Numerical Chladni Figures

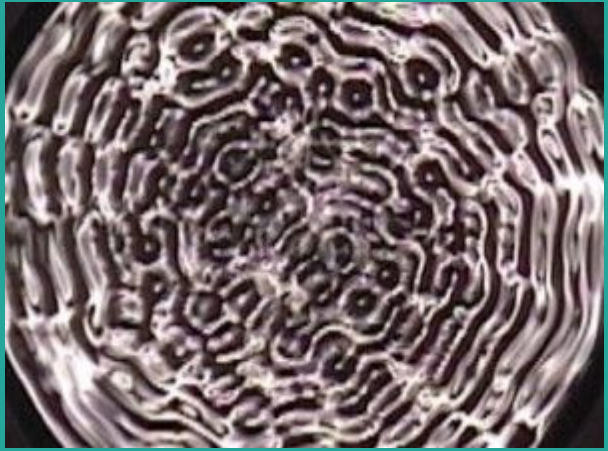
for a vibrating square plate with free boundary condition

Taradutt Pattnaik
13589

Chladni's Experiment

Why is it interesting …

- Visual representation of waves
- Seen in daily life …
  sliding a glass of water
  drums
- Standing bell and healing sounds
- Beautiful patterns even so more for complex geometries



Mathematical model[#4]

Let $z = z(x, y, t)$ be the vertical position of the plate at point (x, y) at time t, then z obeys the following:

$$\frac{\partial^2 z}{\partial t^2} = -\mathcal{L}z$$

The precise form of $\mathscr{L}$ depends on the model used.

Next motion can be analyzed using standing wave solutions of the form $z(x, y, t) = T(t)u(x, y)$ is a standing wave, where $u(x, y)$ is a solution of the eigenvalue problem[#1]:

$$\mathscr{L}u=\lambda u$$

Based on energy arguments we get the conditions given in the next slide.

Problem that we solve:

We need u and λ that satisfy the following conditions:

$$u_{xxxx}+2u_{xxyy}+u_{yyyy}=\lambda u, \qquad (x,y)\in\Omega$$ ...c1

$$u_{yy}+\mu u_{xx}=0, \qquad u_{yyy}+(2-\mu)u_{xxy}=0, \qquad y=\pm H,\ x\in(-L,L)$$

$$u_{xx}+\mu u_{yy}=0, \qquad u_{xxx}+(2-\mu)u_{xyy}=0, \qquad x=\pm L,\ y\in(-H,H)$$ ...c2

$$u_{xy}=0, \qquad (x,y)=(\pm L,\pm H)$$ ...c3

c1 is just $\Delta^2=\lambda u$ and c2 and c3 are the boundary conditions for edge and corner respectively.[#3]

Approach$^{\#4}$:

$$u_{xxxx} + 2u_{xxyy} + u_{yyyy} = \lambda u, \qquad (x, y) \in \Omega \qquad \text{...c1}$$

$$u_{xx} + \mu u_{yy} = 0, \qquad u_{xxx} + (2-\mu)u_{xyy} = 0, \qquad x = \pm L,\ y \in (-H, H)$$

$$u_{yy} + \mu u_{xx} = 0, \qquad u_{yyy} + (2-\mu)u_{xxy} = 0, \qquad y = \pm H,\ x \in (-L, L) \qquad \text{...c2}$$

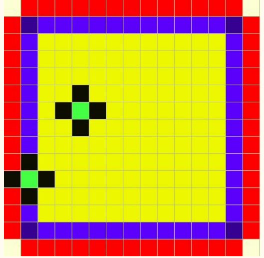
$$u_{xy} = 0, \qquad (x, y) = (\pm L, \pm H) \qquad \text{...c3}$$

As c1 is Δ^2 we are tempted to use 5-point discrete laplacian composed on itself but…

The boundary conditions c2 and c3 cannot be incorporated into that.

