

# Modélisation de la réfraction et dispersion des ondes mécaniques en surface d'un béton incendié

ou Les murmures du béton

Adèle Rawas-Denigot

Master 2 Modélisation, analyse numérique et calcul scientifique

Département de Mathématiques  
Nantes Université  
France  
Septembre 2024

Rapport de stage effectué à l'Université Gustave Eiffel  
Avril - Septembre 2024

Encadré par Pierrick Mora, Massina Fengal, Maximilien Lehujeur et  
Odile Abraham du laboratoire GeoEND

et par Marianne Bessemoulin de Nantes Université

## Remerciements

Je tiens tout d'abord à exprimer ma profonde gratitude à mes tuteurs de stage, Pierric Mora, Odile Abraham, Maximilien Lehujeur et Massina Fengal, dont les conseils, la confiance et le soutien m'ont permis de mener à bien ce projet. Ils m'ont offert une grande liberté et autonomie, ce qui m'a permis de me développer professionnellement tout en explorant mes problématiques de stage. J'ai pu laisser parler ma créativité autant que ma passion pour la résolution d'équations et ce fut un véritable plaisir de pouvoir éclairer certaines questions se posant sur la thèse de Massina. Je suis très heureuse d'avoir senti mon travail valorisé et j'espère sincèrement qu'il leur sera utile dans leurs recherches futures.

Je souhaite également remercier Parisa Shokouhi et son étudiant Michail Skiadopoulos pour le code sur la répartition des ellipses dont je me suis inspirée.

Je remercie les membres du laboratoire de recherche GéoEND pour leur accueil chaleureux. Mon épanouissement au sein du laboratoire et au cours de ce stage a été enrichi par les discussions et les échanges en salle café. Un merci tout particulier aux doctorants, qui ont toujours été d'une grande bienveillance. Leur douceur et leur passion pour la recherche m'ont inspirée et donné envie de poursuivre dans la voie d'une thèse.

Je souhaite aussi exprimer ma reconnaissance aux professeurs du master MACS pour l'année passée à leurs côtés. Leur enseignement et leur accompagnement ont été essentiels à la réalisation de ce projet. Je tiens à remercier tout particulièrement Marianne Bessemoulin, mon encadrante universitaire, pour son suivi et ses relectures attentives, ses commentaires constructifs et ses corrections précieuses de mes écrits. Je leur suis profondément reconnaissante pour leur soutien tout au long de l'année.

Je tiens à remercier ma famille, mes amies et amis, qui ont été présents malgré mes longues conversations sur les simulations d'ultrasons dans le béton. Leur compréhension et leur encouragement ont été essentiels durant ce projet. Je voudrais tout particulièrement remercier William. Son soutien indéfectible, son intelligence et sa présence constante m'ont donné la force et la légèreté tout au long de ce projet. Je lui suis infiniment reconnaissante pour tout ce qu'il m'apporte au quotidien.

Je dédie ce rapport à Ghassan, ancien ingénieur du génie civil, à qui j'ai beaucoup pensé pendant ce stage.

## Table des matières

<b>1</b>	<b>Introduction</b>	<b>3</b>
1.1	Un peu d'histoire	3
1.2	Le laboratoire GéoEND de l'Université Gustave Eiffel	3
1.3	Le sujet	3
1.4	L'objectif	4
<b>2</b>	<b>L'outil FEniCSx pour la résolution d'EDP par éléments finis</b>	<b>6</b>
2.1	Un exemple d'utilisation de FEniCSx pour la résolution de l'équation de Poisson	7
2.1.1	Génération et import d'un maillage modélisant le domaine $\Omega$	7
2.1.2	Définition et écriture du problème variationnel	8
2.1.3	Résolution de problème variationnel avec PETSc	9
<b>3</b>	<b>Étude de l'onde pompe pour le modèle Pompes/Sondes dans le mortier</b>	<b>11</b>
3.1	Formulation faible de l'équation de Helmholtz 2D en domaine fréquentiel avec un terme source onde de surface	12
3.2	Étude numérique	13
3.3	Etude du champ de pompes dans l'intervalle de fréquences 20 à 80 kHz	16
3.3.1	Pompes symétriques	17
3.3.2	Pompes décalées de 20 mm sur l'axe y	20
3.4	Conclusions de fin de section	21
<b>4</b>	<b>Étude de la contribution non linéaire du mortier en réponse à la propagation des ondes de Rayleigh</b>	<b>22</b>
4.1	Étude de l'équation scalaire linéaire	23
4.1.1	Formulation faible	23
4.1.2	Premiers résultats numériques	23
4.2	Système d'équations quasi-linéaire pour l'étude de la contribution non linéaire du matériau	26
4.2.1	Étude du système quasi-linéaire	26
4.2.2	Résultats numériques	27
4.3	Conclusions de fin de section	28
<b>5</b>	<b>Modélisation du squelette granulaire du béton et résolution du système quasi-linéaire sur la dalle de béton</b>	<b>29</b>
5.1	Étude granulométrique et génération d'ellipses	29
5.2	Étude de la contribution non linéaire du béton en réponse à la propagation des ondes de Rayleigh	32
5.3	Conclusions de fin de section	36
<b>6</b>	<b>Conclusions et perspectives</b>	<b>37</b>
<b>7</b>	<b>Annexes</b>	<b>38</b>
7.1	Définitions, propriétés et théorèmes vers la résolution de l'équation de Helmholtz [11]	38
7.1.1	Rappels sur les espaces $L^p$ , $W^{1,p}$ et espaces duaux	38
7.1.2	Existence et unicité d'une solution faible [14]	39
7.2	Rappels sur les éléments finis	39
7.3	Résolution numérique du système linéaire $AU=L$	40
7.4	Méthodes de Préconditionnement : LU et LU Incomplet	42
7.5	Système quasi-linéaire	43
	<b>Références</b>	<b>44</b>

# 1 Introduction

## 1.1 Un peu d'histoire

À partir du I<sup>er</sup> siècle av. J.-C., les Romains firent grand usage du béton '*opus caementicium*', qui était une maçonnerie de blocage constituée d'un mélange de mortier (chaux grasse) et de pierres [1]. Ce nouveau matériau a permis la réalisation d'architectures remarquables comme le Panthéon de Rome, des structures qui sont restées intactes durant deux millénaires sous des climats variés, dans des zones sismiques, et même en contact direct avec l'eau. Au 19<sup>e</sup> siècle, pour renforcer les constructions, le béton a été associé à des barres d'acier, créant ainsi le béton armé, une alliance qui combine les propriétés des deux matériaux. Le béton est l'un des matériaux les plus utilisés pour la construction : la production de ciment a d'ailleurs été multipliée par 4 depuis le début des années 1990 et elle était évaluée en 2019 par le CEMBUREAU (fédération européenne des industries cimentières) à 4,1 milliards de tonnes [2]. Malheureusement, l'acier étant oxydable et le béton poreux, l'air et l'eau de l'environnement peuvent pénétrer dans ses couches inférieures et, in fine, causer des dégâts importants. Par exemple la corrosion des armatures peuvent être provoquées par une réaction de carbonatation alimentée par le CO<sub>2</sub> de l'atmosphère ou encore des ions chlorure provenant d'un environnement maritime. Il existe également d'autres types de dégâts qui démarrent à coeur du béton. Par exemple, la réaction alcali-silice, une interaction entre le ciment et certains granulats, forme un gel qui peut provoquer des fissures et des gonflements. Les dégâts thermiques, comme dans le cas d'incendies, peuvent aussi entraîner une perte de résistance et des fissurations. Les ponts et les centrales nucléaires, souvent construits en béton armé, nécessitent une étude de leur solidité qui ne peut pas toujours se faire par prélèvement d'échantillons, ce qui rend les études non destructives de plus en plus importantes.

## 1.2 Le laboratoire GéoEND de l'Université Gustave Eiffel

Le laboratoire de recherche GeoEND (Laboratoire de Géophysique et Évaluation Non Destructive) est spécialisé dans le développement de méthodes d'auscultation géophysique de subsurface et de méthodes d'évaluation non destructives appliquées aux infrastructures de génie civil. Parmi les méthodes de contrôle non destructif disponibles pour le béton, l'utilisation des ondes ultrasonores est particulièrement intéressante pour évaluer les propriétés mécaniques. La vitesse et l'atténuation de ces ondes peuvent être utilisées pour déterminer les propriétés viscoélastiques ainsi que pour caractériser les propriétés microstructurales des matériaux. Cependant, les mesures précises et leurs interprétations sont difficiles en raison de la nature fortement hétérogène du béton. De nouvelles techniques d'auscultations commencent à voir le jour en utilisant des ondes de surface, comme les ondes Love et les ondes de Rayleigh ultrasonores. La particularité de l'onde de Rayleigh est d'avoir une composante verticale qui est généralement plus facile à mesurer, ce qui est très utile dans la recherche et l'évaluation d'imperfections et de dégradations des premières couches de peau d'un matériau. Une illustration des différents types d'ondes apparaît sur la figure 1.

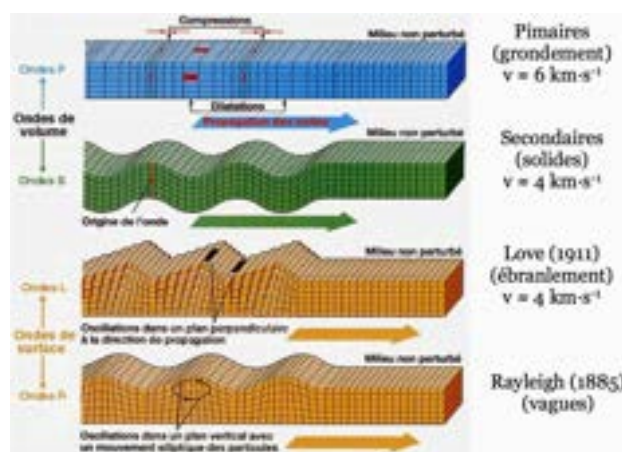


FIGURE 1 – Illustration des ondes de volume et ondes de surface (source : Wikipédia)

## 1.3 Le sujet

Le sujet de la thèse de Massina Fengal est la caractérisation de gradients de propriétés mécaniques non linéaires par ondes de surface.

