Finite Difference Method applied to DML simulation. Part 1. rev 1.

ChristianV (Homeswinghome@diyAudio)

25 mar 2025

Abstract

Simulation of Distributed Mode Loudspeaker (DML) thanks to the Finite Difference Method (FDM)

Contents

Disclaimer : this paper is written in the context of DIY DML building with the target to identify some design rules to help in the panel construction. This document is not written in the context of any academic or scientific work. Its content is reviewed only by the feedback it can get while posting it in audio DIY forum like diyAudio.

Revision 1 : Introduction of the free edge boundary conditions

Minor revision (Ed. 27 Jan. 2024) :

- Improvement of the figures and the text to make clear and to show that the FDM works on points.
- Clarification of the orthotropic material case.
- Clarification of way to deal with the fictive points (see boundary conditions)
- Focus on the physical units
- Sub-paragraphs added with an introduction to damping

This document was exported from a markdown file with pandoc. Done : March 26, 2025.

1 Introduction

Searching in the different possibilities like FEM (Finite element Method) or electrical analogy I found the Finite Difference Method, which is an old technique to solve differential equations as a possible tool to go to a simulation of a DML. This method started in the late 18th with a boom in the 60th before the introduction of FEM.

The advantage and the difficulty of this method is it works closely of the equations. It will be probably not as powerful as an FEM tool can be but it might be more proper to develop the knowledge of DML and the possibility to integrate all the aspects of the problem : geometry, suspension, electromechanical exciter, etc.

Here is the comment about this method in [1] §3 :

> The finite difference method, usually referred as FDM, is a numerical method that solves the equations of boundary problems using mathematical discretization. It is the oldest but still very viable numerical methods for solution of partial differential equation and hence is suitable for solving plate bending equation. This method is sufficiently accurate for thin plate analysis (Ezeh et al., 2013). It is probably the most transparent and the most general method among the various numerical approaches available for analysis of thin plates. Although, a number of finite element based software packages are available now-a-days (2017), the finite difference method is still regarded for its straightforward nature and a minimum requirement on hardware

No need of a specific software neither, Python which is a cross-plateform language has in its standard libraries (numpy, scipy…) the functions to write a FDM script.

To go along all the process from the theory to results from a script, this introduction paper was first limited to the simply supported and clamped boundaries conditions. It was shown the results of the FDM for those boundary conditions are closed to the expect results so the free edge conditions are now included.

Among the different sources about FDM, Numerical Solution of the Poisson Equation Using Finite Difference Matrix Operators [2] inspired largely this paper.

2 Thin plate partial derivative equation (Kirchhoff-Love model)

2.1 Isotropic material

Most (not to say all?) of the papers dealing with DML at the membrane level base their approach on the thin plate theory. From Wiki Kirchhoff–Love plate theory :

> The Kirchhoff–Love theory of plates is a two-dimensional mathematical model that is used to determine the stresses and deformations in thin plates subjected to forces and moments.

For an isotropic homogeneous material, the governing equation is :

$$D(\frac{\partial^4 w}{\partial x^4} + 2\frac{\partial^4 w}{\partial x^2 \partial y^2} + \frac{\partial^4 w}{\partial y^4}) = -q(x, y, t) - \mu\frac{\partial^2 w}{\partial t^2} \tag{1}$$

Where :

- D is the bending stiffness in $N.m$
- x, y the axis of the membrane plane
- w the deflection of the plate perpendicular to its plane (following z axis) in m
- μ the areal mass in kg/m^2
- ∂ the partial derivative (meaning the derivative according to the denominator) of w
- q the load applied per unit of surface in the z direction to the plate in N/m^2

q is the known value (i.e. the force from the exciter…divided, according to the remark just before by its surface of application)

w is the unknown value (the deflection).

The other elements are collected from the plate geometry dimensions and material characteristics.

For example, from w, the speed v at each point of the membrane can be derived to compute the SPL at distance.

The eigenfrequencies (modes) and their according geometry can be also identified from this governing equations.

Some transient simulation (behavior over the time t) can also be derived from the main equation.

2.2 Units

Before going further, it is important to keep in mind that :

Each term (each element of the sum) of this equation is in N/m^2 (newton per square meter, something homogeneous to a pressure).