So we use Finite Volume Method(FVM) which is able to take care of that.

Solution[#1]:



For FVM we integrate both sides of

$$-\Delta w = \lambda u$$

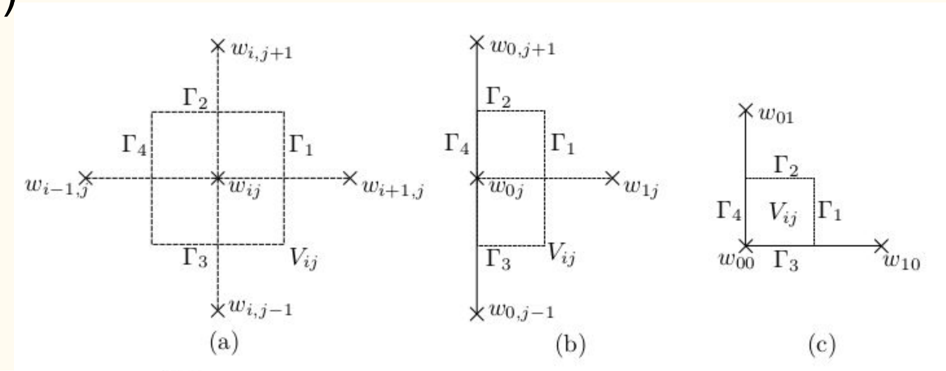
over the control volumes of nodes.

Discretise the square Ω = (−1, 1) × (−1, 1) into (N+1)*(N+1)nodes as u_{ij}

Spacing = h = 2/N

Define w = −Δu, which in discrete form is

$$w_{ij} = \frac{4u_{ij} - u_{i-1,j} - u_{i+1,j} - u_{i,j-1} - u_{i,j+1}}{h^2}$$



$$-\iint_{V_{ij}} \Delta w \, dx\, dy = \lambda \iint_{V_{ij}} u(x,y)\, dx\, dy$$

$$-\int_{\partial V_{ij}} \frac{\partial w}{\partial n}\, ds = \lambda \iint_{V_{ij}} u(x,y)\, dx\, dy$$

Ghost point for edges.

$$\iint_{V_{ij}} u(x,y)\, dx\, dy \approx |V_{ij}| u_{ij} = \begin{cases} h^2 u_{ij} & \text{for interior node} \\ \frac{1}{2} h^2 u_{ij} & \text{for edge nodes,} \\ \frac{1}{4} h^2 u_{ij} & \text{for corner nodes} \end{cases}$$

So we have

$$\Delta^2 u = \lambda u \iff -\Delta w = \lambda u$$

RHS=Bu, For LHS, we hv to find fluxes for each node

Finding LHS for $$-\int_{\partial V_{ij}} \frac{\partial w}{\partial n}\, ds = \lambda \iint_{V_{ij}} u(x,y)\, dx\, dy$$

For interior nodes:

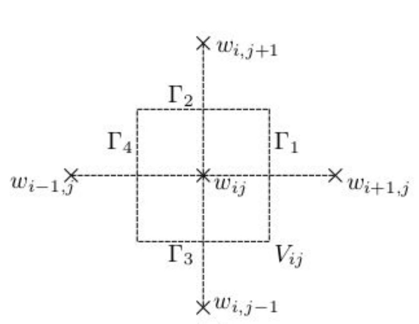
The flux along each piece of ∂V_{ij} is simply approximated by a finite difference. For example, along Γ_1 we have,

$$\int_{\Gamma_1} \frac{\partial w}{\partial n}\, ds \approx h \cdot \frac{w_{i+1,j} - w_{ij}}{h}$$

Similarly doing for τ_2,τ_3 and τ_4 we get

$$-\int_{\partial V_{ij}} \frac{\partial w}{\partial n}\, ds \approx 4w_{ij} - w_{i-1,j} - w_{i+1,j} - w_{i,j-1} - w_{i,j+1}$$

Which is same as discrete Laplacian Δ.