Commençons tout d'abord par préciser les termes utilisés ci-dessus.

**Gradients :** Un système à gradients de propriétés mécaniques est un matériau non homogène présentant des propriétés mécaniques différentes selon les directions de ce système. Le gradient de propriétés désigne alors une variation continue d'une propriété suivant au moins une direction (dans notre cas : la profondeur). Les expériences portent ici sur deux types d'échantillons différents : le mortier (ciment + sable + eau) et le béton (mortier + granulats). Les échantillons ont subi des vieillissements artificiels et contrôlés, tels que des dégradations thermiques permettant de fournir un gradient de température maîtrisé.

**Onde de surface :** Une méthode pour caractériser le béton ou le mortier consiste à utiliser des ondes de surface (illustrées sur la figure 1). Les ondes de Rayleigh sont bien adaptées à la caractérisation de la détérioration de la surface du béton, car leur profondeur de pénétration est proche de la longueur d'onde. Ainsi, elles sont appropriées pour fournir des informations sur les premiers centimètres du béton. Un autre avantage des ondes de surface est qu'elles sont plus pratiques pour les mesures sur site que les mesures en transmission, car la plupart des structures en béton ne sont accessibles que d'un côté.

**Propriétés mécaniques non linéaires :** Les propriétés mécaniques d'un matériau sont les caractéristiques qui définissent son comportement lorsqu'il est soumis à des forces. Elles incluent, par exemple, le module d'élasticité, la résistance à la traction et la masse volumique. Ces propriétés déterminent la relation entre la contrainte appliquée et la déformation qui en résulte. Le tenseur de contraintes est une représentation matricielle décrivant l'état de contrainte d'un matériau, qui peut être linéaire ou non linéaire. Dans le cas linéaire, la contrainte est proportionnelle à la déformation, tandis que dans le cas non linéaire, la contrainte est une fonction non linéaire de la déformation. Cette non-linéarité se traduit par une réponse non linéaire de la dalle lorsqu'elle est soumise à une onde de surface, telle qu'une onde de Rayleigh, entraînant par exemple, la génération de nouveaux harmoniques (ou la dépendance de la vitesse de l'onde à la contrainte appliquée). Ces effets peuvent être décrits par le modèle de Murnaghan, qui est considéré ici. D'autres effets non-linéaires peuvent être observés dans le béton (hystérésis, dépendance à l'historique des déformations, ...) mais seront ignorés ici.

Les propriétés non linéaires sont plus sensibles que les observables linéaires, comme la vitesse ou l'atténuation, aux qualités des contacts, aux changements de microstructure et aux précontraintes. En effet, il est établi que l'acoustique non linéaire est particulièrement sensible aux changements microstructuraux induits par une réaction (ici, un endommagement thermique). Cependant, à ce jour, l'état de l'art ne permet pas encore de déterminer une distribution des propriétés non linéaires en fonction de la profondeur. C'est pourquoi nous nous intéressons à la non-linéarité du tenseur de contrainte  $\sigma$  s'écrivant comme une fonction du tenseur de déformation  $\varepsilon$  de la façon suivante :

$$\sigma = E\varepsilon(1 + \beta\varepsilon), \quad (1)$$

où  $E$  est la constante du module de Young et  $\beta$  représente le coefficient de non-linéarité du matériau, coefficient que l'on cherche déterminer.

## 1.4 L'objectif

Afin d'évaluer le coefficient  $\beta$ , deux modèles d'étude ont été mis en place. L'objectif du stage est de définir un modèle numérique représentant les modèles expérimentaux et de résoudre par éléments finis les équations de propagation des ondes de Rayleigh dans le mortier et le béton de peau grâce à la librairie FEniCSx [3].

Le premier est un modèle d'acoustoélasticité (figure 2). On considère le modèle suivant :

$$\begin{aligned} \varepsilon &= \varepsilon_0 + \varepsilon_\omega \\ &= A_0 + A_\omega \sin(\omega t - kx), \end{aligned}$$

avec  $A_0$ ,  $A_\omega$  les amplitudes respectives des deux termes de déformations, respectant  $|A_\omega| \ll |A_0|$ . On a ainsi :

$$\sigma = \sigma_0 + \sigma_\omega + O(A_\omega^2), \quad (2)$$

avec  $\sigma_0 = E\varepsilon_0(1 + \beta\varepsilon_0)$  et  $\sigma_\omega = E'\varepsilon_\omega$  si on définit  $E' = E(1 + 2\beta\varepsilon_0)$ . Du point de vue expérimental, on peut considérer  $\varepsilon_0$  comme une onde pompe, une onde ultrasonore de forte amplitude et de basse fréquence, servant à moduler le milieu. Par ailleurs, on considère une onde sonde, une onde ultrasonore de haute

fréquence et faible amplitude, pour  $\varepsilon_\omega$ . L'enjeu de ce modèle est de réussir à produire un champ de pompe le plus homogène possible. On cherchera à savoir si cette modulation sur le matériau, causée par le champ de pompes, est suffisamment homogène pour éclairer un gradient de propriété non-linéaire ou si au contraire elle pourrait affecter les résultats de l'onde sonde.

Le premier objectif est de résoudre, dans le domaine fréquentiel, l'équation de Helmholtz en considérant une loi de comportement linéaire, en deux dimensions (vue de côté de la dalle :  $x=50$  cm,  $y=12$  cm). Cette première modélisation permettra d'analyser la déformation de la dalle de mortier lorsqu'elle est soumise à une pompe ayant une fréquence de 30 kHz, dans le but d'étudier l'homogénéité du champ de pompe (pour le balayage fréquentiel les fréquences choisies varient de 20 kHz à 80 kHz).

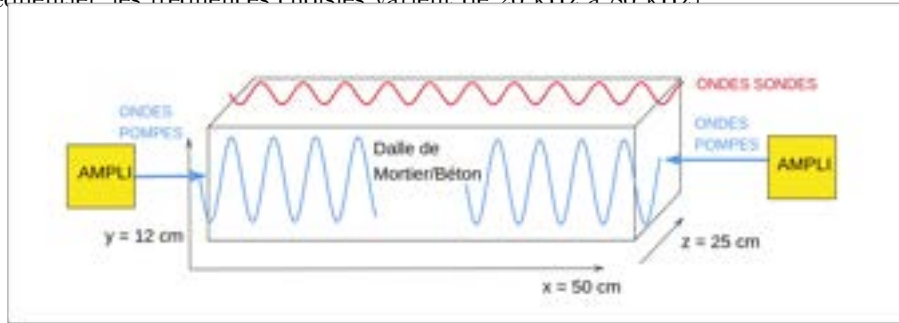


FIGURE 2 – Modèle pompe/sonde pour exciter et observer les fissures fermées

Pour le second modèle (figure 3) on considère  $\varepsilon = A \sin(\omega t - kx)$ . Dans ce cas le tenseur de contrainte devient :

$$\sigma = EA \sin(\omega t - kx) + \frac{1}{2} \beta EA - \frac{1}{2} \beta EA \cos(2\omega t - 2kx). \quad (3)$$

Cela permet de voir le signal comme la somme d'une composante harmonique fondamentale (fréquence  $\omega$ ), d'une réponse statique (fréquence nulle) et d'une seconde harmonique (fréquence  $2\omega$ ).

Le modèle de génération d'harmonique sert à connaître le comportement de l'onde de surface sur le matériau. L'objectif est de savoir si la contribution de la non-linéarité matériau est polluée par d'autres harmoniques. Le second objectif est de résoudre l'équation de dispersion des ondes en 2D et en considérant une loi de comportement non linéaire, cette fois-ci avec une vue de dessus de la dalle ( $x=50$  cm,  $z=25$  cm), en modélisant la géométrie d'un sabot (représenté par un rectangle en vue de dessus) utilisé dans l'expérience pour convertir une onde de volume en onde de surface, et ainsi diriger correctement l'onde de Rayleigh dans une direction à la surface de la dalle (figure 3). En jouant sur la loi de comportement non linéaire dans l'équation des ondes, nous nous retrouverons à résoudre un système quasi-linéaire, ce qui simplifie une partie de l'étude.

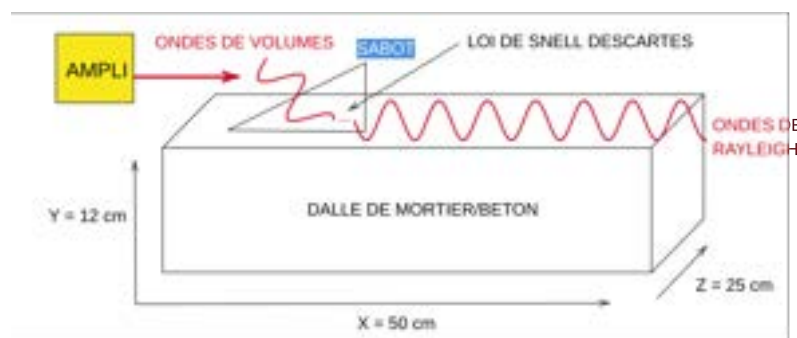


FIGURE 3 – Modèle du sabot sur la dalle, convertissant les ondes de volumes en ondes de surfaces

Dans un troisième temps, afin d'obtenir des résultats se rapprochant des études sur le béton, nous procéderons à la modélisation de granulats (représentés par des ellipses en 2D) à l'intérieur de la dalle, en respectant la courbe granulométrique utilisée pour le moulage des dalles employées lors des expérimentations. Cela nous permettra alors de résoudre les équations des expériences précédentes sur des dalles de béton.

Dans la première partie, nous introduirons brièvement l'outil FEniCSx pour la résolution d'Equations aux Dérivées Partielles (EDP) par éléments finis sur un exemple de problème de Poisson en 2D.