It is a difference from the classical form of the Newton's second law where each term is a force in newton.

2.3 Orthotropic material

In DML building, orthotropic materials (material with different stiffness according to 2 orthogonal axis) are often encountered, like the wood or the plywood. Those materials have different bending stiffness according to the directions (i.e. along or across the grain).

To cover that case, a more general governing equation is used :

$$D_x \frac{\partial^4 w}{\partial x^4} + 2H_{xy} \frac{\partial^4 w}{\partial x^2 \partial y^2} + D_y \frac{\partial^4 w}{\partial y^4} = -q(x, y, t) - \mu \frac{\partial^2 w}{\partial t^2} \tag{2}$$

- D_x is the bending stiffness in the x axis.
- D_y is the bending stiffness in the y axis.
- $H_{xy} = D_1 + 2D_{xy}$.

with :

- $D_x = \frac{h^3}{12} \frac{E_1}{1-\nu_{12}\nu_{21}}$.
- $D_y = \frac{h^3}{12} \frac{E_2}{1-\nu_{12}\nu_{21}}$.
- $D_1 = \nu_{21} D_x$.
- $D_{xy} = \frac{h^3}{12} G_{12}$.
- $\nu_{21} E_1 = \nu_{12} E_2$.

where :

- E_1 is the Young modulus along the x axis.
- E_2 is the Young modulus along the y axis.
- G_{12} is the shear modulus in the $x - y$ plane.
- ν_{12} is the Poisson ratio for transverse strain in the y direction caused by stress in the x direction.
- ν_{21} same in the second direction.

It supposes the axis x, y are the axis of symetry of the orthotropic material meaning one of those axis is the axis of the highest bending stiffness (ie. along the grain for wood), the second of the least bending stiffness (ie. across the grain).

The case of plate sides non parallel to the material axis is not covered here neither the case of non orthogonal axis.

For an isotropic material, it is simplified in :

$$D_x = D_y = D = \frac{h^3}{12} \frac{E}{1-\nu^2}, D_1 = \nu D, D_{xy} = \frac{(1-\nu)D}{2}, H_{xy} = D$$

2.4 Damping

Three types of damping are known in the materials candidate to DML. Those damping types are described in [3].

- Thermoelastic damping : it occurs in materials with a significant thermal conductivity such as metallic plate. As the materials used for DML have a low thermal conductivity, this damping is ignored here.

- Viscoelastic damping. The viscoelastic damping can be splitted in 2 models:
  - Viscous damping (or proportional) which is the most simple form with an effect independent of the frequency. It is responsible for the main damping in the plate. It will be added add hereafter in the plate equation.
  - Viscoelastic damping with a frequency dependence. It models the increase of damping with the frequency. Being more complex, this term needs a deeper study. It is kept out of this part.
- Radiation damping which is the influence of the air that surrounds the plate. This damping occurs when the wavelengths in the plate are short compare to the plate dimension, mainly above the coincidence frequency; frequency at which the speed of the waves in the plate equal the speed of the sound in the air, so in practical some kHz in DML. According to the reference document, this damping is masked by the heavy viscoelastic damping in material like wood so we will make the assumption it will be the case for material suitable for DML.In addition due to the mesh grid that can't be very small due to computation load, the FDM is limited in frequency to the highs.

So let's introduce first a viscous damping proportional to the speed of displacement of each point of the plate in the form $\mu R_f \frac{\partial w}{\partial t}$, with R_f being in s^{-1}.

Some indicative values of R_f can be found in [4] and in [5].

- Aluminum : 0.032s^{-1}
- Glass : 0.88s^{-1}
- Carbon : 0.8s^{-1}
- Wood : 2.4s^{-1}
- Guitar top : $7s^{-1}$

The goal here is not to give precise values but to have a first idea of order of magnitude.

So with that, the plate equation becomes :

$$D_x \frac{\partial^4 w}{\partial x^4} + 2H_{xy} \frac{\partial^4 w}{\partial x^2 \partial y^2} + D_y \frac{\partial^4 w}{\partial y^4} = -q(x,y,t) - \mu \frac{\partial^2 w}{\partial t^2} - \mu R_f \frac{\partial w}{\partial t}$$

3 Finite Difference Method (FDM)

3.1 Introduction to the FDM

The key idea of the Finite Difference Method (FDM) (see FDM wikipedia page) is to work on a discretized transformation (space and time) of the original continuous domain and to replace the partial derivatives by approximations based on the values of the points surrounding the point of interest. Surrounding points are according to the derivatives so either in space in the plate plan or in time.

