Numerical Project:
Multiscale Approach to assist Optomechanics Measurements

Clément Perrochaud

November 2025

with

Benjamin Bacq-Labreuil

clement.perrochaud@etu.unistra.fr
benjamin.bacq-labreuil@ipcms.unistra.fr

Laboratoire Institut de **physique**
et **chimie des matériaux** de Strasbourg | IPCMS | UMR 7504
de l'Université de Strasbourg
et du cnrs

Introduction

The main goal of this numerical project — of which the November 2025 "Project" week is an extension of a voluntary internship carried out in June 2025 [1] — is to model and solve the vibrations of a quasi-2D nano-membrane, hereafter referred to as a "nano-drum". These nano-drums are multilayered membranes with a diameter on the order of $5\ \mu\mathrm{m}$ and a thickness of $7\ \mathrm{nm}$, suspended over a hole and kept in vacuum. They can be set into vibration using LASERs. The numerical results obtained here should then be compared with experimental measurements, led by the group of Arnaud Gloppe, ideally with the goal of predicting their observations.

The main assumption in this study is that the vibrations of the nano-drum can be considered linear. Under this assumption, the quantities of interest are the vibration eigenmodes and their corresponding eigenfrequencies. The starting point will be ab initio calculations. To summarize the structure of this work, we will first (*learn, and then*) use Density Functional Theory (DFT) to obtain the membrane's elastic parameters ab initio — this constitutes the part of the project focused on the November "Project" week. Then, we will construct a vibration model of the nano-drum using these parameters, and finally solve it to obtain the eigenmodes and eigenfrequencies. In particular, we will be studying the vibration of $NiPS_3$ membranes [3].

1 Continuum Physics

With a more dedicated study of continuum mechanics [1], we could show the following equations of motion for a quasi-2D membrane parameterized by $z(x, y, t)$:

1) For an **unstressed homogeneous membrane**:

$$-\partial_{ij} D_{ijkl}\, \partial_{kl} z + A(x, y) \cos(\omega t) = \rho h \partial_{tt} z \qquad (1)$$

with ρ its mass density, h its thickness, with

$$D_{ijkl} := \lambda_{ijkl} h^3/12$$

the stiffness tensor, and where λ is the elasticity tensor, writing $\partial_{ij\dots} := \partial^{\cdots}/\partial x_i \partial x_j \cdots$ and $\partial_{tt} := \partial^2/\partial t^2$, using the Einstein summation notation, and where the gray part is induced by the LASER we shine on the nano-drum.

2) For a **stressed homogeneous membrane**:

$$\frac{1}{2}\partial_j N_{ij}(x, y)\, \partial_i z - \partial_{kl} D_{ijkl}\, \partial_{ij} z + A(x, y) \cos(\omega t) = \rho h \partial_{tt} z \qquad (2)$$

with $N_{ij}(x, y) := \lambda_{ijkl} T_{kl}(x, y) h$, where

$$T(r, \theta) = \frac{r^2}{R^2} \begin{bmatrix} \cos^2\theta & \sin\theta\cos\theta \\ \sin\theta\cos\theta & \sin^2\theta \end{bmatrix} T_{\max}$$

3) For a **multilayered stressed membrane**:

$$\frac{1}{2}\partial_j A_{ijkl}\, T_{kl}(x, y) \partial_i z + \partial_{ij} C_{ijkl}\, \partial_{kl} z + A(x, y) \cos(\omega t) = \rho h \partial_{tt} z \qquad (3)$$

assuming the membrane has *symmetrical* layers (like A-B-B-A) to avoid non-linear effects, and writing

$$A := \sum_{a=0}^{N} \lambda^a (z_{a-1} - z_a) \qquad C := \sum_{a=0}^{N} \frac{\lambda^a}{3} (z_{a-1}^3 - z_a^3)$$

over each layers a, with z_a the height of the interface between the layer a and $a+1$.

In all of these results, the single main set of parameters describing the dynamics of the system is the elasticity tensor of the material λ.