(a)

Implementation in code:

We start by making two discrete laplacians D1 and D2. first D1 acts on Vector containing ALL the nodes(interior,boundary,corner and ghost) say **u** to extract **w**, then D2 acts on **w**. We do this splitting for a reason to be discussed next.

Finding LHS for $-\int_{\partial V_{ij}} \frac{\partial w}{\partial n}\, ds = \lambda \iint_{V_{ij}} u(x,y)\, dx\, dy$

For edge nodes:

Consider a left boundary node at (0,j) for it we have

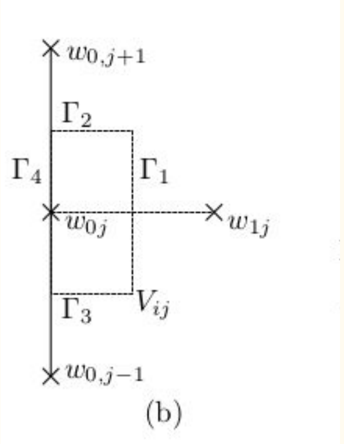
$$-\int_{\partial V_{0j}} \frac{\partial w}{\partial n}\, ds \approx 2w_{0j} - w_{1j} - \frac{1}{2}w_{0,j-1} - \frac{1}{2}w_{0,j+1} - \int_{\Gamma_4} \frac{\partial w}{\partial n}\, ds$$

Next we use boundary conditions to get:

$$\begin{aligned}
\int_{\Gamma_4} \frac{\partial w}{\partial n}\, ds &= \int_{y_{j-1/2}}^{y_{j+1/2}} \frac{\partial}{\partial x}(u_{xx} + u_{yy})\, dy \\
&= \int_{y_{j-1/2}}^{y_{j+1/2}} \big[\underbrace{u_{xxx} + (2-\mu)u_{xyy}}_{=0} - (1-\mu)u_{xyy}\big]\, dy \\
&= -(1-\mu)\int_{y_{j-1/2}}^{y_{j+1/2}} u_{xyy}\, dy \\
&= -(1-\mu)\left[u_{xy}(x_0, y_{j+1/2}) - u_{xy}(x_0, y_{j-1/2})\right].
\end{aligned}$$

where $u_{xy}(x_0, y_{j-1/2}) \approx \dfrac{u_{1,j} - u_{-1,j} - u_{1,j-1} + u_{-1,j-}}{2h^2}$

$D_2 D_1 \mathbf{u}$



Implementation:

Replace the last integral as w(-1,j) ie value ghost point to left of w(0,j). And update the coefficients in D_2. Now last integral depends only on u values of neighbours so make necessary changes in D1.

For corner nodes: The procedure is same as that of edge.

Final touches

$$\iint_{V_{ij}} u(x,y)\,dx\,dy \approx |V_{ij}|u_{ij} = \begin{cases} h^2 u_{ij} & \text{for interior node} \\ \frac{1}{2}h^2 u_{ij} & \text{for edge nodes,} \\ \frac{1}{4}h^2 u_{ij} & \text{for corner nodes} \end{cases}$$

Eliminating ghost points:

Now that we have all the boundary conditions ready, D_1D_2 u gives the LHS, but RHS doesn't have ghost points. So we have to eliminate them.This is done by using the edge boundary conditions

$$u_{xx} + \mu u_{yy} = 0 \quad \text{for } x = \pm 1, \qquad u_{yy} + \mu u_{xx} = 0 \quad \text{for } y = \pm 1$$

Which gives for example

$$u_{-1,j} = 2(1+\mu)u_{0j} - u_{1j} - \mu u_{0,j-1} - \mu u_{0,j+1}$$

And similarly.

These relations are used to delete ghost nodes from vectorised **u** and reducing D_1D_2 to say $A_0$to match size of **u.** This is achieved by taking Schur's complement wrt ghost points.