The coefficients to apply to the neighbor points can be find for example in the wiki page "Finite difference coefficient".

For each degree of derivatives, several approximations are proposed. The usually encountered one is the approximation of degree 2.

The problem is then transformed in a linear system of equation represented in a matrix format.

The left member of the governing equation is only space dependant; the right member is time dependant.

3.2 Space discretization

The first step which is the easiest one is to mesh the domaine of the membrane. We will use the most standard and probably basic shape for the mesh : a rectangle of size Δ_x by Δ_y. Other shapes like hexagons are possible and maybe more suitable for circular membrane shape.

There is no need Δ_x and Δ_y being equal and constant over the membrane.

The plate of dimensions L_x by L_y is divided in N_x parts of Δ_x elementary length in the x direction and in N_y parts of Δ_y elementary length in the y direction.

The FDM evaluates the differential equation at the points (j, k) (j index of the x axis, k of the y axis) which are the intersections of the grid. This is a difference with FEM which works with elementary element being a surface (2D polygon) or a volume (3D).

It results from that a new working space which is a collection of $N_j = N_x + 1$ by $N_k = N_y + 1$ points with index j in $[0..N_x]$ range and k in $[0..N_y]$ range.

The points along the perimeter form the boundary. Due to the application to DML, no internal boundary (like the hole of a guitar top) is included here.

See figure 1

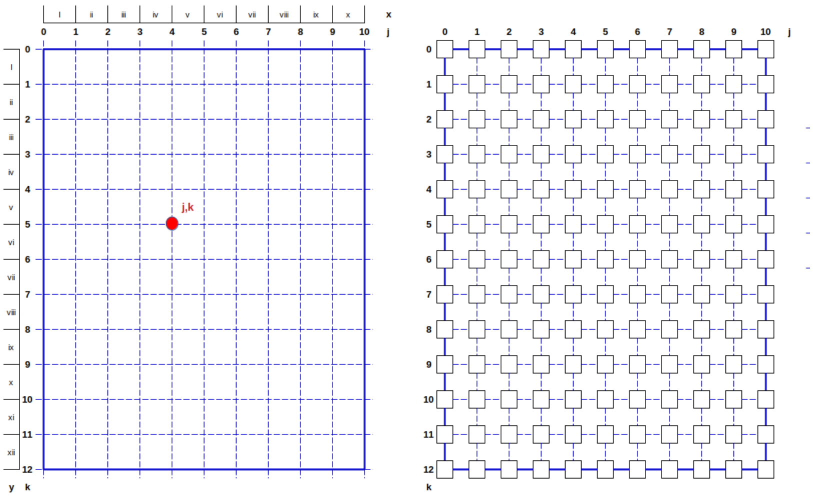


Figure 1: DML plate mesh

Note : one important thing to keep in mind is the 2D view of figure 1 is the geometrical view. To make the simulation, one equation per point will be written leading to N_jN_k equations and then a N_jN_k by N_jN_k system matrix in front so $(N_jN_k)^2$ coefficients in the matrix. Hopefully, most of them are null. This is one task of the Python script to write the system matrix.

In this idea, a tool called a stencil is used to show which and how each neighbor points are combined to be embedded in the system matrix.

The stencil for plate problem so called biharmonic operator is used for the left side of the governing equation.

The stencil shape is shown in **figure 2**.

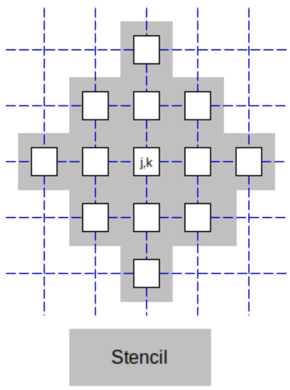


Figure 2: Biharmonic operator stencil

The grey area shows the points that are included in the calculation of the left side of the governing equation (derivatives related to the geometry) for the point of coordinates j, k.