2 DFT ab initio: How to get the Elasticity Tensor

The time-independent Schrödinger equation cannot be solved analytically for systems containing more than two interacting particles. Calculating the exact ground-state electron density $n(\vec{r})$ for such interacting systems remains exponentially hard. Density Functional Theory (DFT), introduced in the 1960s by Walter Kohn and Pierre Hohenberg via the two Hohenberg–Kohn theorems, overcomes this limitation by mapping the interacting system to a fictitious non-interacting system of electrons in an effective potential $V_{\text{KS}}[n](\vec{r})$ that reproduces the same density. The corresponding Kohn-Sham Hamiltonian is quadratic, so its diagonalization scales roughly cubically with the system size, depending on the system size. The exchange-correlation functional $E_{\text{xc}}[n]$ is the part of the energy which remain unknown[1], and it must be approximated. In practice, one starts with an initial guess for the electron density and iteratively refines it until self-consistency is achieved. In this project, we use the open-source, large-scale DFT code Abinit [2] to compute the ground-state total energy of the system as a function of atomic displacements. From this energy dependence, the material's elasticity tensor can be extracted.

The first step is to learn how to use Abinit and to ensure that our calculations are correct. After some trial and error with the H_2 molecule (e.g., finding the equilibrium energy and bond length), we can then proceed to study the system of interest: $NiPS_3$.

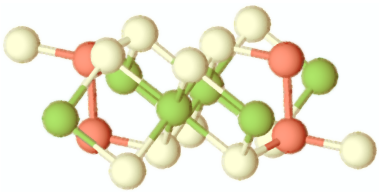

Figure 1: $NiPS_3$ unit cell; with Ni, P & S atoms.

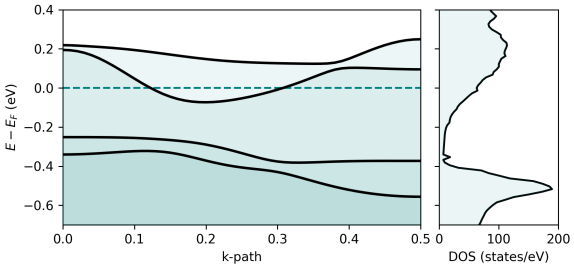

Figure 2: Band structure ($\Gamma \to N$; monoclinic) and density of states of $NiPS_3$, computed with Abinit, and matching with expected results [3].

The elasticity tensor λ is defined as

$$\lambda_{ijkl} := \left.\frac{\partial^2 E}{\partial u_{ij} \partial u_{kl}}\right|_{u_{ij}=0} \qquad (4)$$

with E the energy density (per unit of volume), and u the strain tensor defined as

$$u_{ij} := \frac{1}{2}\left(\frac{\partial r_i}{\partial x_j} + \frac{\partial r_j}{\partial x_i} + \sum_k \frac{\partial r_k}{\partial x_i}\frac{\partial r_k}{\partial x_j}\right)$$

where $\vec{r}$ is the displacement relative to the equilibrium position (this definition comes from continuum mechanics, so one must imagine the entire unit cell being deformed; with indices referring to x, y, or z). With these definitions, and given the ability to compute the system's energy for small displacements of the nuclei, the elasticity tensor can be obtained from DFT. Using the Voigt notation[2], we obtain (5).

[1] The energy can be split like $E[n] = T[n] + E_{\text{H}}[n] + E_{\text{ext}}[n] + E_{\text{xc}}[n]$, where each terms are respectively the kinetic energy of the non-interacting electrons, the Hartree term, the external energy term, and finally the *rest*, which is the unknown exchange-correlation functional.