Les théorèmes, propriétés et définitions pour l'étude théorique des équations sont placés dans la partie 7 : Annexes.

## 2 L'outil FEniCSx pour la résolution d'EDP par éléments finis

FEniCS, désormais évolué sous le nom de FEniCSx (version 8.0) [3] est une plateforme de calcul open-source permettant de résoudre des EDP avec la méthode des éléments finis. Construite sur une base de C++ et de Python, elle est composée d'un langage spécifique permettant de définir des formes variationnelles (UFL), un compilateur de formes d'éléments finis pour générer un code d'éléments finis de bas niveau (FEniCS Form Compiler (FFC)) et un environnement de résolution de problèmes d'éléments finis automatisé ainsi qu'une bibliothèque (DOLFIN). Cette plateforme permet également un interfaçage avec des bibliothèques de hautes performances comme PETSc (Portable, Extensible Toolkit for Scientific Computation) qui permet la résolution de systèmes d'équations linéaires et non linéaires complexes. L'utilisation de FEniCSx se fait par une interface Python dans Jupyter à travers une image Docker.

The FEniCS Book, *Automated Solution of Differential Equations by the Finite Element Method*, publié en 2012 et accessible gratuitement en ligne, décrit en détail les composants logiciels de FEniCS (version 1.3) et présente un certain nombre d'applications de FEniCS à des problèmes en mécanique des fluides, mécanique des solides, électromagnétisme et géophysique.

La plateforme FEniCSx offre aussi un tutoriel interactif [5] de prise en main très efficace, dont le but est de démontrer, par des exemples simples, comment résoudre des Equation aux Dérivées Partielles par éléments finis en utilisant FEniCSx.

Une documentation [6] est également fournie sur l'ensemble des codes de développement en C++ et en Python, qu'il s'agisse des codes de surface ou des couches plus profondes, indispensables au développement de la plateforme FEniCSx.

Je me suis également beaucoup servi du logiciel comet-fenicsx [7] développé par Jeremy Bleyer, notamment pour l'import de maillages plus complexes et la définition des conditions de bords sur ces maillages. Il y introduit plusieurs exemples destinés à compléter les tutoriels existants de FEniCSx dans la documentation officielle, en se concentrant sur des applications spécifiques dans le domaine de la mécanique numérique des solides et des structures.

**DolfinX [8] : Dynamic Object-oriented Library for FINite element computation** Présentation de quelques bibliothèques très utiles :

- .io** (input/output) c'est une bibliothèque Dolfin qui facilite la gestion des données numériques en fournissant des fonctionnalités d'entrée et de sortie.
- .fem** (finite element method) contient les outils pour assembler et manipuler les formes d'éléments finis. Cela implique la construction de matrices de masse, de matrices de rigidité, de vecteurs de charge, etc., à partir des expressions mathématiques qui définissent les intégrales sur le domaine de calcul. Cela peut inclure des opérations telles que l'intégration numérique, l'évaluation de fonctions de base, et la manipulation des termes de source et de conditions aux limites.
- .mesh** Contient la création, le marquage et le raffinement du maillage.

DolfinX est donc responsable de la gestion du maillage, de l'assemblage des systèmes d'équations et de l'interfaçage avec les solveurs PETSc.

**UFL : Unified Form Language [9] :** UFL sert à définir des formulations faibles (variationnelles) des EDP de manière expressive et concise. UFL est un outil puissant et flexible qui facilite la définition et la manipulation des formulations faibles des EDP dans FEniCSx. UFL utilise une notation proche de la notation mathématique standard, ce qui rend la description des problèmes plus intuitive et plus facile à lire.

Son intégration avec d'autres composants de FEniCSx et sa syntaxe mathématique intuitive sont très utiles pour travailler avec les éléments finis.

**Ce que FEniCSx ne fait pas :** FEniCSx admet certaines limites dans son développement. Par exemple la lecture des maillages mixtes n'est pas possible. Dans un des cas étudiés Gmsh n'arrive pas à produire un maillage uniquement en quadrangles, l'étude a donc été menée en mailles triangulaires pour contourner ce problème. Une autre limite, qui n'a pas eu d'impact sur le stage, est que l'espace fonctionnel doit obligatoirement être unique. Cela empêche l'étude d'équations scalaires couplées à des équations vectorielles. Contrairement à d'autres logiciel "tout-en-un" comme Abaqus, FEniCSx ne gère pas le post-processing, ni les interfaces comme Pyvista ou Paraview. L'installation de FEniCSx n'est d'ailleurs pas évidente, c'est pourquoi nous sommes passés par une image Docker pour l'utiliser.



Maintenant, pour mieux comprendre ces concepts, nous allons présenter un exemple que nous avons traité en master 2, permettant de comparer et d'illustrer les avantages de FEniCSx dans la résolution d'un problème classique d'équation aux dérivées partielles. Cet exemple portera sur la résolution de l'équation de Poisson, que nous avons étudiée dans le cadre du projet de méthodes numériques pour les fluides incompressibles. Il met en évidence la simplicité d'utilisation et l'efficacité de FEniCSx pour formuler et résoudre un problème variationnel, tout en bénéficiant de l'interfaçage avec PETSc pour la résolution des systèmes d'équations.

## 2.1 Un exemple d'utilisation de FeniCSx pour la résolution de l'équation de Poisson

Afin d'introduire l'outil de résolution par éléments finis FenicsX, nous illustrerons une partie de ses possibilités sur un exemple étudié pendant cette année de master, et comparerons les résultats obtenus avec FenicsX et ceux obtenus dans le projet du module de Méthodes numériques pour les fluides incompressibles [13] obtenus en Fortran 90.

Soit  $\Omega$  un ouvert borné de  $\mathbb{R}^N$ ,  $N \in [1, 2, 3]$ . L'équation de Poisson s'écrit comme le problème aux limites suivant :

$$-\Delta u(x, y) = f(x, y), \quad x, y \in \Omega, \quad f \in L^2(\Omega), \quad (4)$$

$$u(x, y) = 0, \quad x, y \in \partial\Omega. \quad (5)$$

Soit  $V \subset H^1(\Omega)$  un espace de fonctions tests. Pour tout  $v \in V$ , on effectue le produit scalaire de  $L^2$  entre le problème aux limites et  $v$  sur le domaine  $\Omega$ . On obtient :

$$-\int_{\Omega} \Delta u v \, dx = \int_{\Omega} f v \, dx.$$

D'après la formule de Green appliquée sur la première intégrale, en définissant  $\cdot$  comme le produit scalaire euclidien standard et  $n$  le vecteur normal extérieur à la frontière du domaine  $\Omega$ , on obtient :

$$\int_{\Omega} \nabla u \cdot \nabla v \, dx - \int_{\partial\Omega} (\nabla u \cdot n) v \, d\sigma = \int_{\Omega} f v \, dx.$$

Ainsi les intégrales sont bien définies et on prend  $V = H_0^1(\Omega)$  pour que  $u$  et  $v$  appartiennent au même espace. Cette dernière hypothèse nous permet alors de vérifier toutes les hypothèses du théorème de Lax-Milgram et par conséquent de l'utiliser sur le problème variationnel suivant : Trouver  $u \in H_0^1(\Omega)$  tel que

$$a(u, v) = L(v), \quad \forall v \in H_0^1(\Omega), \quad (6)$$

avec  $a(u, v) = \int_{\Omega} \nabla u \cdot \nabla v \, dx$  et  $L(v) = \int_{\Omega} f v \, dx$ . Ainsi, l'équation (6) admet une unique solution et nous pouvons résoudre cette équation numériquement par éléments finis.

### 2.1.1 Génération et import d'un maillage modélisant le domaine $\Omega$

Comme mentionné précédemment, la librairie DolfinX contient la librairie .mesh qui permet de générer des maillages, triangulaires ou rectangulaires, 2D ou 3D, directement dans le programme de résolution de l'équation. Dans cet exemple, on choisit d'importer un domaine à deux dimensions carré ( $L=1.0$  m), avec des mailles triangulaires réalisé au préalable dans un programme module\_create\_model que l'on importe dans le programme principal et que l'on convertit en un maillage Dolfinx grâce à la librairie gmshio. Cette librairie permet de récupérer le domaine sur lequel nous allons résoudre l'équation ainsi que les tags des cellules et de leurs arêtes.

---

```
from module_create_model import create_model
```

```
longueur = 1.0
largeur = 1.0
taille_mailles = 1e-1
dimension_domaine = 2
```

```
model = create_model(plate_lenght=longueur,
```



---

```

    plate_height=hauteur, plate_cell_sz=taille_mailles, cell_type='triangle'
)

domain, cell_tag, facet_tag = gmshio.model_to_mesh(model,
    MPI.COMM_WORLD, rang, gdim=dimension_domaine
)

```

---

Il est également possible de visualiser le maillage grâce à pyvista (figure 4) :

---

```

import pyvista
from dolfinx import plot
pyvista.start_xvfb()
tdim = domain.topology.dim
fdim = tdim - 1
domain.topology.create_connectivity(tdim, tdim)
topology, cell_types, geometry = plot.vtk_mesh(domain, tdim)
grid = pyvista.UnstructuredGrid(topology, cell_types, geometry)
plotter = pyvista.Plotter()
plotter.add_mesh(grid, show_edges=True)
plotter.view_xy()
plotter.show()

```

---

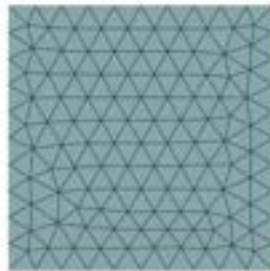


FIGURE 4 – Visualisation d'un maillage carré, aux mailles triangulaires, grâce à pyvista

### 2.1.2 Définition et écriture du problème variationnel

La bibliothèque DolfinX contient le module 'fem' qui permet de lier l'espace géométrique créé par 'gmsh' aux espaces fonctionnels et de définir l'espace des fonctions tests, en précisant la dimension de la solution (scalaire ou vectorielle), ainsi que la méthode et le degré d'éléments finis utilisés pour la résolution. Dans notre cas, nous définissons  $V$  sur notre domaine carré, et nous utilisons des éléments finis triangulaires de Lagrange de degré 1 pour la résolution, tel que :

---

```

from dolfinx.fem import functionspace
# Définition de l'espace des éléments finis de Lagrange de degré 1 pour une solution scalaire
V = fem.functionspace(domain, ("Lagrange", 1, (1,)))

```

---

À partir de cet espace, il est alors possible de créer n'importe quelle fonction :

---

```

solution_ex = fem.Function(V)
solution_ex.interpolate(lambda x: np.sin(2*np.pi*x[0])*np.sin(2*np.pi*x[1]))
f = fem.Function(V)
f.interpolate(lambda x: 8*np.pi**2*np.sin(2*np.pi*x[0])*np.sin(2*np.pi*x[1]))

```

---

qui définissent la solution exacte de l'équation, ainsi que sa fonction source  $f(x, y) = 8\pi^2 \sin(2\pi x) \sin(2\pi y)$ .