The **figure 3** shows the stencil in place for the point of coordinates $2, 2$. All the points in the light yellow area have a stencil inside the plate geometry so their behavior is fully defined by the points of the plate.

For the points out of this area, for keeping a direct relation between the plate and its matrix representation, the choice made is to subsitute the points out of the grid by a combinaition of the deflection of the points in the plate based on the boundary condition knowledge. This create specific stencils. Those stencils change with the boundary conditions.

An other possibility is to add fictive points in the system matrix so that the biharmonic stencil can be applied to all the plate points. The deflection of the fictive points is extracted from the boundary conditions. This possibility was considered but not kept because of a missing simplification step when it comes to search for the eigenfrequencies (modes, see in the next parts).

The next figures are a more “compact” representation where a square cell figures a point. The distance between them is no more shown. **Figure 3**

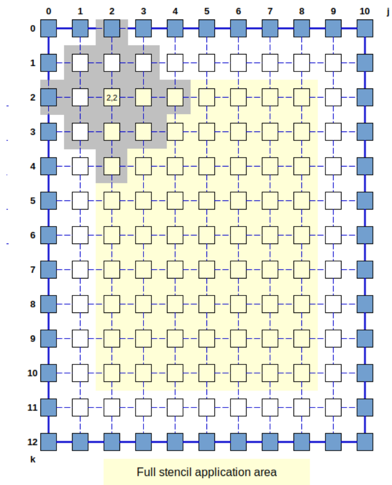


Figure 3: Stencil in 2,2

3.3 Time discretization

In the right member of the governing equation, the second order time derivative can be replaced by the finite difference of a second derivative :

$$\frac{\partial^2 w}{\partial t^2} \approx \frac{1}{{\Delta_t}^2}(w_{t+1} - 2w_t + w_{t-1})$$

The first order time derivative involved in the proportional damping is approximated by :

$$\frac{\partial w}{\partial t} \approx \frac{1}{2\Delta_t}(w_{t+1} - w_{t-1})$$

Δ_t being the sampling time (time difference between 2 samples).

w_{t-1} being the past plane displacement, w_t the current (present) plane displacement, w_{t+1} the future plane displacement. So with 2 time steps, the past and the present, it is possible to compute the next step.

With all of that, the governing equation is now :

$$[M_{Sys}][w]_t = -[q]_t - \frac{\mu}{{\Delta_t}^2}([w]_{t-1} - 2.[w]_t + [w]_{t+1}) - \frac{\mu R_f}{2\Delta_t}(w_{t+1} - w_{t-1})$$

So

$$[w]_{t+1} = (\frac{2}{R_{ft}\Delta_t}[I] - \frac{\Delta_t}{\mu R_{ft}}[M_{Sys}])[w]_t + (\frac{R_f}{2R_{ft}} - \frac{1}{R_{ft}\Delta_t})[w]_{t-1} - \frac{\Delta_t}{R_{ft}\mu}[q]_t$$

With

$$R_{ft} = \frac{R_f}{2} + \frac{1}{\Delta_t}$$

With $q = \frac{F}{\Delta_x \Delta_y}$, F being the force from the exciter

The way the $[M_{Sys}]$ is built is detailled below.

4 Boundary conditions

4.1 Boundary conditions for clamped, simply supported

Clamped, simply supported or free edge are the most common boundary conditions studied. Clamped and simply supported conditions are detailed here. The free edge conditions are added after.

- Simply supported : the edge behaves like a hinge. The plate can move freely around the edge axis but not in the z direction perpendicular to the plate.
- Clamped : the rotation around the edge is blocked like a rusty hinge.

In a mathematical wording, for the points of the edges, this is translated by :

- Clamped : $w_{k,j} = 0$, $\frac{\partial w}{\partial n} = 0$
- Simply supported : $w_{k,j} = 0$, $\frac{\partial^2 w}{\partial n^2} = 0$

n being the direction perpendicular to the edge (normal).

