

# Vibration de la membrane d'un tambour

Nicolas Bonneel, Etienne Bresciani

22 mars 2004



# Table des matières

<b>1</b>	<b>Résolution du problème en <math>r</math></b>	<b>4</b>
1.1	Résolution de (2) . . . . .	4
1.2	Résolution de (1) . . . . .	5
1.3	Solution générale . . . . .	5
1.4	Ammortissement . . . . .	7
1.5	Un exemple . . . . .	7
<b>2</b>	<b><math>\varphi</math> dépendant de <math>r</math> et de <math>\theta</math></b>	<b>8</b>
2.1	Principe . . . . .	8
2.2	Résultats . . . . .	10
2.2.1	Fonctions propres . . . . .	10
2.2.2	Animations . . . . .	11

# Introduction

Le projet consiste en la modélisation de la membrane d'un tambour. Pour cela on étudie le déplacement en tout point  $x$  de la membrane en fonction du temps. On cherche donc une fonction  $u(x, t)$ .

Lorsque l'on tape sur un tambour, il n'y a quasiment pas de déplacements horizontaux, et ces déplacements n'influent pas beaucoup sur le son. Les composantes horizontales du déplacement seront donc considérées nulles et alors  $u = u_z e_z$ .

L'équation qui régit le déplacement, qui est donc une équation scalaire, est la suivante :

$$\frac{\partial^2 u}{\partial t^2}(x, t) - c^2 \Delta u(x, t) = 0.$$

On a aussi la condition aux limites  $u(x, t) = 0$  sur le bord du tambour.

Pour résoudre l'équation, on utilise la méthode de séparation des variables en posant

$$u(x, t) = \varphi(x)\psi(t),$$

on reporte dans l'équation qui devient

$$\psi''(t)\varphi(x) - c^2\psi(t)\Delta\varphi(x) = 0.$$

On divise par  $u(x, t) = \varphi(x)\psi(t)$ , on obtient :

$$\frac{\psi''(t)}{\psi(t)} = c^2 \frac{\Delta\varphi(x)}{\varphi(x)}.$$

Comme le membre de gauche ne dépend que de  $t$  et le membre de droite que de  $x$ , on en déduit que les deux sont constants, et donc il existe  $\lambda \in \mathfrak{R}$  tel que

$$\psi''(t) = c^2\lambda\psi(t),$$

$$\Delta\varphi(x) = \lambda\varphi(x).$$

On notera  $R$  le rayon du tambour. La résolution du problème se fera en coordonnées polaires, puisque nous travaillons sur un objet cylindrique : on aura donc  $x = x(r, \theta)$ .

Nous organiserons notre travail en deux parties : dans la première nous nous contenterons de chercher les solutions qui ne dépendent que de  $r$ , puis dans la deuxième nous développerons une méthode numérique pour trouver les solutions qui dépendent à la fois de  $r$  et de  $\theta$ .

# 1 Résolution du problème en $r$

Dans cette partie nous supposons que l'impulsion initiale sur le tambour se fait ponctuellement au centre du tambour. Il est donc raisonnable de chercher les solutions ne dépendant que de  $r$  puisque le problème est symétrique. On résoud alors le système suivant :

$$\begin{cases} \psi''(t) = c^2 \lambda \psi(t) & (1) \\ \Delta \varphi(r) = \lambda \varphi(r) & (2) \\ \varphi(r) = 0 \text{ pour } r = R & (3) \end{cases}$$

## 1.1 Résolution de (2)

On résoud d'abord (2) qui nous donnera entre autre les valeurs de  $\lambda$ . En polaires lorsque  $\varphi$  ne dépend que de  $r$  on obtient :

$$\Delta \varphi(r) = \varphi''(r) + \frac{1}{r} \varphi'(r) - \frac{\lambda}{c^2} \varphi(r)$$

et on sait de plus que  $\lambda$  est négatif donc (2) s'écrit :

$$\varphi''(r) + \frac{1}{r} \varphi'(r) + \frac{|\lambda|}{c^2} \varphi(r) = 0.$$

On pose  $s = r \frac{\sqrt{|\lambda|}}{c}$  et on réécrit l'équation en fonction de  $s$  avec

$$\begin{cases} \varphi'(r) = \frac{\partial \varphi}{\partial s} \frac{\partial s}{\partial r} = \frac{\sqrt{|\lambda|}}{c} \varphi'(s) \\ \varphi''(r) = \frac{\sqrt{|\lambda|}}{c} \frac{\partial \varphi'}{\partial s} \frac{\partial s}{\partial r} = \frac{\sqrt{|\lambda|}}{c} \frac{\sqrt{|\lambda|}}{c} \varphi''(s) = \frac{|\lambda|}{c^2} \varphi''(s) \\ \frac{1}{r} = \frac{\sqrt{|\lambda|}}{c} \frac{1}{s} \end{cases}$$