Nous définissons également les conditions de bords, ici Dirichlet homogènes, grâce aux fonctions 'fem.dirichletbc' et 'mesh.exterior\_facet\_indices' qui récupèrent les tags correspondant aux arêtes sur le bord du domaine.

---

```
from dolfinx import fem, mesh

# Localisation des facettes extérieures
boundary_facets = mesh.exterior_facet_indices(domain.topology)

# Localisation des degrés de liberté sur ces facettes
boundary_dofs = fem.locate_dofs_topological(V, fdim, boundary_facets)

# Définition de la fonction de Dirichlet u_D
uD = fem.Function(V)
uD.interpolate(lambda x: 0*x[0] + 0*x[1]) # u_D = 0

# Application de la condition de Dirichlet
bc = fem.dirichletbc(uD, boundary_dofs)
```

---

Maintenant que l'espace mathématique est créé et les conditions de bords sont définies, nous définissons la variable  $u$  qui sera notre solution, ainsi que la fonction test  $v$ , en utilisant le module 'ufl' de DolfinX :

---

```
import ufl
u = ufl.TrialFunction(V)
v = ufl.TestFunction(V)
```

---

Cette bibliothèque 'ufl' nous permet d'écrire la forme bilinéaire et la forme linéaire qui formeront le problème variationnel en question.

---

```
a = ufl.dot(ufl.grad(u), ufl.grad(v)) * ufl.dx
L = f * v * ufl.dx
```

---

FEniCSx interprète 'ufl.dx' comme étant les cellules définies dans le maillage, on peut également définir à la main un dx ou ds de la façon suivante :

---

```
dx = ufl.Measure("dx", domain=domain, subdomain_data=cell_tag)
ds = ufl.Measure("ds", domain=domain, subdomain_data=facet_tag)
# les tags sont définis directement dans le programme gmsh

# si le domaine est divisé en deux parties avec des paramètres matériaux différents
a = parametre_1 * ufl.dot(ufl.grad(u), ufl.grad(v)) * dx(1)
  + parametre_2 * ufl.dot(ufl.grad(u), ufl.grad(v)) * dx(2)

# si la fonction source ne s'applique que sur une partie de la surface du domaine
L = f * v * ds(3)
```

---

### 2.1.3 Résolution de problème variationnel avec PETSc

La bibliothèque 'dolfinx.fem.petsc' propose une classe permettant d'utiliser PETSc, un backend d'algèbre linéaire pour résoudre des systèmes linéaires ou non-linéaires. PETSc est une dépendance obligatoire de DolfinX, il est donc nécessaire d'importer explicitement l'enveloppe DolfinX pour interagir avec PETSc. L'exemple suivant montre comment définir et résoudre un problème linéaire en utilisant un solveur par Gradient Conjugué, avec un préconditionneur basé sur une décomposition LU :

---

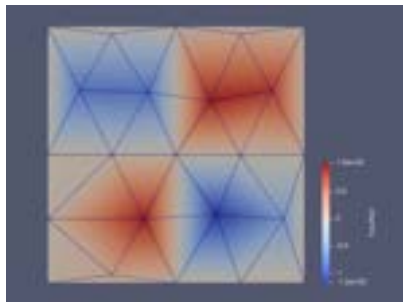
```

from dolfinx.fem.petsc import LinearProblem
problem = LinearProblem(a_, L_, bcs=[bc], petsc_options={"ksp_type": "cg", "pc_type": "lu"})
uh = problem.solve()

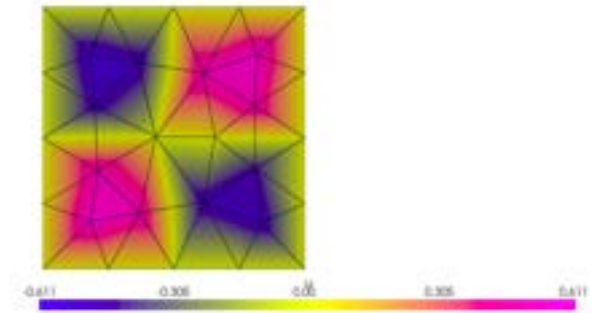
```

---

Les figures 5 et 6 illustrent les solutions obtenus avec les deux logiciels. La table 1 illustre l'efficacité du logiciel FEniCSx. En effet, pour des tailles de mailles similaires, l'erreur entre la solution exacte et la solution approchée par FenicsX est plus faible que celle trouvée avec le code Fortran pendant l'année de master. La comparaison est à prendre avec des pincettes, le code en Fortran n'est peut être pas très optimisé, mais le but est simplement de présenter l'outil FenicsX comme étant très pratique pour ces résolutions d'équations de type "fluides incompressibles".

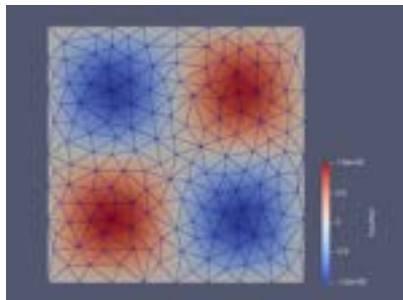


(a) Problème résolu avec une erreur  $L^2$  de  $5e-1$  avec un programme Fortran pendant l'année de master

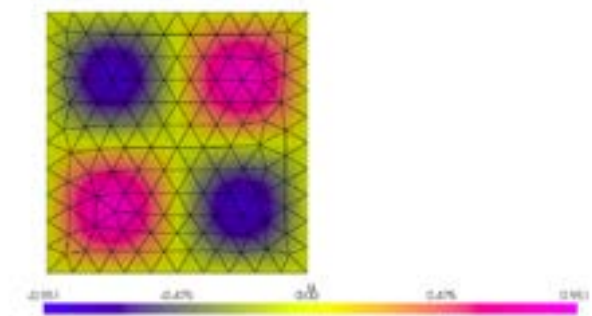


(b) Problème résolu avec une erreur  $L^2$  de  $9e-2$  avec un programme FenicsX

FIGURE 5 – Visualisation des solutions résolues (a) en Fortran et (b) en FenicsX pour des mailles de taille  $3e-1$  cm



(a) Problème résolu avec une erreur  $L^2$  de  $2e-1$  avec un programme Fortran pendant l'année de master



(b) Problème résolu avec une erreur  $L^2$  de  $2e-2$  avec un programme FenicsX

FIGURE 6 – Visualisation des solutions résolues (a) en Fortran et (b) en FenicsX pour des mailles de taille  $1e-1$  cm

Logiciel	Err mailles triangles $10^{-1}$	Err mailles triangles $10^{-2}$	Err mailles quadrangles $10^{-2}$
Fortran 90	$5.10^{-1}$	$2.10^{-1}$	-
FEniCSx	$9.10^{-2}$	$2.10^{-2}$	$10^{-5}$

TABLE 1 – Erreurs  $L2$  pour des maillages triangulaires et quadrangulaires avec des éléments finis de degré 1

### 3 Étude de l'onde pompe pour le modèle Pompes/Sondes dans le mortier

**Motivations** Les pompes sont des ondes ultrasonores de "basses" fréquences (ici entre 20 et 80 kHz) et de grandes amplitudes, elles permettent d'ouvrir par excitation les fissures fermées dans le béton. Le premier objectif de ce stage est de modéliser une onde pompe et de calculer le déplacement particulière de la dalle engendré par la propagation de l'onde. Pour commencer sur une géométrie simple, nous étudierons ce déplacement dans la dalle de mortier ; le béton sera l'objet d'étude de la partie 5.

Les sondes sont des ultrasons de fréquences plus élevées afin de ne pas être confondues avec les pompes et ont une amplitude plus petite, ce sont leur signaux qui sont étudiés et qui permettent de remonter aux caractéristiques du matériau.

Le but de cette première modélisation est l'étude de la distribution d'amplitude de l'onde en tout point en réponse à une sollicitation sinusoïdale entretenue. On fera, par la suite, un balayage en fréquence de 20 à 80 kHz, dans le but de modéliser le champ de pompe envoyé sur la dalle (figure 7). Cette méthode est fréquemment utilisée au laboratoire GéoEND, dans des contextes différents. La stratégie est de balayer un spectre assez large dans le but d'obtenir, en moyenne, une sollicitation uniforme. Dans le cadre de la thèse, la question se pose de savoir si les configurations habituellement utilisées sont efficaces pour révéler un gradient sous la surface.

On essaiera de répondre aux deux questions suivantes : Le champ de déformation généré par les ondes pompes est-il homogène ? Quels paramètres peuvent être modifiés afin d'optimiser cette homogénéité ?