4.2 Finite differences at the boundaries

The first and second derivatives are as follow in finite differences :

$$\frac{\partial w}{\partial x} \approx \frac{1}{2\Delta_x}(w_{j+1} - w_{j-1})$$

$$\frac{\partial^2 w}{\partial x^2} \approx \frac{1}{{\Delta_x}^2}(w_{j+1} - 2w_j + w_{j-1})$$

Taking the example of the left edge with $j = 0$ and $w_{0,k} = 0$, the values of the fictive nodes are known:

- Clamped : $w_{-1} = w_1$
- Simply supported : $w_{-1} = -w_1$

Then replacing the w_{-1} in the generic stencil, the points just before the edge (white cells) have a specific stencil (see **figure 4**) where 1 (clamped edge) or -1 (simply supported edge) is added to the coefficient of the central cell.

With that and by setting to 1 the coefficient of each edge points, to 0 the other on the same line, in the system matrix, the boundaries are taking into account.

4.3 Boundary conditions for free edges

With free edge conditions as the displacement of the edges is not set to 0 as for clammped or simply supported conditions, the stencil has to be applied up to the edge points which leads to fictive points next to the edge but also just after.

Hopefully, there are enough specific equations at the edge to replace the stencil where the fictive points are involved by specific stencils.

This is described more in detail in the part 1b.

The **figure 5** below shows some, not all, of those specific cases.

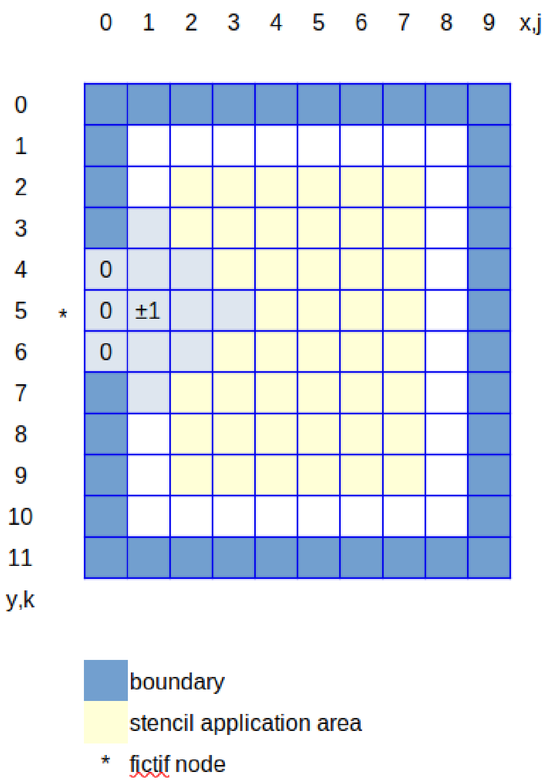


Figure 4: Edge stencil(Claped and Simply Supported edges)

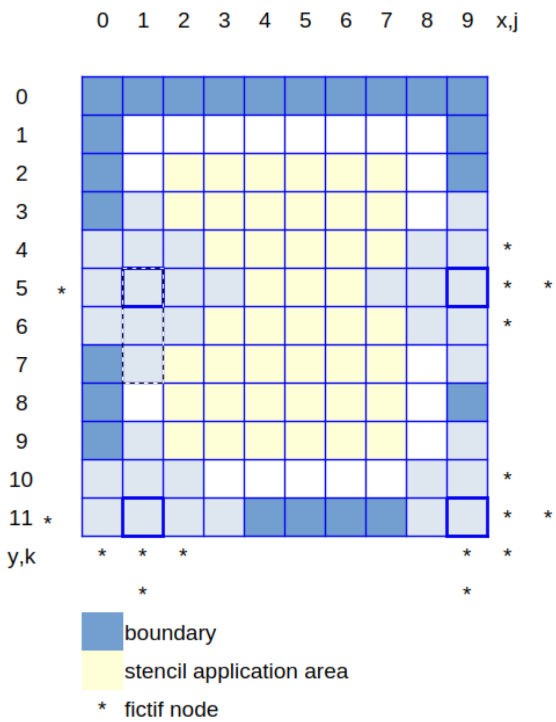


Figure 5: Edge stencil (Free edges)

5 System matrix

I borrow from M Zaman [2] the **figure 6** for the transformation from the 2D representation to the 1D one and back.