[2] λ_{ab} with $a \in \{(xx), (yy), (zz), (yz), (xz), (xy)\}$; all other λ-values vanish.

$$\lambda_{\text{NiPS}_3} \approx \begin{bmatrix} 2.2887249 & 0.7369630 & 0.4661706 & 0.0614206 & -0.0776553 & 0.0226973 \\ 0.7369629 & 2.1834731 & 0.5262856 & 0.2040513 & -0.0694516 & 0.0614680 \\ 0.4661681 & 0.5262881 & 1.5346111 & -0.0143981 & 0.0079828 & -0.0480715 \\ 0.0614206 & 0.2040514 & -0.0143998 & 0.5398182 & -0.0523263 & -0.0729190 \\ -0.0776553 & -0.0694515 & 0.0079794 & -0.0523263 & 0.4743805 & 0.0676816 \\ 0.0226973 & 0.0614680 & -0.0480729 & -0.0729190 & 0.0676816 & 0.7306900 \end{bmatrix} \quad (10^2\text{GPa}) \quad (5)$$

3 Numerical Solving

At this point, we have the equation of motion (3) and the parameters (5). We can rewrite this equation as

$$Kz := \frac{1}{\rho h}\big[\frac{1}{2}A_{ijkl}\,\partial_j\,T_{kl}(x,y)\partial_i + C_{ijkl}\,\partial_{ijkl}\big]z = \partial_{tt}z \quad (6)$$

so that is why we can study this problem linearly, and seek the eigenmodes $z_\alpha(x,y)$ and eigenfrequencies f_α of the vibration. We will solve it using finite-element method, dividing $z(x,y)$ as a $(N \times N)$ grid. As we will need to build the operator K as a matrix, we will have to see $z[n,m]$ as a $(1 \times N^2)$ vector (the z_α being eigenvectors) by stacking each slices of the the grid on top of each other $z[Nn+m] \leftarrow z[n,m]$.

3.1 Building *K*

K is here a $(N^2 \times N^2)$ matrix. We need to build it considering the effects of the operators ∂_i — which are the vertical or horizontal derivative — applied to the state $z[n,m]$, as well as on the operator $T_{kl}[n,m]$ — which is non-uniform.

For the sake of simplicity, let us imagine a $(1 \times N)$ grid, what would be the operator ∂ ? Using the growth rate definition $\partial_x f(x) = \frac{f(x+\mathrm{d}x)-f(x-\mathrm{d}x)}{2\mathrm{d}x}$ we can deduce

$$\partial \to \frac{1}{2\mathrm{d}x}\begin{bmatrix} -2 & 2 & 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 & 0 & 1 \\ 0 & 0 & 0 & 0 & -2 & 2 \end{bmatrix}$$

So ∂ is equivalent to a sparse matrix with diagonal $(-1,0,1)/2\mathrm{d}x$. And the very same way:

$$\partial^2 \equiv (1,-2,1)/\mathrm{d}x^2$$
$$\partial^3 \equiv (1,-2,0,2,-1)/8\mathrm{d}x^3$$
$$\partial^4 \equiv (1,-4,6,-4,1)/\mathrm{d}x^4$$

Now we need to restore our grid "as $(N \times N)$", sliced as $(1 \times N^2)$, with vertical and horizontal derivative ∂_x and ∂_y. One could manage to convince oneself that we simply need to use Kronecker product — roughly defined as $(A \otimes B)_{ij} := a_{ij}B$ — giving

$$\partial_x^n \partial_y^m = \partial^n \otimes \partial^m$$

For example:

$$\partial_{xy} \to \frac{1}{2\mathrm{d}x}\begin{bmatrix} -2\partial & 2\partial & 0 & 0 & 0 \\ -\partial & 0 & \partial & 0 & 0 \\ 0 & -\partial & 0 & \partial & 0 \\ 0 & 0 & -\partial & 0 & \partial \\ 0 & 0 & 0 & -2\partial & 2\partial \end{bmatrix}$$

where one must imagine the ∂ developed as sub-matrices like the preceding expression. In some way, one can also see this as a matrix of matrix, each of them affecting the *slice*-vectors of the full $(N \times N)$ z-grid.