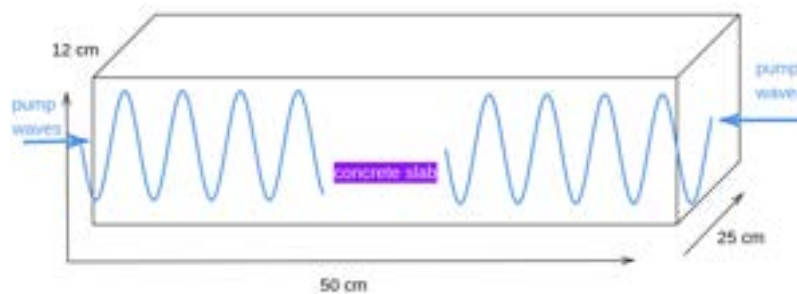


FIGURE 7 – Schéma 3D du premier modèle d'étude représentant la dalle et les deux ondes pompes

**Objectifs** La première étape est de vérifier la fiabilité du code et, pour cela, nous testerons plusieurs paramètres de résolution intégrés à la librairie PETSc de FEniCSX en regard d'études de convergence et de comparaison des temps de calcul CPU.

Dans un second temps, l'objectif est de modéliser le champ de pompe dans la dalle et d'analyser la répartition moyenne de ce champ. C'est pourquoi, après avoir résolu l'équation avec un terme source constant, nous nous intéressons au balayage fréquentiel afin d'étudier l'homogénéité du champ de pompe. Cette étude permettra de regarder si les sollicitations de la pompe sont homogènes sur toute la dalle. Le contraire pourrait poser problème car cela mélangerait les mesures expérimentales entre les effets dus à l'endommagement et ceux causés par les pompes.

**Paramètres** La dalle a les dimensions suivantes :  $L=50$  cm de longueur,  $l=25$  cm de largeur et  $h=12$  cm de hauteur, et de  $2265 \text{ kg/m}^3$  et  $E = 28.8 \text{ GPa}$  pour le mortier, ainsi qu'un coefficient de poisson  $\nu = 0.2$ . Cette première étude consiste alors à résoudre l'équation de Helmholtz en 2 dimensions ( $x \in [0, L], y \in [0, h]$ ) en domaine fréquentiel avec un terme source d'amplitude constante sur une partie du bord (ici à gauche) de la dalle. Le diamètre de la source est de 4 cm.

Les paramètres que nous pourrions faire varier seront notamment l'alignement des pompes, à gauche et à droite, par rapport à la droite (D) visible sur la figure 8, ainsi que le pas fréquentiel utilisé pour le balayage du champ et la plage de fréquences que nous prendrons ici assez large (allant de 20 à 80 kHz).



FIGURE 8 – Visualisation du maillage ainsi que l'emplacement des pompes sources sur une tranche de la dalle en deux dimensions. Les flèches rouges indiquent la direction du déplacement imposé par les sources. Les flèches grises indiquent que la position des sources peut être ajustée verticalement.

### 3.1 Formulation faible de l'équation de Helmholtz 2D en domaine fréquentiel avec un terme source onde de surface

Soit  $\Omega = [0, L] \times [0, h]$ , un domaine représentant une tranche de la dalle vue de côté.

On introduit les coefficients de Lamé : le module de compression  $\Lambda$  et le module de cisaillement  $\mu$  tels que :

$$\Lambda = \frac{E\nu}{(1+\nu)(1-2\nu)},$$

$$\mu = \frac{E}{2(1+\nu)},$$

et on note les tenseurs de déformation et de contraintes respectivement :

$$\varepsilon(u) = \text{sym}(\nabla(u)),$$

$$\sigma(\varepsilon(u)) = \Lambda \text{Tr}(\varepsilon(u))I_2 + 2\mu\varepsilon(u).$$

On notera que le tenseur de contraintes est linéaire dans cette partie.

Soit  $S \subseteq \partial\Omega_{\text{gauche}}$  où  $\partial\Omega_{\text{gauche}}$  est le bord gauche du rectangle  $\Omega$ . On se place dans le domaine fréquentiel et on pose  $\rho = \frac{\mu}{c^2}$ ,  $c$  est la célérité de l'onde et  $\omega = 2\pi f$  la fréquence angulaire. L'équation de Helmholtz est la suivante [12] :

$$\begin{cases} \text{Trouver } u = (u_x, u_y) \in \mathbf{H}^2(\Omega), \text{ tel que} \\ -\rho\omega^2 u(x) - \text{div}(\sigma(u(x))) = 0, \quad x \in \Omega, \\ \sigma \cdot n = q(x), \quad x \in \partial\Omega, \end{cases} \quad (7)$$

avec  $q = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$  sur  $S$  et nulle partout ailleurs et  $n$  le vecteur normal extérieur à la surface du domaine. On a  $q \in L^2(\Omega)^2$ .

Pour la suite, on notera  $(\cdot)$  le produit scalaire dans  $\mathbb{R}^2$ . On définit le produit tensoriel  $(:)$  entre deux matrices carrées  $A$  et  $B$  de taille  $n$  de la façon suivante  $(A : B) = \text{Tr}(A^*B) = \text{Tr}(AB^*)$  où  $\text{Tr}$  est la trace d'une matrice et  $A^*$  désigne la matrice adjointe de  $A$ .

Soit  $V$  un espace de fonctions tests tel que  $V \subset \mathbf{H}^1$ . Pour toute fonction  $v \in V$ , le produit scalaire dans  $L^2$  avec (7), qui est bien défini d'après la propriété de [régularité](#), nous donne :

$$\langle -\rho\omega^2 u - \text{div}(\sigma(u)), v \rangle_{L^2(\Omega)} = 0. \quad (8)$$

La formule de [Green](#) vectorielle dans le plan complexe, nous donne :

$$\langle -\text{div}(\sigma(u)), v \rangle_{L^2(\Omega)} = \int_{\Omega} \sigma(u) : \varepsilon(v) \, dx - \int_{\partial\Omega} \sigma(u) \cdot n v \, ds,$$

ou encore :

$$\langle -\text{div}(\sigma(u)), v \rangle_{L^2(\Omega)} = \int_{\Omega} \sigma(u) : \varepsilon(v) \, dx - \int_S (q \cdot v) \, ds.$$

Cette égalité couplée à (8) nous permettent d'écrire la suivante :

$$\int_{\Omega} \sigma(u) : \varepsilon(v) dx - \rho\omega^2 \int_{\Omega} u \cdot v dx = \int_S (q \cdot v) ds.$$

Le problème variationnel peut alors être écrit et il est de la forme suivante :  
Trouver  $u \in V = H^1(\Omega)$  tel que

$$a(u, v) = L(v) \text{ pour tout } v \in H^1(\Omega) \quad (9)$$

avec

$$a(u, v) = \int_{\Omega} (\sigma(u) : \varepsilon(v)) dx - \rho\omega^2 \int_{\Omega} (u \cdot v) dx,$$

$$L(v) = \int_S (q \cdot v) ds.$$

### 3.2 Étude numérique

L'équation n'a pas de solution analytique simple liée à cette étude, donc pour comparer les différentes méthodes de résolution, on choisit de résoudre l'équation sur un maillage de taille  $\lambda_S/200$  (correspondant à un pas de 0.5 mm), où  $\lambda_S$  est la longueur d'onde de l'onde de cisaillement dépendant de la fréquence de la source de la façon suivante :  $c = \lambda_S f$  avec  $c$  la vitesse de propagation de l'onde. On considère un maillage avec des mailles formées par des quadrangles et des éléments finis de Lagrange de degré 2 (définition 9). On obtient alors ce premier résultat (figure 9) de déplacement particulière de la section du bloc de mortier que l'on utilisera comme solution de référence par la suite.

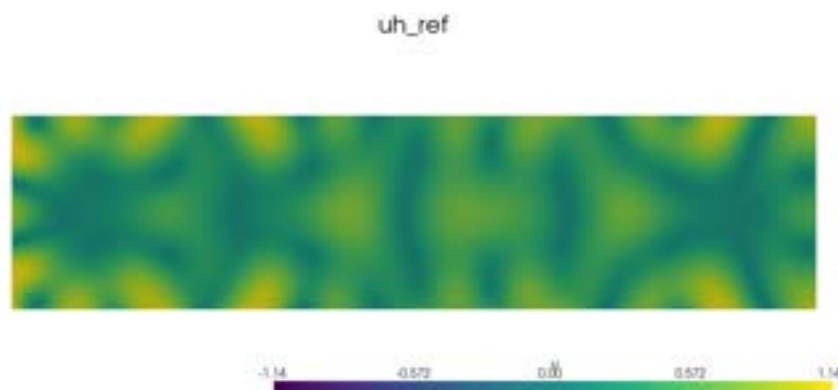


FIGURE 9 – Solution de référence du déplacement particulière sur la dalle de mortier en 2 dimensions pour une fréquence de 30 KHz

Une fois calculée, on peut exporter puis importer la solution de référence grâce à la librairie *adios4dofinix*, comme dans le code ci-dessous :

---

```
from pathlib import Path
import adios4dofinix

# nom du fichier qui contiendra la fonction solution
filename = Path("save_function_cglsu.bp")
function = uh_ref

# (export) écriture du maillage associé à la solution
adios4dofinix.write_mesh(filename, function.function_space.mesh, engine='BP4')
# écriture de la solution
adios4dofinix.write_function(filename, function, engine='BP4', time=1.0)
```

---

```

# dans un autre programme
from dolfinx.cpp.mesh import GhostMode
u_ref = function

# (import) lecture du maillage dans un autre programme
in_mesh = adios4dolfinx.read_mesh(
    filename, MPI.COMM_WORLD, engine='BP4', ghost_mode=GhostMode.shared_facet)

# attention : le maillage lu n'est pas forcément celui de départ (ordre des noeuds!=)
# on redéfinit l'espace en fonction
V_in = functionspace(in_mesh, u_ref.ufl_element())
u_in = Function(V_in)
# lecture de la solution dans un autre programme
adios4dolfinx.read_function(filename, u_in, engine='BP4', time=1.0)

```

---

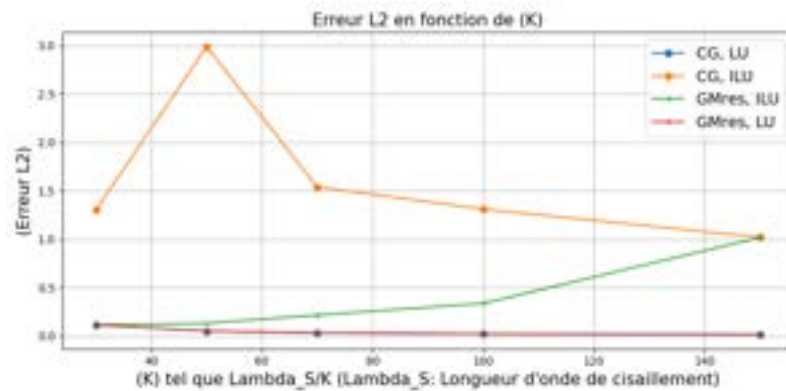
Après un calcul d'interpolation des solutions approchées par différentes méthodes : le gradient conjugué (CG) et le résidu minimal généralisé (GMres), couplés aux préconditionneurs LU ou LU Incomplet (ILU) dans une boucle sur le pas d'espace, on peut observer les courbes d'erreurs (sur la figure 10) entre les solutions calculées dans la boucle et la solution de référence, en norme  $L^2$  sur le graphique de la figure 10a. Après avoir comparé ces deux solveurs et préconditionneurs (7.4), l'étude de convergence montre que le solveur (CG, LU) est le meilleur. La figure 10b nous donne l'erreur en échelle logarithmique pour la méthode de résolution du couple Gradient Conjugué, préconditionnement LU et on observe un ordre de convergence égal à 1.