On obtient

$$\frac{|\lambda|}{c^2} \varphi''(s) + \frac{\sqrt{|\lambda|}}{c} \frac{\sqrt{|\lambda|}}{c} \frac{1}{s} \varphi'(s) + \frac{|\lambda|}{c^2} \varphi(s) = 0.$$

On peut alors simplifier par  $\frac{|\lambda|}{c^2}$  et multiplier par  $s^2$  :

$$s^2 \varphi''(s) + s \varphi'(s) + s^2 \varphi(s) = 0$$

On reconnaît alors une forme d'équation solutionnée par les fonctions de Bessel. La solution générale est alors

$$\varphi(r) = c_1 J_0\left(r \frac{\sqrt{|\lambda|}}{c}\right) + c_2 Y_0\left(r \frac{\sqrt{|\lambda|}}{c}\right)$$

où  $c_1 \in \mathfrak{R}$  et  $c_2 \in \mathfrak{R}$ ,  $J_0$  est la fonction de Bessel du premier type et  $Y_0$  la fonction de Bessel du deuxième type.

Il faut remarquer que  $Y_0$  diverge en  $r = 0$  ce qui n'est pas compatible avec le modèle physique. Donc on a nécessairement  $Y_0 = 0$  et alors

$$\varphi(r) = c_1 J_0\left(r \frac{\sqrt{|\lambda|}}{c}\right).$$

On utilise maintenant la condition limite qui est  $\varphi(R) = 0$ , ce qui signifie que l'on doit avoir  $J_0\left(R \frac{\sqrt{|\lambda|}}{c}\right) = 0$ . Or la Bessel est une fonction définie sur  $\mathfrak{R}_+$  qui oscille autour de zéro. Schéma de la Bessel  $J_0$  :

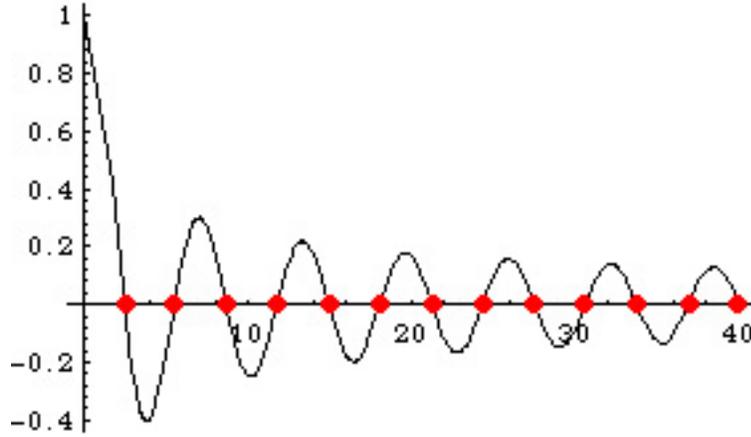


FIG. 1 – Schéma de la Bessel  $J_0$ .

On aura donc une infinité de valeurs  $\lambda_n$  satisfaisant  $J_0(R\frac{\sqrt{|\lambda_n|}}{c}) = 0$ .  
 En pratique pour trouver les  $\lambda_n$  on résoud informatiquement  $J_0(x) = 0$  et on en déduit  $\lambda_n = -(\frac{cx}{R})^2$ .

On obtient donc une infinité de solutions de la forme

$$\varphi_n(r) = c_n J_0(r \frac{\sqrt{|\lambda_n|}}{c}).$$

Ces fonctions représentent en fait les fréquences propres de vibration de la membrane du tambour puisque ce sont toutes les réponses en espace du déplacement. Comme nous étudions un problème symétrique dans cette partie, les fréquences propres sont ici toutes symétriques. Voici en illustration les première fréquences propres de vibrations calculées à la constante  $c_n$  près : (R=32)

## 1.2 Résolution de (1)

Comme on a une infinité de  $\lambda_n$  on aura une infinité de  $\psi_n$  solutions de

$$\psi''(t) - \psi'(t) = \lambda_n.$$

Les  $\lambda_n$  étant négatifs les solutions sont :

$$\psi_n(t) = a_n \cos(\sqrt{|\lambda_n|}t) + b_n \sin(\sqrt{|\lambda_n|}t).$$

Etant donné qu'il n'y a pas de condition initiale à cette équation différentielle d'ordre 2 on trouve un espace vectoriel de dimension 2 de solutions,  $a_n$  et  $b_n$  sont des constantes arbitraires.