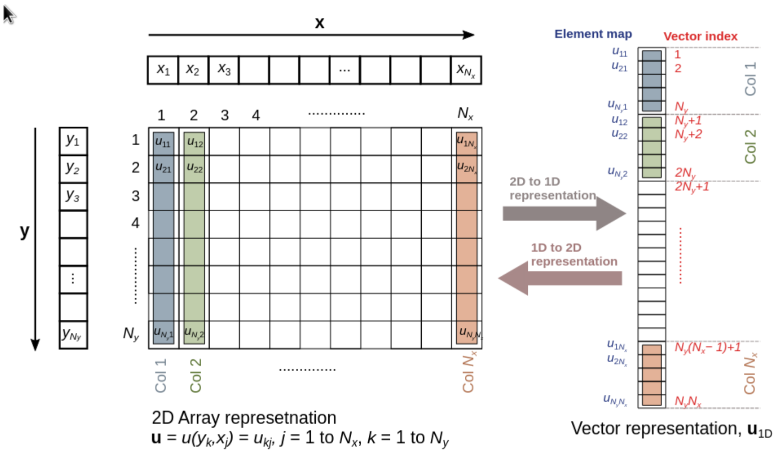


Figure 6: 2D to 1D transformation

The system matrix has to be filled as in **figure 7** (simple example of a 6 by 7 elements plate)

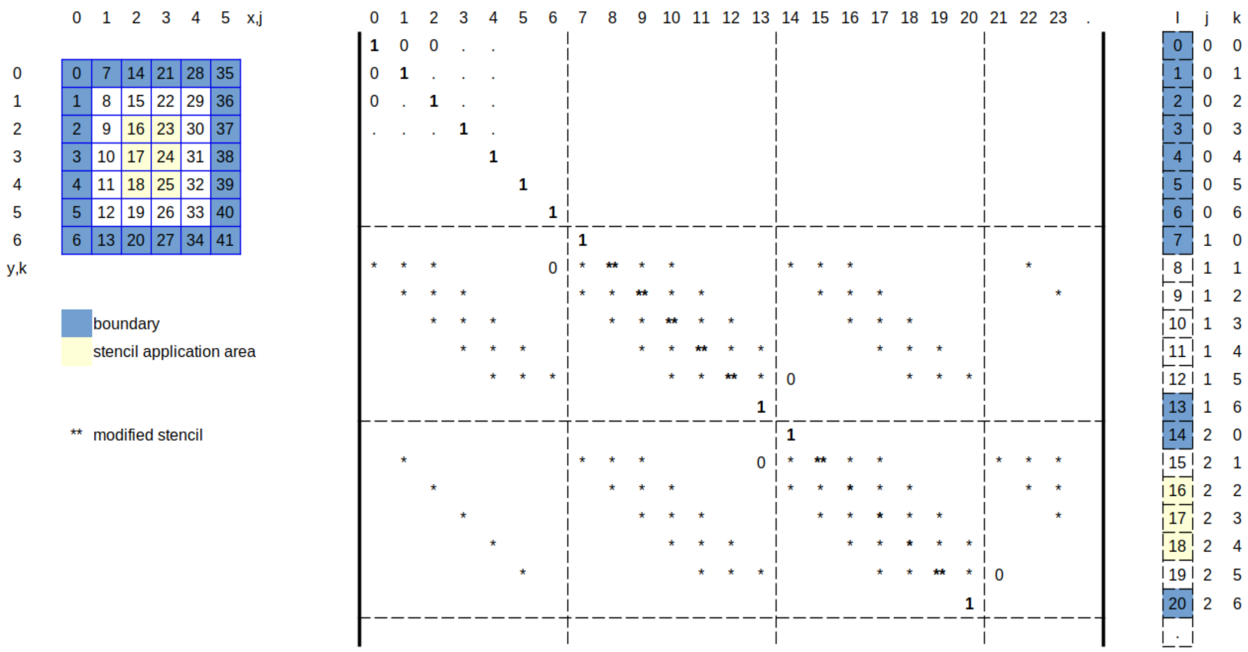


Figure 7: System matrix

5.1 System matrix filling

The system matrix is a N_jN_k by N_jN_k matrix where each point of the 2D geometric plate has one line and one column.

