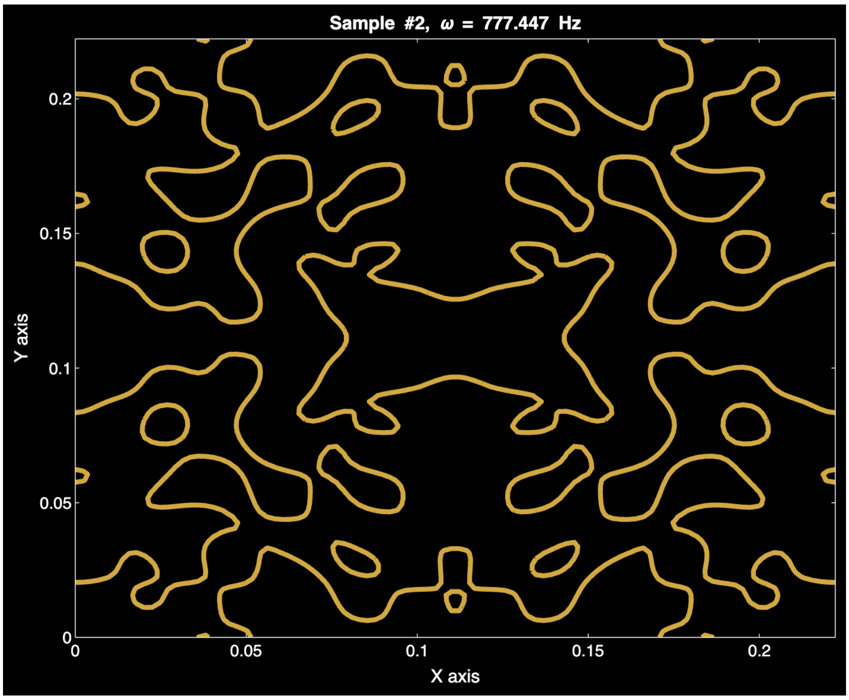
Chladni Plate Inspired Benchmark

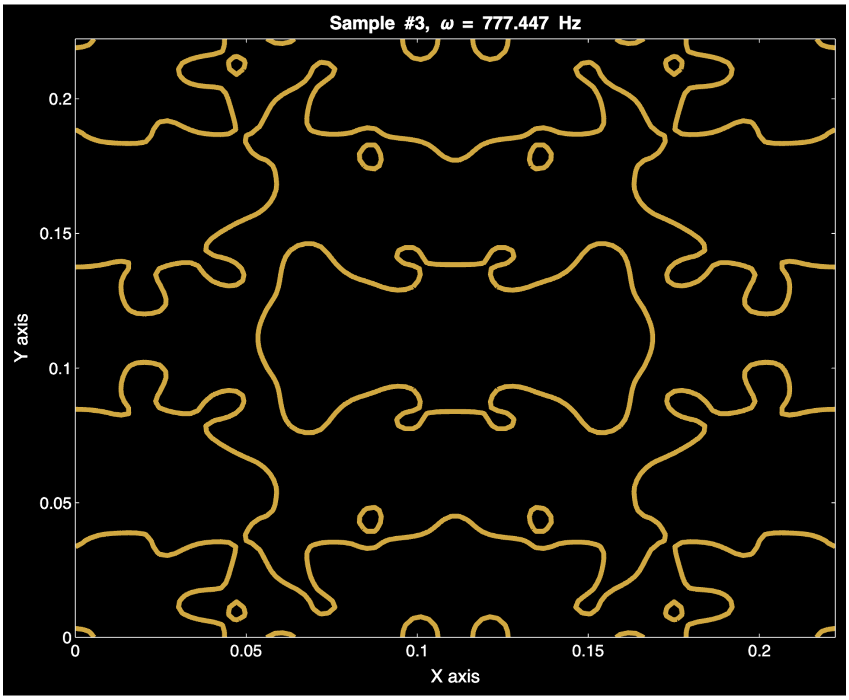
1 Introduction

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.









2 Mathematical Model

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

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

where:

- γ: Damping constant,

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

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

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

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 boundary conditions are applied, meaning:

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

- **Initial Conditions:** The plate starts from rest:

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

3 Solution Procedure

The solution involves the following steps:

3.1 Step 1: Eigenvalues and Eigenfunctions

The eigenfunctions $\phi_{mn}(x, y)$ and eigenvalues λ_{mn} for Neumann boundary conditions are:

$$\phi_{mn}(x, y) = \cos\left(\frac{m\pi x}{L}\right) \cos\left(\frac{n\pi y}{M}\right), \tag{5}$$

$$\lambda_{mn} = \nu^2 \left[\left(\frac{m\pi}{L}\right)^2 + \left(\frac{n\pi}{M}\right)^2\right]. \tag{6}$$

3.2 Step 2: Fourier Transform

Apply the Fourier transform in x and y to eliminate spatial derivatives:

$$\tilde{u}(k_x, k_y, t) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} u(x, y, t) e^{-i(k_x x + k_y y)} dx dy. \tag{7}$$

3.3 Step 3: Laplace Transform

The Laplace transform with respect to t eliminates time derivatives:

$$\mathcal{L}\{u\}(s) = \int_0^{\infty} u(x, y, t) e^{-st} dt. \tag{8}$$

In the Laplace domain:

$$s^2 \tilde{U} + \gamma s \tilde{U} + \lambda_{mn} \tilde{U} = \tilde{S}(s). \tag{9}$$

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

Solve for $\tilde{U}$ in the Laplace domain:

$$\tilde{U} = \frac{\tilde{S}(s)}{s^2 + \gamma s + \lambda_{mn}}. \tag{10}$$

3.5 Step 5: Inverse Laplace Transform

Perform the inverse Laplace transform:

$$U(x, y, t) = \mathcal{L}^{-1}\left(\frac{\tilde{S}(s)}{s^2 + \gamma s + \lambda_{mn}}\right). \tag{11}$$

3.6 Step 6: Inverse Fourier Transform

Reconstruct the solution using the inverse Fourier transform:

$$u(x, y, t) = \sum_{m,n} \frac{\phi_{mn}(x, y)\tilde{S}(s)}{\|\phi_{mn}\|^2(s^2 + \gamma s + \lambda_{mn})}. \tag{12}$$

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

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

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

$$S_k(x, y, t) \;=\; \sum_{m=1}^{M_{\max}} \sum_{n=1}^{N_{\max}} \alpha_{mn}^{(k)} \cos\Big(\frac{m\pi x}{L}\Big) \cos\Big(\frac{n\pi y}{M}\Big)\, f(t), \tag{13}$$

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.