En fait, on remarque que la réelle différence entre les méthodes de résolution du système est le préconditionnement (7.4). On constate que le préconditionnement ILU n'est pas adapté à notre problème.

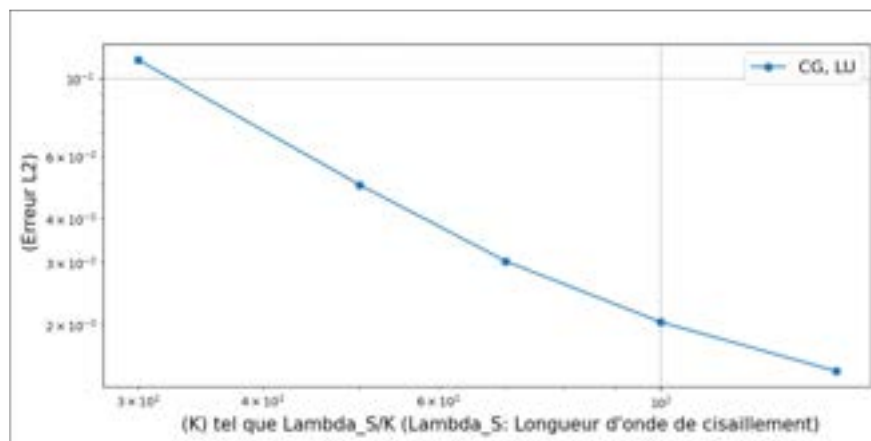
On observe également la courbe de temps de calcul CPU en fonction du nombre de mailles par longueur d'onde de cisaillement  $\lambda_S$  (figure 11).

L'erreur entre la solution faible et la solution de la formulation du problème aux limites est représentée sur la figure 12. Elle est presque nulle, sauf sur la surface où se situe la source.





(a) Erreurs en norme L2 en fonction du pas du maillage (les courbes bleue (CG, LU) et rouge (GMRES, LU) sont confondues)

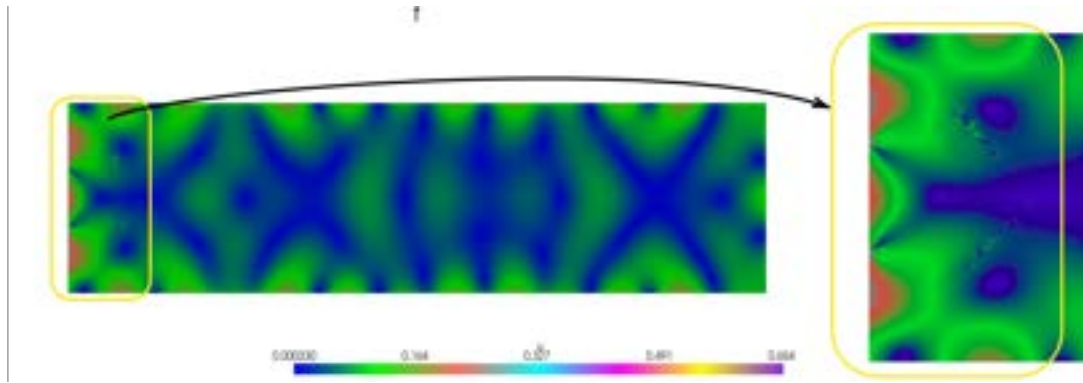


(b) Étude de convergence en échelle logarithmique de la méthode de résolution en Gradient Conjugué avec un préconditionnement LU

FIGURE 10 – Etude numérique des principales méthodes de résolution fournies par PetsC



FIGURE 11 – Temps de résolution CPU en fonction du pas du maillage (les courbes bleue (CG, LU) et rouge (GMRES, LU) sont confondues)

FIGURE 12 – Erreur de la solution calculée sur maillage grossier (pas de maille =  $\lambda_S/20 = 5\text{mm}$ )

### 3.3 Etude du champ de pompes dans l'intervalle de fréquences 20 à 80 kHz

Afin de modéliser le champ de pompe de l'expérience, on se place de nouveau sur la même face de la dalle de mortier et on envoie cette fois-ci deux sources des deux cotés de la plaque. En incluant ensuite le programme de résolution par éléments finis dans une boucle fréquentielle allant de 20 kHz à 80 kHz avec un pas de 100 Hz, afin de simuler le champ de pompe et pour éviter les singularités, nous ajoutons une petite partie imaginaire ( $5.10^{-4}i$ ) à la fréquence angulaire. On obtient la solution de déplacement  $u_h(f)$ , sur la figure 13, évaluée sur le point milieu de la dalle ( $x=25\text{ cm}$ ,  $y=6\text{ cm}$ ), sur le point de coordonnées ( $x=29\text{ cm}$ ,  $y=9\text{ cm}$ ) et sur le point presque au bord de la dalle de coordonnées ( $x=31\text{ cm}$ ,  $y=11.8\text{ cm}$ ). On observe de nombreux modes d'excitations (chaque pic correspond à un mode).

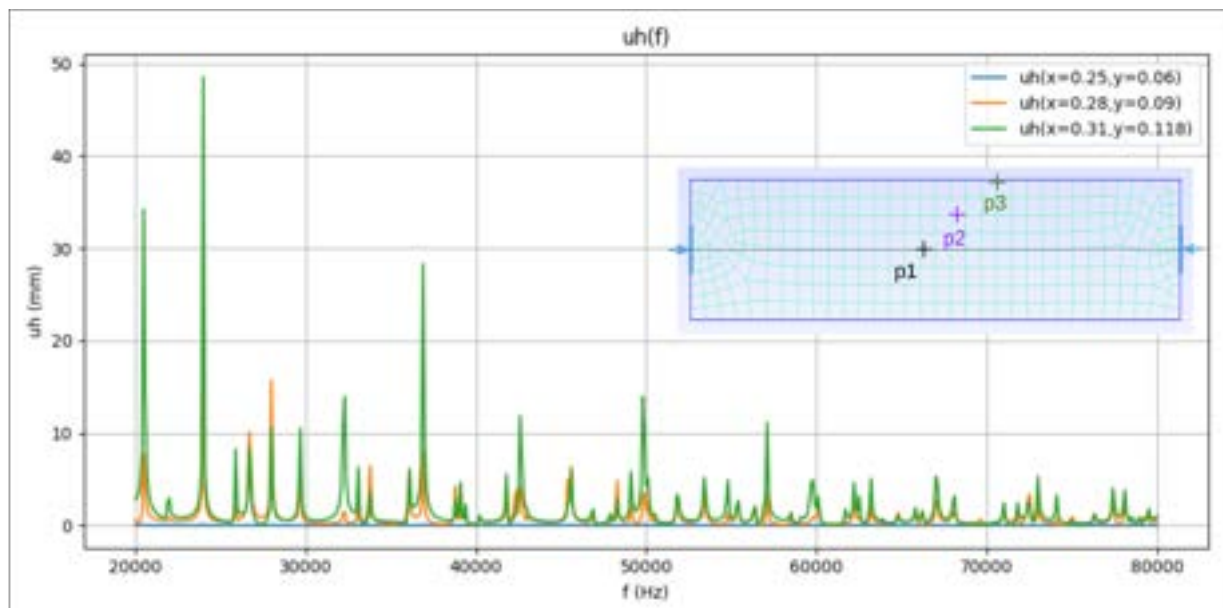


FIGURE 13 – Déplacements particuliers de trois points (p1, p2, p3) de la dalle de mortier pour des fréquences allant de 20 à 80 kHz avec un pas de 100 Hz

Le but de cette étude est de comparer les résultats numériques avec ceux expérimentaux. En effet, durant l'expérience et afin de mesurer l'énergie du champ de pompes sur les échantillons de béton, un grand nombre de mesures sont relevées puis moyennées. Suivant la position des pompes, l'énergie du champ de pompes ainsi que la vitesse observées sont plus ou moins homogènes.

Numériquement, afin de visualiser l'énergie du champ de pompes, on intègre sur un domaine de fréquences, allant de 20 kHz à 80 kHz avec un pas de 100 Hz, les solutions du déplacement particulières obtenues après une résolution de l'équation de Helmholtz (7) par éléments finis de Lagrange de degré 1 (définition 9).