So each point (k, j) of the plate gets a unique index i as an identifier.

$$i = k + N_y j$$

See **figure 8**

As shown before, the stencils (linear combinaition of the neighbor points) to apply to a point i depend on its location.

y,k \ x,j	0	1	2	3	4	5	6	7	8	9
0	0	12	24	36	48	60	72	84	96	108
1	1	13	25	37	49	61	73	85	97	109
2	2	14	26	38	50	62	74	86	98	110
3	3	15	27	39	51	63	75	87	99	111
4	4	16	28	40	52	64	76	88	100	112
5	5	17	29	41	53	65	77	89	101	113
6	6	18	30	42	54	66	78	90	102	114
7	7	19	31	43	55	67	79	91	103	115
8	8	20	32	44	56	68	80	92	104	116
9	9	21	33	45	57	69	81	93	105	117
10	10	22	34	46	58	70	82	94	106	118
11	11	23	35	47	59	71	83	95	107	119

Individual point index

Figure 8: Individual point index

To fill the system matrix, each point is identified belonging to one area according to the direction (North, East, South or West) and the type of point (Corner, Edge, Inner Boundary, next to the corner on the edge, Inner point)

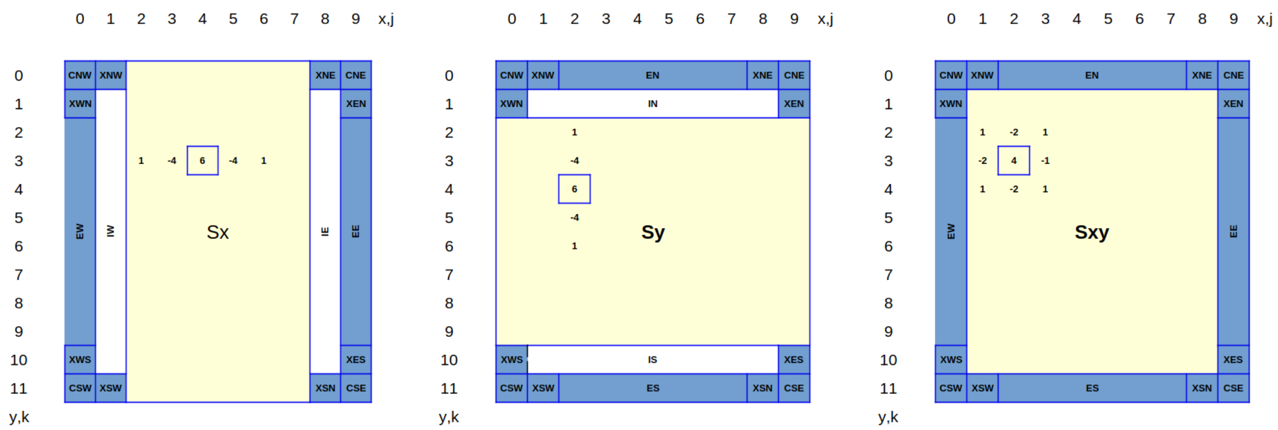
According to the area, the system matrix is filled.

To have the possibility to work with orthotropic materials and various boundary conditions, the system matrix is built as the sum of sub-matrices.

The system matrix is composed as follow :

$$M_{Sys} = \frac{D_x}{{\Delta_x}^4} A_x + 2 \frac{H}{{\Delta_x}^2 {\Delta_y}^2} A_{xy} + \frac{D_y}{{\Delta_y}^4} A_y$$

Where A_x, A_{xy}, A_y are composed according to the stencils of **figure 9 and 10**



Point type naming (sub matrix Ax) Point type naming (sub matrix Ay) Point type naming (sub matrix Axy)

Figure 9: Point type naming

For more versatility in the case the boundary conditions are limited to clamped and simply supported, A_x, A_y can be split each in 2 matrices; one A_{xi} filled with the full or truncated stencil without the "a" coefficient (see figure 8), one A_{xb} with the value "1" at the inner boundary. This A_{xb} will be added or substracted from A_{xi} according the boundaries conditions. With this option the computation effort of the system matrix is lower if it has to be done several time in the same script (ie. comparison of different boundary conditions).

In a first implementation, the matrices were filled based on loops on the k, j indexes, the nature of each points being identified. In a new implementation, a more "pythonic" approach is used based on the possibility of numpy on the indexes. The code is more compact and faster, opened to any stencils and areas.