Next we solve generalised eigen value problem: A_0**u**=λB**u**.

Then we finaly reshape vectorised **u** to get the solution grid for each λ.

Results:

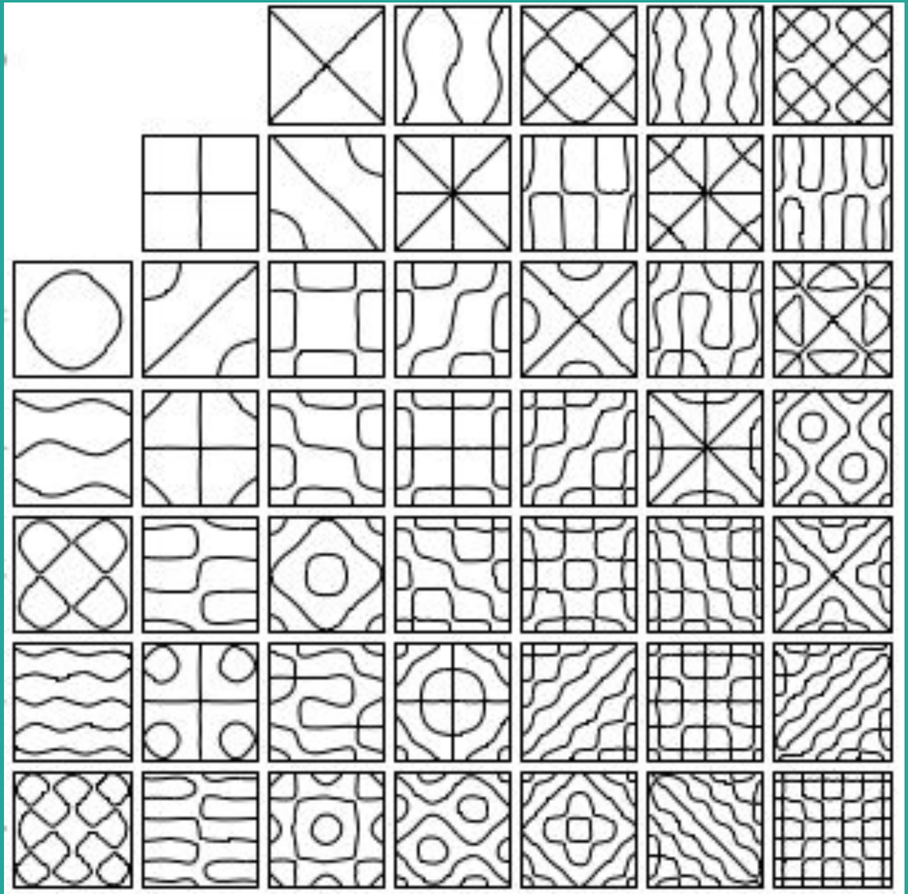
Eigen values(λ) comparion:

λ values where $\lambda = (freq.)^2$		
Computed values		Experiment #1
By program	From #1	
12.4	12.5	12.4
25.9	26.0	26.4
35.6	35.6	36.2
80.8	80.9	77.5
234.9	235.4	215.0
269.0	269.3	260.0
320.1	320.7	310.0
374.6	375.2	364.0
728.7	730.0	698.0

Since λ=freq.^{2},
Experimentally they
are obtained from frequency

Results:

Computed Chladni patterns or eigenfunctions u(x,y) for different eigenvalues(λ):



References

#1 Pg132-142, vol 720, *Spectral theory and Applications*, CRM Summer School

July 4–14, 2016, Université Laval, Québec, Canada

#2 http://cymatica.com/wp-content/uploads/2010/03/dearprudence

#3 http://blog.teachersource.com/2010/05/21/chladni-plates/

#4 *Chladni Figures and the Tacoma Bridge: Motivating PDE Eigenvalue Problems via Vibrating Plates*, Gander2012