La formule utilisée pour définir la moyenne du déplacement, de la vitesse ou encore de la déformation sur

les fréquences est la suivante :

$$\bar{u} = \frac{1}{\Delta f} \sqrt{\int_f |u(f)|^2 df}, \text{ où } u \text{ est la solution de l'équation de Helmholtz.} \quad (10)$$

### 3.3.1 Pompes symétriques

La modulation des propriétés induite par l'onde pompe est proportionnelle à la déformation ( $\varepsilon = \varepsilon_0$ ) ainsi, pour vérifier l'homogénéité de l'onde pompe appliquée sur la dalle on s'intéresse à la déformation que l'on calcule de la façon suivante :

$$\text{Tr}(\varepsilon(u)) = \text{Tr}(\text{sym}(\nabla(u))),$$

et à la vitesse :

$$v = i\omega u.$$

On calcule ensuite l'énergie de ces deux fonctions, déjà moyennées sur la longueur, en appliquant la formule (10) dans le but de calculer leur gradient ensuite.

Dans le cas de deux pompes mises face à face (même ordonnée  $y$ ), on peut observer la vitesse particulière et sa moyenne (figure 14) ainsi que la déformation définie par la trace du tenseur de déformation ( $\varepsilon$  défini dans la partie 3.2) et sa moyenne sur l'axe des  $x$  (figure 15).

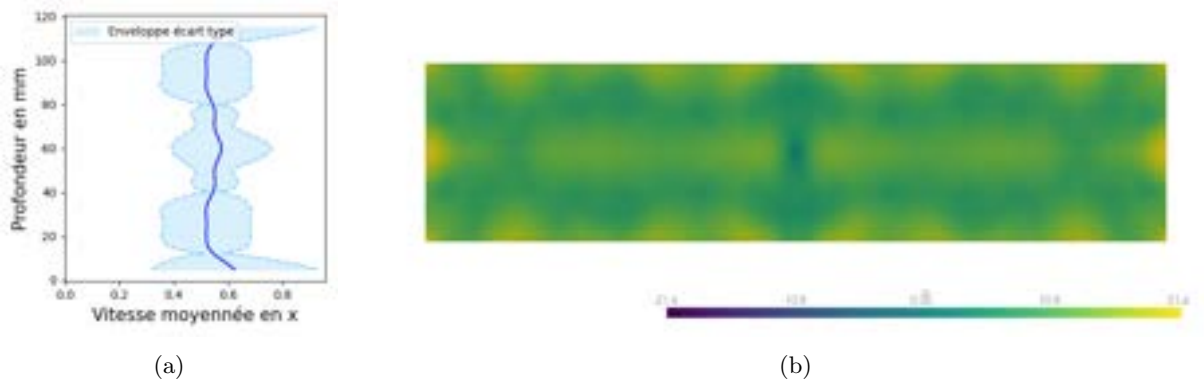


FIGURE 14 – Vitesse : (a) Moyenne de la vitesse particulière (mm/μs) moyennée sur la longueur et exprimée en fonction de la largeur (axe  $y$ ) pour des pompes symétriques et (b) Champ moyen de la Vitesse particulière (mm/μs) pour un balayage fréquentiel de 20 à 80 kHz avec un pas de 100 Hz et des pompes symétriques

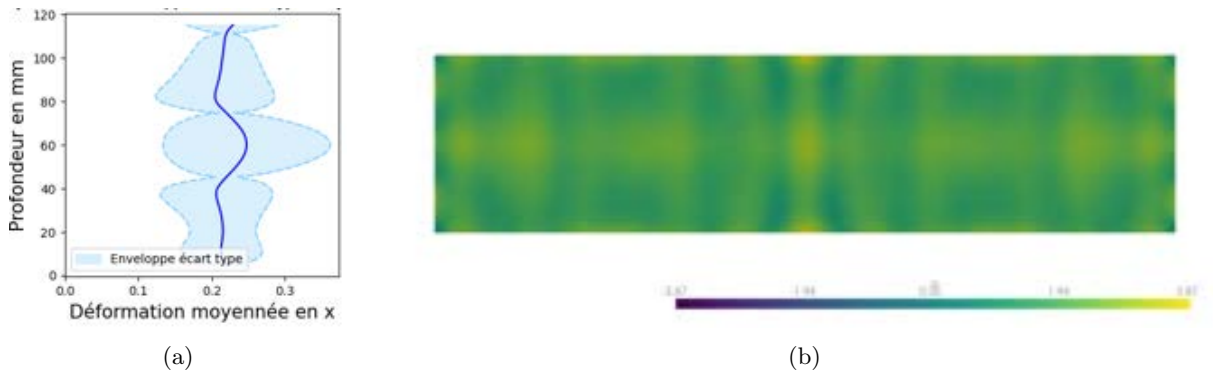


FIGURE 15 – Déformation : (a) Moyenne de la déformation  $\text{Tr}(\varepsilon)$  moyennée sur l'axe des  $x$  et tracée en fonction de  $y$  pour des pompes symétriques et (b) Champ moyen de la déformation  $\text{Tr}(\varepsilon)$  pour un balayage fréquentiel de 20 à 80 kHz avec un pas de 100 Hz et des pompes symétriques

Dans ce premier cas où les pompes sont symétriques, pour la déformation, la moyenne de la variance en fonction de la profondeur est de 9.2 % et pour la vitesse la moyenne de la variance est de 7.1 %.

On introduit ci-dessous une partie du code qui résout l'équation de Helmholtz, gère le moyennage des fonctions sur la longueur de la dalle et qui récupère la moyenne  $L^2$  du déplacement, de la vitesse et de la déformation. Il est lui même inclu dans une boucle *For* balayant les fréquences avec un pas de 100 kHz. L'outil *ParallelEvaluator* est utilisé pour évaluer les fonctions sur tous les points  $x$  du maillage et permet ainsi un moyennage des fonctions sur la longueur de la dalle, il a été développé avant le début du stage par Pierrick Mora.

ds(6) et ds(7) correspondent aux parties gauche et droite de la surface de la dalle, distinguées et tagées dans le module de création du maillage, lieux d'emplacement des deux sources de l'onde pompe.

---

```

a_1 = inner(sigma(u1), epsilon(v1)) * dx - rho * omega_f**2 * inner(u1, v1) * dx
L_1 = inner(f_1, v1) * ds(6) - inner(f_1, v1) * ds(7)

problem_cg = LinearProblem(a_1, L_1, bcs=[], \
    petsc_options={"ksp_type": "cg", "pc_type": "lu"})

uh_f = problem_cg.solve()

eps = tr(sym(grad(uh_f)))
eps_exp = fem.Expression(eps, V0.element.interpolation_points())
eps_fonction = fem.Function(V0)
eps_fonction.interpolate(eps_exp)

vh_moy_x = []
eps_moy_x = []
for x_list in x_out:

    points_out = np.array([x_list,
                           y_out,
                           z_out])

    # Déclarer un ParallelEvaluator sur le maillage
    paraEval = ParallelEvaluator(domain1, points_out)
    # Points d'évaluation
    uh_f_local = uh_f.eval(paraEval.points_local, paraEval.cells_local)
    eps_local = eps_fonction.eval(paraEval.points_local, paraEval.cells_local)

    # Rassembler les données vers le processus maître
    uh_f_global = paraEval.gather(uh_f_local, root=0)
    eps_global = paraEval.gather(eps_local, root=0)

    if domain1.comm.rank == 0:
        u_.append(uh_f_global)
        eps_moy_x.append(eps_global)
        vh_moy_x.append(1j*omega_f*uh_f_global)

eps_moy_x = np.sqrt(np.mean(np.abs(eps_moy_x)**2, axis=0))
vh_moy_x = np.sqrt(np.mean(np.linalg.norm(vh_moy_x, axis=-1)**2, axis=0))
eps_moyenne.append(eps_moy_x)
vitesse_moyenne.append(vh_moy_x)

```

---

Nous pouvons également observer les résultats obtenus lors de l'expérimentation. Le dispositif expérimental mesurant la moyenne des champs de vitesse sur la dalle est représenté sur la figure 16. On y voit le vibromètre laser permettant d'établir une moyenne de la vitesse sortante des particules en fonction de la profondeur. Néanmoins, comme on peut le voir, cette moyenne est faite en trois dimensions avec des transducteurs espacés d'une distance fixe (quelques centimètres) alors que le moyennage effectué numériquement est un moyennage sur deux dimensions, avec un pas plus petit que celui des expériences. En 3D, on mesure une vitesse qui est alors une composante hors plan qui n'est pas modélisée ici mais qui ressemble à l'effet Poisson qui est proportionnel à la déformation. Les valeurs expérimentales sont tracées en fonction de la profondeur de la dalle sur la figure 17a et mises en parallèle avec l'énergie de la déformation moyenne sur la figure 17b.

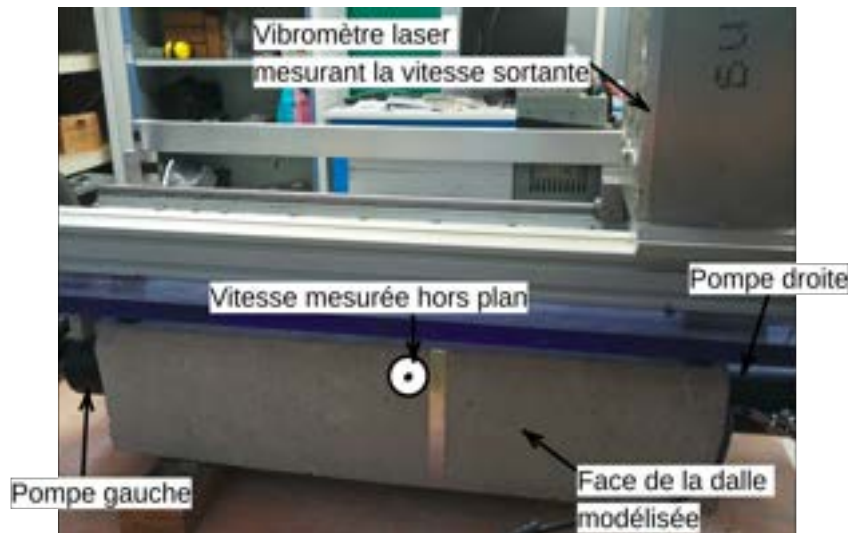
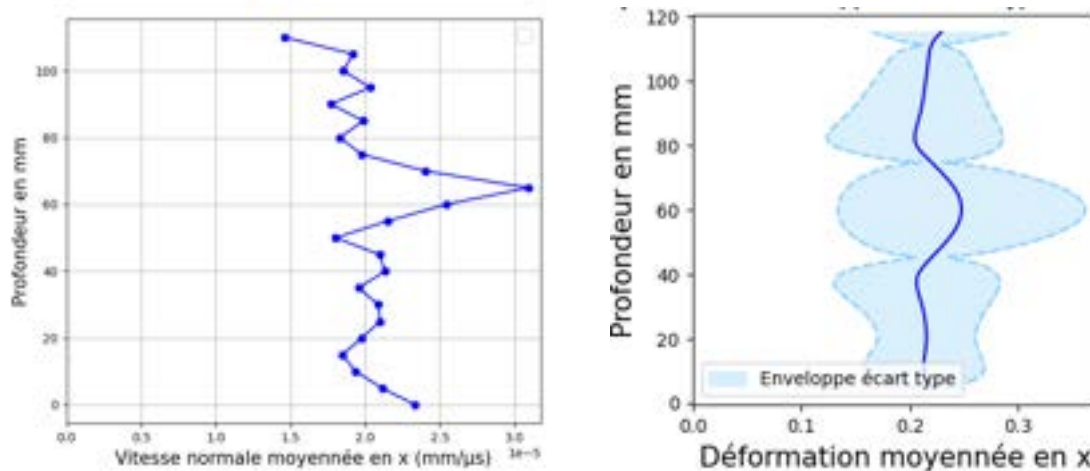