So with the initial notation, thanks to the Schwarz theorem and the convention $i \in \{x=0, y=1\}$, we can write $\partial_{ijkl} = \partial^{\,i+j+k+l} \otimes \partial^{\,4-i-j-k-l}$

In the first term of (6), we can split the matrix part as

$$\partial_j T_{kl}(x,y)\partial_i = \underbrace{\big(\partial_j T_{kl}(x,y)\big)}_{:=F_{jkl}(x,y)}\partial_i + T_{kl}(x,y)\partial_{ij}$$

How does $T_{kl}(x,y)$ is applied ? Numerically, we first would build it like a 4D-array $T[k,l,n,m]$. Is should be applied as the sum (remember, Einstein notation) of an operator that multiply each cells of $z[n,m]$, which is actually an vector $z[Nn+m]$. In other words, the applied matrix should be the sum over k and l of a diagonal matrix of which the N^2 long diagonal is the coefficients of our array.

$$T_{kl}(x,y) \to \sum_{kl}\begin{bmatrix} T[k,l,0,0] & 0 & \cdots & 0 \\ 0 & T[k,l,0,1] & & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & T[k,l,N-1,N-1] \end{bmatrix}$$

And then $T_{kl}(x,y)\partial_{ij}$ should simply be the matrix product between the two.

Now, what about $F_{jkl}(x,y) = \partial_j T_{kl}(x,y)$? Here the operator ∂_j apply a derivation onto $T_{kl}(x,y)$ along de j axis. This can simply computed with an usual gradient numerical functions to obtain $F_{jkl}(x,y)$! Now, the form of the product is the same as before (a map multiplied with a derivation operator), so we apply the same rule we applied on $T[k,l,n,m]$ to $F[j,k,l,n,m]$, and multiply the obtained matrix with the one of the the operator ∂_i.

We finally have all the necessary to build K.

3.2 Solving *K*

Now to solve K we *simply* need to diagonalize it. To do it efficiently we can use sparse matrices which will reduce the calculation complexity and memory weight, as K is a matrix of N^4 values.

Also, an other thing to take into account is the shape of the hole on which the nano-drum is suspended, which is not perfectly squared — not a square at all actually. The

shape of the nano-drum has indeed a crucial role on the eigenmodes and eigenfrequencies. To apply this subtlety, we can simply apply a mask onto K. Outside the hole, the membrane does not vibrate at all, so we can imagine that here $z[n, m] = 0$. Thanks to that we can simply exclude those cells out of the grid, and then do the same to the associated line and column of K, which will also reduce the calculation complexity. If we exclude M cells out of the $(N \times N)$ grid, K will be a matrix of $(N^2 - M)^2$ values.

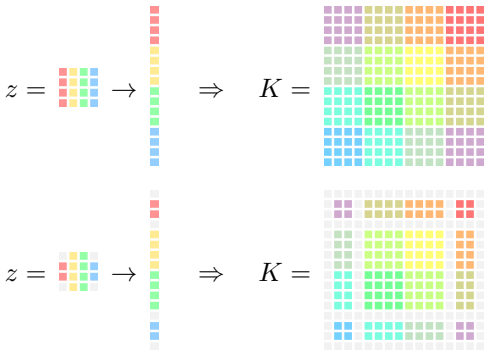

Figure 3: Without mask & with a mask (on each corners).

Then, to yield the true result, we just need to overwrite our cropped result on top of a zero-grid.

3.3 Motion

For the sake of visualization, we can even solve for the motion of the system, which is simply a driven harmonic oscillator. Knowing their eigenmodes $z_\alpha(x, y)$ and eigenfrequencies f_n, we can write

$$z(x, y, t) = \sum_\alpha A_\alpha z_\alpha(x, y) \cos(2\pi f t) \tag{7}$$

with

$$A_\alpha = \frac{A}{4\pi^2 \rho h (f_\alpha^2 - f^2)} \tag{8}$$

where A is the amplitude of the LASER, assumed broad enough to consider A constant.

3.4 Results

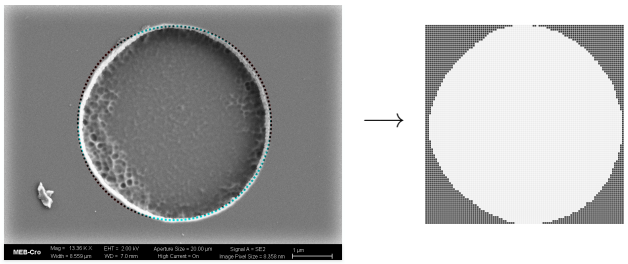

Figure 4: The mask we chose, based on a real hole used in the experiments carried our by the group of A. Gloppe.