## 1.3 Solution générale

La solution générale du déplacement est la somme de toutes les solutions élémentaires :

$$u(x, t) = \sum_{n=0}^{+\infty} \varphi_n(r) \psi_n(t) = \sum_{n=0}^{+\infty} c_n J_0(r \frac{\sqrt{|\lambda_n|}}{c}) (a_n \cos(\sqrt{|\lambda_n|}t) + b_n \sin(\sqrt{|\lambda_n|}t)).$$

Soit, en posant  $A_n = c_n a_n$  et  $B_n = c_n b_n$ ,

$$u(x, t) = \sum_{n=0}^{+\infty} \varphi_n(r) \psi_n(t) = \sum_{n=0}^{+\infty} J_0(r \frac{\sqrt{|\lambda_n|}}{c}) (A_n \cos(\sqrt{|\lambda_n|}t) + B_n \sin(\sqrt{|\lambda_n|}t)).$$

**Recherche des constantes :**

Il est important de noter que les  $J_0(r \frac{\sqrt{|\lambda_n|}}{c})$  sont orthogonaux pour le produit scalaire  $\langle \cdot, \cdot \rangle$  défini par :

$$\langle f, g \rangle = 2\pi \int_0^R r f(r) g(r) dr.$$

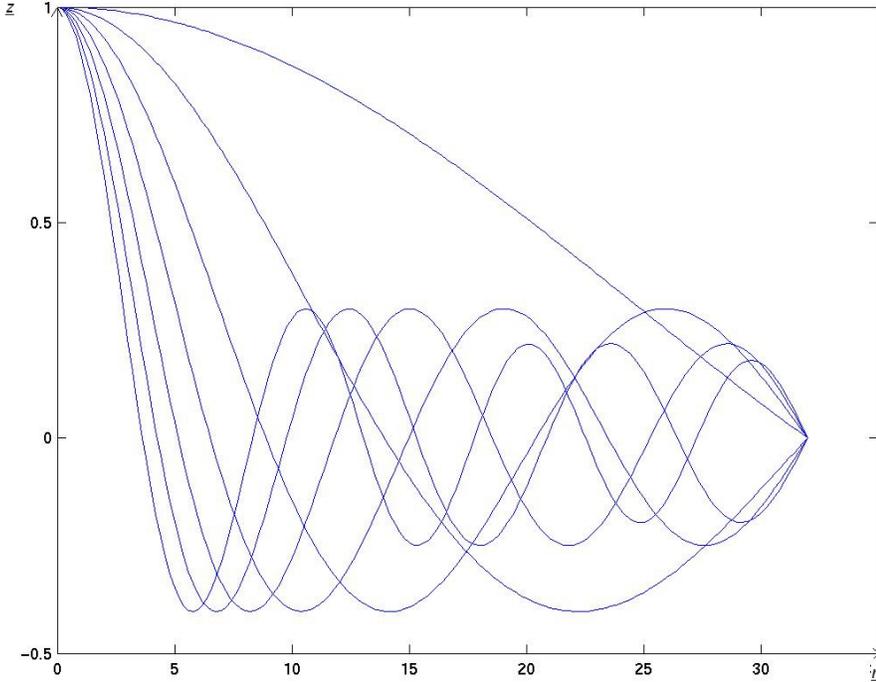


FIG. 2 – Fonctions propres de vibration de la membrane.

En effet, on a initialement  $\langle f, g \rangle = \iint fgdxdy$ . En faisant le changement de variable  $x = r \cos \theta$  et  $y = r \sin \theta$  on obtient  $\langle f, g \rangle = \iint fgrdrd\theta$ . Comme nous sommes dans le cas où rien ne dépend de theta, on a  $\langle f, g \rangle = \int_0^R rf(r)g(r)dr \int_0^{2\pi} d\theta$ , d'où le produit scalaire. On a alors  $\langle J_0(r\frac{\sqrt{|\lambda_n|}}{c}), J_0(r\frac{\sqrt{|\lambda_m|}}{c}) \rangle = 0$  pour tout  $m \neq n$  (non démontré ici).

On introduit deux conditions initiales pour déterminer les constantes : ce sont les deux conditions qui définissent le déplacement initial de la membrane, à  $t = 0$ , c'est à dire que l'on suppose  $u(x, 0) = u_0$  et  $\frac{\partial u}{\partial t}(x, 0) = u'_0$  connus.

On a :

$$u(x, 0) = \sum_{n=0}^{+\infty} A_n J_0(r\frac{\sqrt{|\lambda_n|}}{c})$$

Les  $A_n$  peuvent donc s'interpréter comme étant les coordonnées de la décomposition de  $u_0$  dans la base des  $J_0(r\frac{\sqrt{|\lambda_n|}}{c})$ .