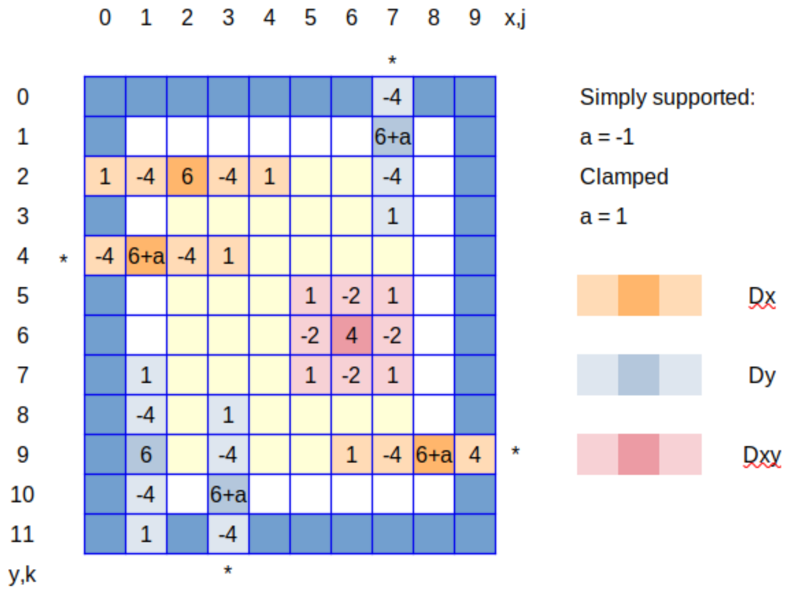


Figure 10: Dx, H, Dy (Simply supported or Clamped edges)

References

[1] H. Roknuzzaman, “Application of finite difference method for the analysis of a rectangular thin plate with eccentric opening,” Feb. 2017. Available: https://www.researchgate.net/publication/313632391_Application_of_Finite_Difference_Method_for_the_Analysis_of_a_Rectangular_Thin_Plate_with_Eccentric_Opening

[2] M. A. Zaman, “Numerical solution of the poisson equation using finite difference matrix operators,” *Electronics*, vol. 11, no. 15, p. 2365, 2022, Available: https://www.mdpi.com/2079-9292/11/15/2365#B20-electronics-11-02365

[3] A. Chaigne and C. Lambourg, “Time-domain simulation of damped impacted plates. I. Theory and experiments,” *The Journal of the Acoustical Society of America*, vol. 109, no. 4, pp. 1422–1432, 2001, Available: https://www.researchgate.net/profile/Christophe-Lambourg/publication/12011354_Time-domain_simulation_of_damped_impacted_plates_I_Theory_and_experiments/links/5450f9990cf24e8f7376af0e/Time-domain-simulation-of-damped-impacted-plates-I-Theory-and-experiments.pdf

[4] C. Lambourg, A. Chaigne, and D. Matignon, “Time-domain simulation of damped impacted plates. II. Numerical model and results,” *The Journal of the Acoustical Society of America*, vol. 109, no. 4, pp. 1433–1447, 2001, Available: https://hal.science/hal-01374311/file/Lambourg_16163.pdf

[5] G. Derveaux, A. Chaigne, P. Joly, and E. Bécache, “Time-domain simulation of a guitar: Model and method,” *The Journal of the Acoustical Society of America*, vol. 114, no. 6, pp. 3368–3383, 2003, Available: https://web.archive.org/web/20170810163750id_/http://eaton.math.rpi.edu/CSUMS/Papers/Instruments/Time%20domain%20simulation%20of%20a%20guitar%20model%20and%20method.pdf

[6] M. Barton, “Finite difference equations for the analysis of thin rectangular plates with combinations of fixed and free edges,” TEXAS UNIV AT AUSTIN DEFENSE RESEARCH LAB, 1948. Available: https://apps.dtic.mil/sti/pdfs/ADA800156.pdf

[7] I. Thompson, I. D. Abrahams, and A. N. Norris, “On the existence of flexural edge waves on thin orthotropic plates,” *The Journal of the Acoustical Society of America*, vol. 112, no. 5, pp. 1756–1765, 2002, Available: https://coewww.rutgers.edu/~norris/papers/2002_JASA_112_1756-1765.pdf