Here, we built a (100×100) grid, we use the elastic tensor we got from DFT (5), and we choose the mask shown above we got from electronic microscopy of a hole actually used in experimental for nano-drum setups. Here are our results:

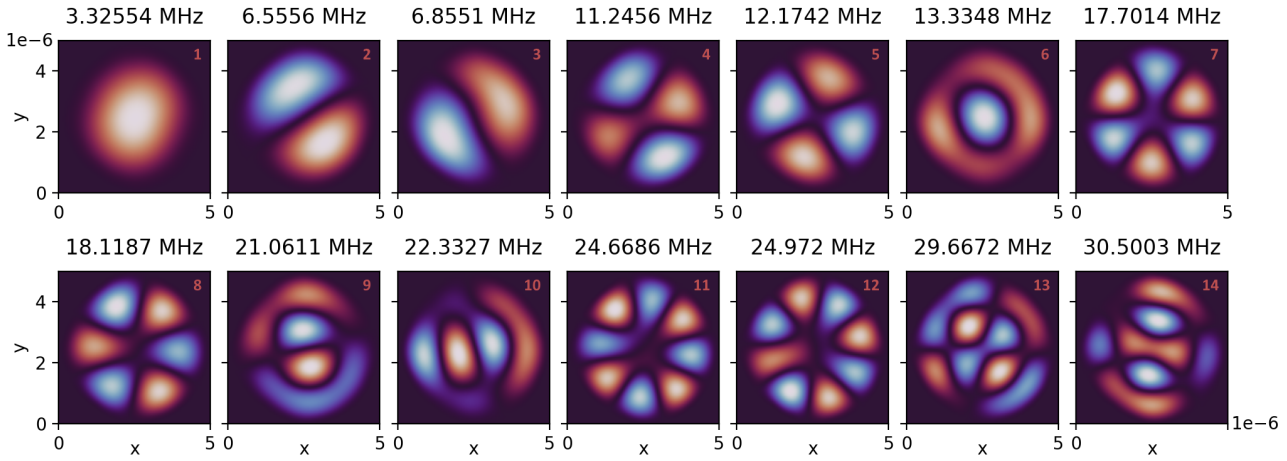

Figure 5: 14 first vibration eigenmodes and their eigenfrequencies.

We can particularly see on the previous figure the anisotropy of the eigenmodes due to the non-symmetries of the $NiPS_3$ crystal and of the hole. We can compare these result with one which would be obtain with an unstressed material (1) and a perfectly circular hole and witness this perturbative organic-looking anisotropy vanish and the eigenfrequencies shift — higher or lower depending on the eigenmode.

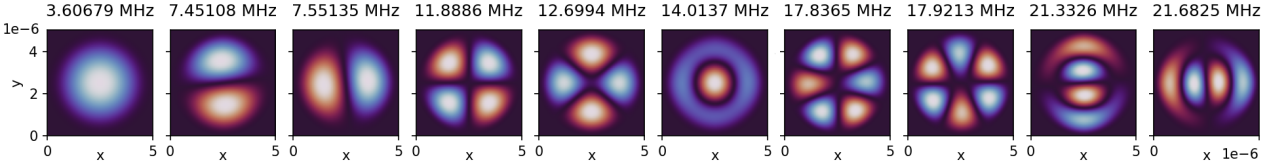

Figure 6: Eigenmodes and eigenfrequencies of a perfectly circular unstressed nano-drum.