On a alors  $\langle u(x, 0), J_0(r\frac{\sqrt{|\lambda_n|}}{c}) \rangle = A_n \langle J_0(r\frac{\sqrt{|\lambda_n|}}{c}), J_0(r\frac{\sqrt{|\lambda_n|}}{c}) \rangle$  (car les autres produits scalaires sont tous nuls), d'où :

$$A_n = \frac{\langle u_0(r), J_0(r\frac{\sqrt{|\lambda_n|}}{c}) \rangle}{\langle J_0(r\frac{\sqrt{|\lambda_n|}}{c}), J_0(r\frac{\sqrt{|\lambda_n|}}{c}) \rangle} = \frac{\int_0^R ru_0(r)J_0(r\frac{\sqrt{|\lambda_n|}}{c})dr}{\int_0^R rJ_0^2(r\frac{\sqrt{|\lambda_n|}}{c})dr}$$

Pour la condition  $\frac{\partial u}{\partial t}(x, 0) = u'_0(r)$ , on dérive  $u(x, t)$  en fonction du temps :

$$\frac{\partial u}{\partial t}(x, t) = \sum_{n=0}^{+\infty} J_0(r\frac{\sqrt{|\lambda_n|}}{c}) (-A_n \sqrt{|\lambda_n|} \sin(\sqrt{|\lambda_n|}t) + B_n \sqrt{|\lambda_n|} \cos(\sqrt{|\lambda_n|}t)),$$

ce qui donne  $u'_0(r) = \sum_{n=0}^{+\infty} B_n \sqrt{|\lambda_n|} J_0(r\frac{\sqrt{|\lambda_n|}}{c})$ .

On utilise le même raisonnement que précédemment, on obtient :

$$B_n = \frac{1}{\sqrt{|\lambda_n|}} \frac{\langle u'_0(r), J_0(r \frac{\sqrt{|\lambda_n|}}{c}) \rangle}{\langle J_0(r \frac{\sqrt{|\lambda_n|}}{c}), J_0(r \frac{\sqrt{|\lambda_n|}}{c}) \rangle} = \frac{1}{\sqrt{|\lambda_n|}} \frac{\int_0^R r u'_0(r) J_0(r \frac{\sqrt{|\lambda_n|}}{c}) dr}{\int_0^R r J_0^2(r \frac{\sqrt{|\lambda_n|}}{c}) dr}.$$

Le déplacement est donc complètement déterminé.

En pratique, nous avons calculé les constantes informatiquement, en approchant les intégrales par la méthode des rectangles.

## 1.4 Ammortissement

Jusqu'ici, nous avons basé notre étude en partant de l'équation  $\frac{\partial^2 u}{\partial t^2}(x, t) - c^2 \Delta u(x, t) = 0$ , qui ne contient pas de terme d'amortissement. Dans la réalité, les vibrations de la membrane d'un tambour diminuent au cours du temps. Il faudrait donc rajouter un terme d'amortissement dans l'équation de départ, ce que nous avons fait dans les simulations matlab. L'équation donne alors :

$$2a \frac{\partial u}{\partial t} + \frac{\partial^2 u}{\partial t^2}(x, t) - c^2 \Delta u(x, t) = 0.$$

Le changement intervient donc uniquement dans la fonction du temps  $\psi(t)$ , qui vérifie alors l'équation  $\psi''(t) + 2a\psi'(t) - \lambda\psi(t) = 0$ , dont les solutions sont

$$\psi(t) = e^{-at}(a_n \cos(\sqrt{|\lambda_n| - a^2}t) + B_n \sin(\sqrt{|\lambda_n| - a^2}t)).$$

## 1.5 Un exemple

L'exemple qui suit a été réalisé sous Matlab (document joint). La fonction initiale  $u_0$  choisie ici est  $u(x, 0) = |\sin(\frac{\pi}{2R}r)| - 1$ , qui correspond bien à une impulsion crédible respectant  $u(R) = 0$ . Voici la représentation en 3D de la membrane du tambour pour  $R = 32$  et  $t = 1s$  :

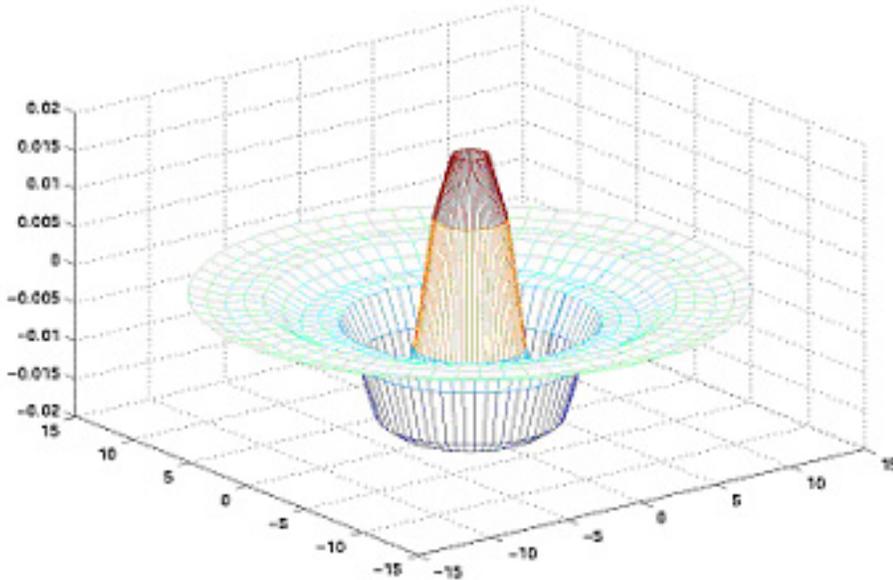


FIG. 3 – Représentation 3D

## 2 $\varphi$ dépendant de $r$ et de $\theta$

### 2.1 Principe

Nous prenons ici en compte la dépendance en  $\theta$  du déplacement vertical de la membrane du tambour.  $u$  est donc fonction de  $\theta$  et donc les  $\varphi_\lambda$  sont fonctions de  $\theta$  aussi.

En utilisant la méthode de séparation des variables, on obtient une forme de la solution suivante :

$$u(r, \theta, t) = \sum_{\lambda} \Psi_{\lambda}(t) \cdot \varphi_{\lambda}(r, \theta)$$

où  $\psi_{\lambda}(t)$  est toujours la solution de l'équation  $\frac{\psi''}{\psi} = \lambda$ , soit, comme on l'a vu en première partie :

$$\psi_{\lambda}(t) = a_{\lambda} \cdot \sin(\sqrt{-\lambda} \cdot t) + b_{\lambda} \cdot \cos(\sqrt{-\lambda} \cdot t)$$

(car  $\lambda < 0$ ) et où  $\varphi_{\lambda}$  est solution de l'équation  $c^2 \cdot \frac{\Delta \varphi}{\varphi} = \lambda$ .

Le Laplacien en coordonnées polaire, s'écrit, lorsqu'il est fonction de  $\theta$  :  $\frac{1}{r} \frac{\partial}{\partial r} r \frac{\partial}{\partial r} + \frac{1}{r^2} \frac{\partial^2}{\partial \theta^2}$  qui s'écrit encore  $\frac{\partial^2}{\partial r^2} + \frac{1}{r} \frac{\partial}{\partial r} + \frac{1}{r^2} \frac{\partial^2}{\partial \theta^2}$

On obtient donc l'équation pour  $\varphi_{\lambda}$  :

$$c^2 \cdot \left( \frac{\partial^2 \varphi}{\partial r^2} + \frac{1}{r} \frac{\partial \varphi}{\partial r} + \frac{1}{r^2} \frac{\partial^2 \varphi}{\partial \theta^2} \right) = \lambda \cdot \varphi$$

Résoudre (numériquement !) cette équation revient à trouver les valeurs propres et les fonctions propres de la matrice du système linéaire formé par le terme de gauche de cette équation en discrétisant les dérivées. En prenant les discrétisations centrées des dérivées suivantes :

$$(\varphi_{i,j} = \varphi(i \cdot \Delta r, j \cdot \Delta \theta))$$

$$\frac{\partial^2 \varphi}{\partial r^2}(r, \theta) \simeq \frac{\varphi_{i-1,j} - 2 \cdot \varphi_{i,j} + \varphi_{i+1,j}}{\Delta r^2}$$

$$\frac{\partial \varphi}{\partial r}(r, \theta) \simeq \frac{\varphi_{i+1,j} - \varphi_{i-1,j}}{2 \cdot \Delta r}$$

$$\frac{\partial^2 \varphi}{\partial \theta^2}(r, \theta) \simeq \frac{\varphi_{i,j-1} - 2 \cdot \varphi_{i,j} + \varphi_{i,j+1}}{\Delta \theta^2}$$

on obtient le système linéaire suivant :

$$c^2 \cdot \left( \frac{\varphi_{i-1,j} - 2 \cdot \varphi_{i,j} + \varphi_{i+1,j}}{\Delta r^2} + \frac{1}{r} \cdot \frac{\varphi_{i+1,j} - \varphi_{i-1,j}}{2 \cdot \Delta r} + \frac{1}{r^2} \cdot \frac{\varphi_{i,j-1} - 2 \cdot \varphi_{i,j} + \varphi_{i,j+1}}{\Delta \theta^2} \right) = \lambda \cdot \varphi_{i,j}$$