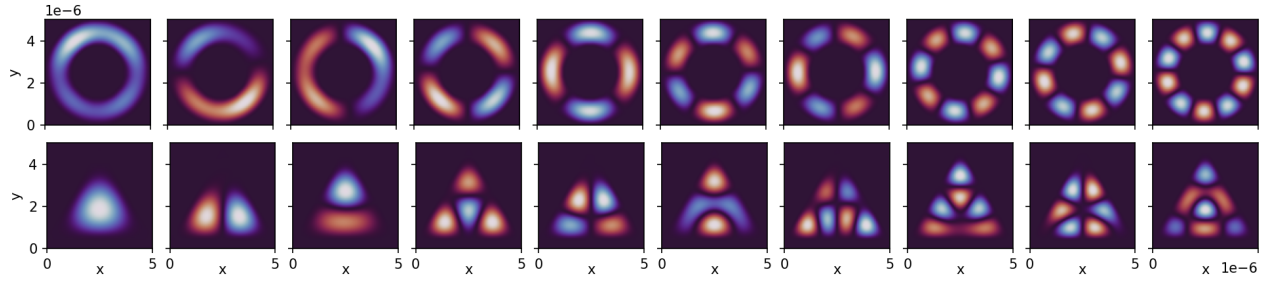

Figure 7: Eigenmodes of more peculiar shapes ("o"-shaped and triangular) of unstressed nano-drums.

And finally we can visualize the superpositions of these eigenmodes induced by the LASER, in motion. All of the python implementation is available in the GitHub repository [1], including the animation script.

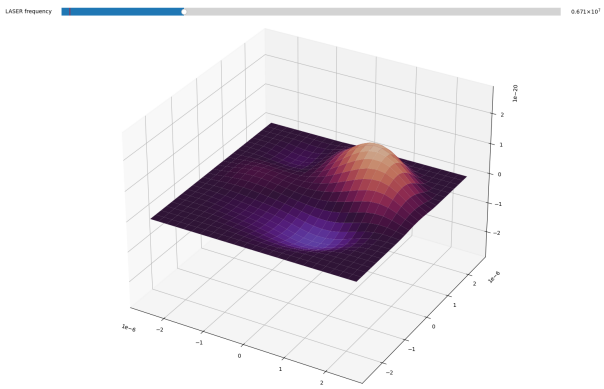

Figure 8: Animation of the dynamical solver for a setup like the Fig.5.

Conclusion

We were here able to create a solver to find the eigenmodes and eigenfrequencies of the vibration of a 2D membrane given the physical expression of its dynamics — assumed linear and involve non-uniform strain and gradient components — and the shape of the hole on which the membrane is placed. These results should be ideally compared to real measurements carried out by the group of A. Gloppe, but unfortunately, we could not obtain them for reference. We can nonetheless appreciate the complexity of the eigenmodes we obtained Fig.5, showing strong anisotropy due to the shape of the hole and the elasticity of $NiPS_3$ — which is itself anisotropic — as well as quite chaotic behaviour at higher frequencies.

We were also able to use DFT ab initio (with Abinit) to obtain the precious elasticity tensor of $NiPS_3$ (5). The main remark here is that our only result is for non-magnetic $NiPS_3$. But in general, this material has a magnetic order, which contributes to electron density, resulting in a different elasticity tensor. In this regard, a great opening would be to find the elasticity with a magnetic $NiPS_3$ and to compare it with real measurements. Additionally, the criticality of the this magnetic order may lead to a temperature dependance of the vibration. These ideas were, however, too ambitious to explore within a single week, alongside the handling of Abinit.

References

[1] Matteo Delenda & Clément Perrochaud, *NanoDrum2025* repository, https://github.com/ClementPerrochaud/NanoDrum2025/tree/main

[2] X. Gonze et al., *ABINIT: First-principles approach to material and nanosystem properties*, https://www.abinit.org

[3] Non magnetic $NiPS_3$, Materials Project, https://legacy.materialsproject.org/materials/mp-676040/

[4] M.J. Rutter, c2x tutorials for Abinit https://www.c2x.org.uk/bands/Si_abinit.html & https://www.c2x.org.uk/DoS/Si_abinit.html

[5] F.Jollet, M.Torrent, Holzwarth, M. Giantomassi, M.J. van Setten, E.Bousquet, M.J. Verstraete, D.R.Hamann, X.Gonze, G.-M.Rignanese, *The PseudoDojo: Training and grading a 85 element optimized norm-conserving pseudopotential table*, https://www.pseudo-dojo.org/