En regroupant les termes, on peut mettre sous la forme :

$$c^2 \cdot (A(r) \cdot \varphi_{i-1,j} + B(r) \cdot \varphi_{i,j-1} + C(r) \cdot \varphi_{i,j} + B(r) \cdot \varphi_{i,j+1} + D(r) \cdot \varphi_{i+1,j}) = \lambda \cdot \varphi_{i,j}$$

où :

$$A(r) = \frac{1}{\Delta r^2} - \frac{1}{2 \cdot r \cdot \Delta r} \text{ noté A}$$

$$B(r) = \frac{1}{r^2 \cdot \Delta \theta^2} \text{ noté B}$$

$$C(r) = -\frac{2}{\Delta r^2} - \frac{2}{r^2 \cdot \Delta \theta^2} \text{ noté C}$$

$$D(r) = \frac{1}{\Delta r^2} + \frac{1}{2 \cdot r \cdot \Delta r} \text{ noté D}$$

On traite le problème pour  $r=0$  indépendamment (car l'équation n'est pas définie en  $r=0$ ) : Le point  $(0,0)$  sera la moyenne des points environnants  $(\Delta r, \theta)$ , soit, en notant  $N_\theta$  et  $N_r$  respectivement le nombre de points différents dans la discretisation en  $\theta$  et en  $r$ ,

$$\varphi_{0,0} = \frac{\sum_{j=0}^{N_\theta-1} \varphi_{1,j}}{N_\theta}$$

L'équation suivante, pour  $r = \Delta r$  et  $\theta = 0$ , s'écrit (entre autre) en fonction de  $\varphi_{0,0}$  :

$$c^2 \cdot (A(\Delta r) \cdot \varphi_{0,0} + B(\Delta r) \cdot \varphi_{1,N_\theta} + C(\Delta r) \cdot \varphi_{1,0} + B(\Delta r) \cdot \varphi_{1,1} + D(\Delta r) \cdot \varphi_{2,0}) = \lambda \cdot \varphi_{i,j}$$

mais comme on n'intégrera pas le terme  $\varphi_{0,0}$  dans les inconnues puisqu'il est calculé comme la moyenne des valeurs environnantes, cette équation s'écrira :

$$c^2 \cdot \left( \left( B(\Delta r) + \frac{A(\Delta r)}{N_\theta} \right) \cdot \varphi_{1,N_\theta} + \left( C(\Delta r) + \frac{A(\Delta r)}{N_\theta} \right) \cdot \varphi_{1,0} + \left( B(\Delta r) + \frac{A(\Delta r)}{N_\theta} \right) \cdot \varphi_{1,1} + D(\Delta r) \cdot \varphi_{2,0} + \right. \\ \left. \frac{A(\Delta r)}{N_\theta} \cdot \varphi_{1,2} + \frac{A(\Delta r)}{N_\theta} \cdot \varphi_{1,3} + \frac{A(\Delta r)}{N_\theta} \cdot \varphi_{1,4} + \dots + \frac{A(\Delta r)}{N_\theta} \cdot \varphi_{1,N_\theta-1} \right) = \lambda \cdot \varphi_{1,0}$$

Les points  $(\Delta r, \theta)$  seront calculés de la même manière.

On fera attention à ce que les  $\theta$  restent dans l'intervalle  $[0, 2\pi]$  en utilisant des modulus, et on utilisera les conditions aux limites :  $\varphi_{N_r, \theta} = 0$

La matrice du terme de gauche s'écrit alors :

$$\left. \begin{array}{cccccccccccccccccccc} C + \frac{A}{N_\theta} & B + \frac{A}{N_\theta} & \frac{A}{N_\theta} & \dots & \dots & \dots & \dots & \frac{A}{N_\theta} & B + \frac{A}{N_\theta} & D & 0 & 0 & \dots & \dots & \dots & \dots & \dots & \dots \\ B + \frac{A}{N_\theta} & C + \frac{A}{N_\theta} & B + \frac{A}{N_\theta} & \frac{A}{N_\theta} & \dots & \dots & \dots & \dots & \frac{A}{N_\theta} & 0 & D & 0 & 0 & \dots & \dots & \dots & \dots & \dots \\ \frac{A}{N_\theta} & B + \frac{A}{N_\theta} & C + \frac{A}{N_\theta} & B + \frac{A}{N_\theta} & \frac{A}{N_\theta} & \dots & \dots & \dots & \frac{A}{N_\theta} & 0 & D & 0 & 0 & \dots & \dots & \dots & \dots & \dots \\ \vdots & \vdots \\ \vdots & \vdots \\ \vdots & \vdots \\ B + \frac{A}{N_\theta} & \frac{A}{N_\theta} & \dots & \dots & \dots & \dots & \dots & B + \frac{A}{N_\theta} & C + \frac{A}{N_\theta} & B + \frac{A}{N_\theta} & \dots & D \\ A & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & C & B & 0 & \dots & \dots & 0 & B & D \\ 0 & A & \dots & B & C & B & 0 & \dots & \dots & \dots & \dots \\ 0 & \dots & A & \dots & 0 & B & C & B & \dots & \dots & \dots & \dots \\ 0 & \dots & 0 & \dots \\ 0 & \dots & 0 & \dots \\ 0 & \dots & 0 & \dots \\ 0 & \dots & 0 & \dots \\ 0 & \dots & 0 & \dots \\ 0 & \dots & 0 & \dots \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dots & A & \dots & \dots & B & 0 & \dots & B \end{array} \right\}$$

avec le vecteur des inconnues sous la forme :

$$\begin{pmatrix} \varphi_{1,0} \\ \varphi_{1,1} \\ \vdots \\ \varphi_{1,N_\theta} \\ \varphi_{2,0} \\ \vdots \\ \varphi_{2,N_\theta} \\ \vdots \\ \varphi_{N_r-1,0} \\ \vdots \\ \varphi_{N_r-1,N_\theta} \end{pmatrix}$$

Ainsi, en stockant la matrice dans un programme Fortran qui se chargera de trouver les valeurs propres et les vecteurs propres, on obtiendra :

- Pour les valeurs propres : les  $\lambda$  cherchés que l'on pourra utiliser dans le calcul de  $\psi_\lambda$
- Pour les vecteurs propres : les  $\varphi_\lambda$  en certains points (les points de la discrétisation)

Pour calculer les valeurs propres et les vecteurs propres, on utilisera la bibliothèque LA\_PACK et la fonction la\_geev().

On s'assurera que les  $\lambda$  sont bien négatifs et sont bien réels (propriété de symétrie de Laplacien).

On se sert ensuite de l'orthogonalité des vecteurs propres pour décomposer une fonction  $u_0$  donnée dans la base des vecteurs propres en utilisant le produit scalaire  $\langle u_0, \varphi_\lambda \rangle = \int u_0 \cdot \varphi_\lambda * r dr d\theta$  avec  $\varphi_\lambda$  le vecteur propre sur lequel on projette le  $u_0$ . On utilisera une méthode des trapèzes pour la double intégration.  $u_0$  nous donnera ainsi les  $a_\lambda$ . On fixera les  $b_\lambda$  à 0 (mais on pourrait les calculer grâce à un  $(\frac{\partial u}{\partial t})_{t=0}$  et ce même produit scalaire). Une interface utilisateur se chargera de demander la fonction  $u_0$  désirée (par exemple :  $\cos(\theta) * \sin(\frac{r}{R} * 2\pi)$  ou  $|\cos(\theta) * \sin(\frac{r}{R} * 2\pi)| \dots$ )

Le programme Fortran se chargera ensuite de sommer les  $\Psi_\lambda(t) \cdot \varphi_\lambda(r, \theta)$  pour tous les  $\lambda$  trouvés, avec les coefficients  $a_\lambda$  et  $b_\lambda$  trouvés pour  $\Psi_\lambda(t)$ . Il stockera les  $u(r, \theta, t)$  ainsi trouvés pour différents t, dans un fichier texte.

On fera alors appel à Matlab pour l'affichage 3D de l'animation (ou des différents  $\varphi_\lambda$  selon ce que l'on veut observer...) en utilisant la fonction load('fichier.txt') de Matlab qui charge un fichier texte formaté dans une matrice 2D. On fera appel à reshape pour extraire la 3ème dimension voulue (le temps) et ainsi obtenir une hypermatrice  $M(r, \theta, t)$  pour différents r,  $\theta$  et t, puis on utilisera la fonction surf(X,Y,Z) pour afficher la surface obtenue pour différents t en l'animant avec getframe() puis movie(), en s'assurant de l'utilisation de coordonnées polaires.

## 2.2 Résultats

- Les  $\lambda$  sont bien réels et négatifs (aux erreurs numériques près donnant de très rares parties imaginaires différentes de 0, et de l'ordre de  $10^{-3}$  à  $10^{-5}$ ).

- Le temps de calcul reste raisonnable (quelques secondes) pour une animation de 81 images pour environ 200 points de discrétisation, et sont considérablement plus importants (plusieurs minutes) pour environ 900 points. Cela s'explique par le fait que le programme calcul des valeurs propres et des vecteurs propres qui n'ont pas des temps de calculs polynomiaux. Les résultats sont suffisants à partir d'une centaine de points.

### 2.2.1 Fonctions propres

Voici les 6 premières fonctions propres  $\varphi_\lambda$  de vibration de la membrane du tambour, pour 256 points de discrétisation :

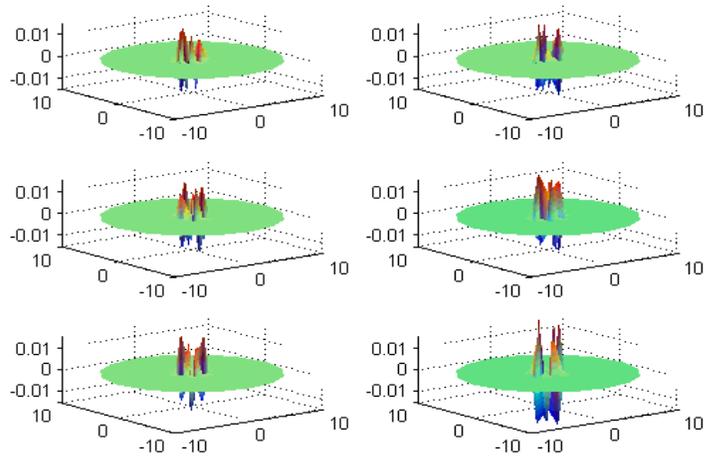


FIG. 4 – Fonctions propres

### 2.2.2 Animations

Voici deux images de la membrane du tambour au milieu de la modélisation pour 144 points de modélisation :

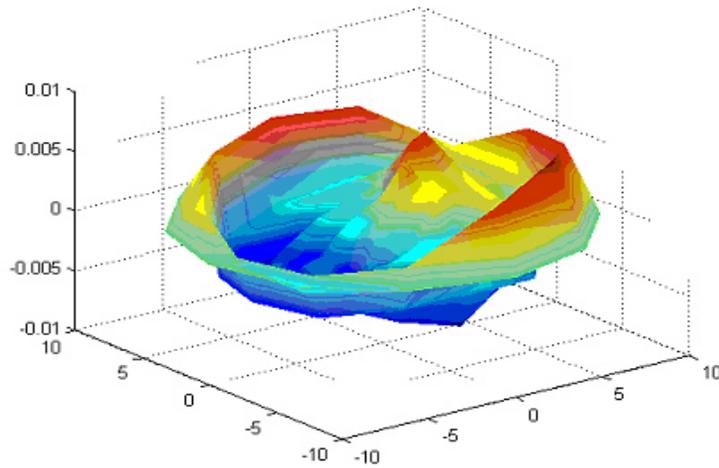


FIG. 5 – Déplacement initial :  $\cos(r)$  décentré.

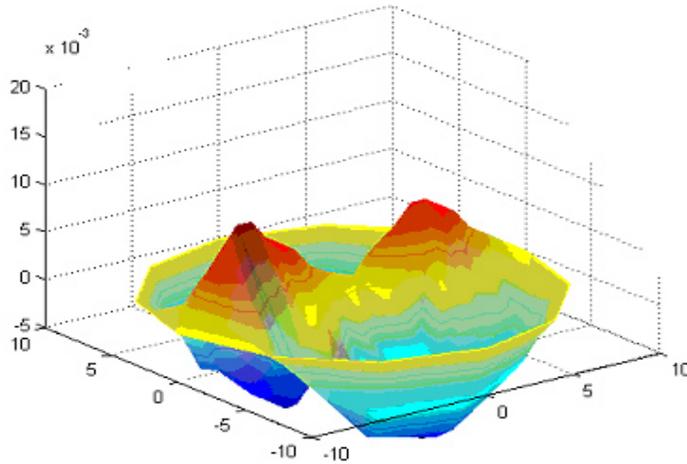


FIG. 6 – Déplacement initial :  $\sin(\theta)\sin(\frac{r}{R})$ .

## Conclusion

Nous avons développé un algorithme efficace pour étudier la vibration de la membrane d'un tambour. Nous avons ainsi pu observer les différentes harmoniques de la vibration, qui étaient toutes symétriques pour le problème ne dépendant que de  $r$ , et non pour le problème dépendant également de  $\theta$ . Un déplacement se décompose à chaque instant sur la base orthogonale formée par ces fonctions propres et s'écrit donc comme une somme pondérée de ces fonctions.

Ce projet nous a entre autre permis de manipuler Matlab sous différentes formes, notamment pour les animations etc., et nous avons pu résoudre en Fortran une équation aux dérivées partielles par une méthode de discrétisation. Nous avons ainsi pu avoir une approche de la modélisation physique et de la mécanique par des outils mathématiques, but principale de ce département !