FIGURE 16 – Expérience menée par Massina Fengal dans le cadre de sa thèse (en cours) mesurant la vitesse normale à la dalle modulée par un champ d'ondes pompes



(a) Vitesse particulière (mm/s) moyennée expérimentalement pour des pompes symétriques

(b) Moyenne de la déformation  $\text{Tr}(\varepsilon)$  moyennée sur l'axe des x et tracée en fonction de y pour des pompes symétriques

Dans la prochaine sous partie, afin de se rapprocher d'un gradient de déformation plus homogène, nous allons jouer sur le paramètre de position des sources des ondes pompes et les décaler chacune dans un sens différent par rapport à l'axe de symétrie (D) de la figure 8.

### 3.3.2 Pompes décalées de 20 mm sur l'axe y

Dans le programme, nous pouvons demander à la fonction générant le domaine de décaler les sources des ondes pompes d'une certaine distance (ici 2 cm entre chaque pompes ou de 1 cm par rapport à l'axe de la droite (D)) (voir figure 8). Ainsi, après une résolution de l'équation par éléments finis de Lagrange de degré 1, on peut tracer les fonctions découlant de la solution. Sur la figure 18, on observe le déplacement particulaire évalué en 3 points de la dalle, tracé en fonction du champ de fréquences allant de 20 à 80 kHz avec un pas de 100 Hz. On remarque plus de modes d'excitations, ce qui est dû à la géométrie moins symétrique qui entraîne alors un excitation des modes anti-symétriques en plus des modes symétriques.

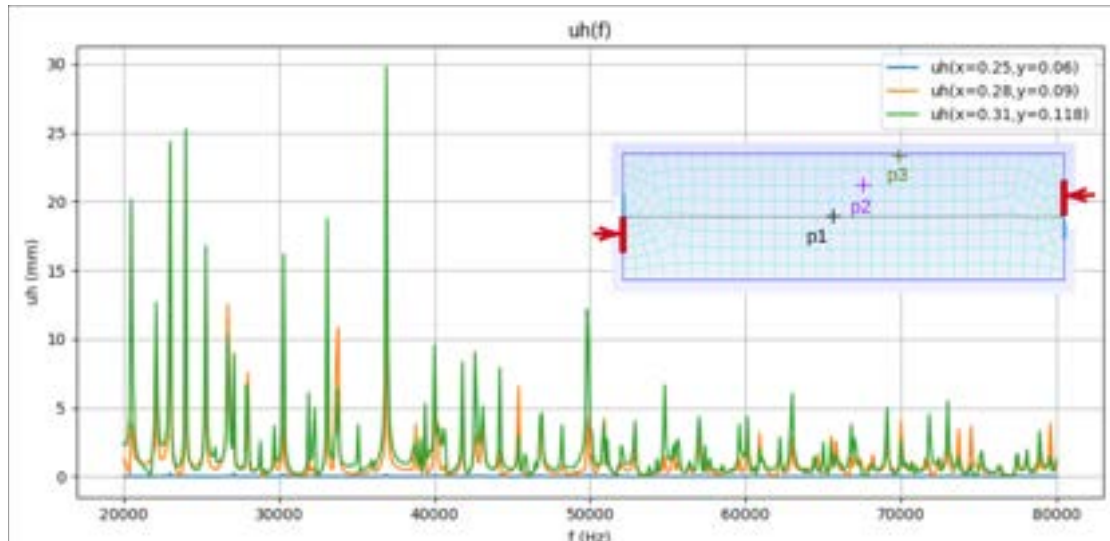


FIGURE 18 – Déplacements particuliers de trois points (p1, p2, p3) de la dalle de mortier pour des fréquences allant de 20 à 80 kHz avec un pas de 100 Hz pour des pompes ayant leurs sources décalées de 20 mm

On observe également les fonctions suivantes : la vitesse ainsi que sa moyenne (figure 19) et la déformation ainsi que sa moyenne (figure 20) mais cette fois-ci dans le cas où les pompes sont décalées de 20 mm par rapport à la droite (D), passant au milieu de l'axe y.

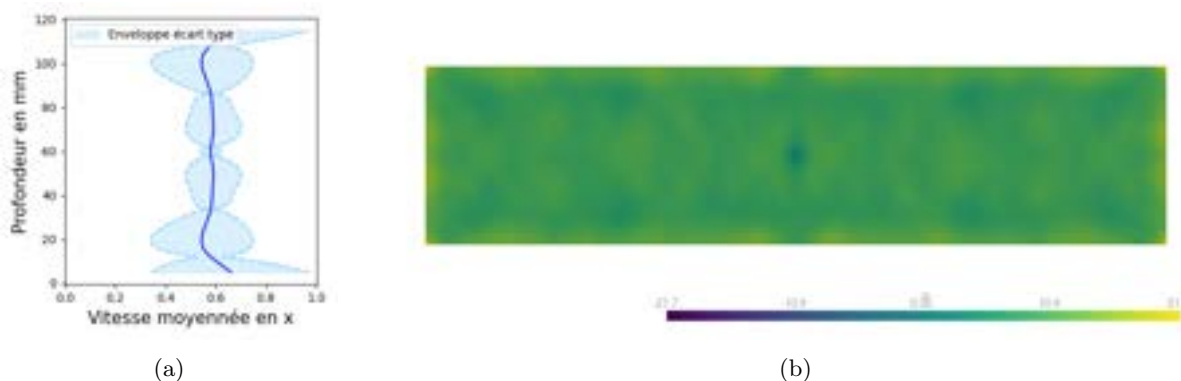


FIGURE 19 – Vitesse avec pompes décalées de 2 cm : (a) Moyenne de la Vitesse particulaire (mm/μs) moyennée sur l'axe des x et tracée en fonction de y et (b) Champ moyen de la vitesse particulaire (mm/μs) pour un balayage fréquentiel de 20 kHz à 80 kHz avec un pas de 100 Hz



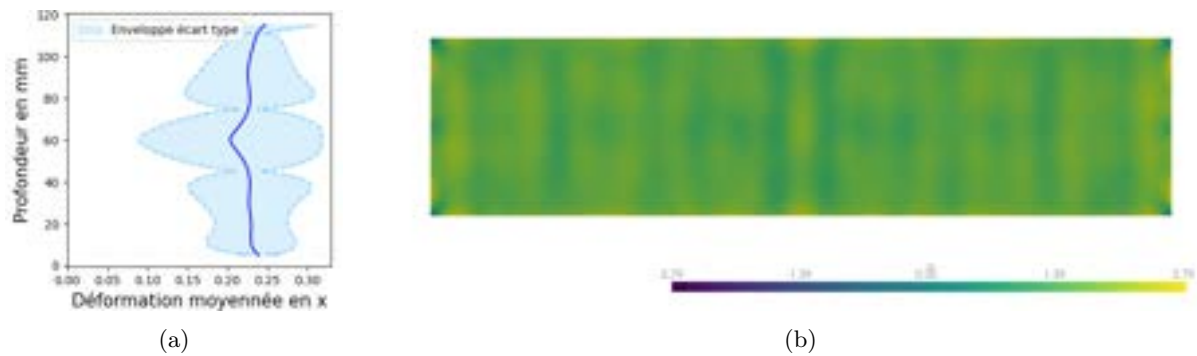


FIGURE 20 – Déformation avec pompes décalées de 2 cm : (a) Moyenne de la déformation  $\text{Tr}(\varepsilon)$  moyennée sur l'axe des  $x$  et tracée en fonction de  $y$  et (b) Champ moyen de la déformation pour un balayage fréquentiel de 20 kHz à 80 kHz avec un pas de 100 Hz

Cette fois, lorsque les pompes sont décalées de 2 cm, pour la déformation, la moyenne de la variance en fonction de la profondeur est de 4.8 % et pour la vitesse la variance est de 5.8 %.

On remarque que le champ moyen de déformation résultant d'un balayage en fréquence d'ondes pompes est plus homogène dans la configuration des pompes décalées que dans celle des pompes symétriques.

On aurait également pu choisir un pas inférieur à 100 kHz pour effectuer le balayage en fréquence, mais il aurait alors fallu paralléliser le code, car cela aurait augmenté le temps de calcul.

### 3.4 Conclusions de fin de section

Dans cette partie du rapport, nous avons commencé par une étude numérique approfondie afin de tester la performance des méthodes de résolution disponibles dans FEniCSx, en particulier celles intégrées à la librairie PETSc. Les résultats ont permis d'identifier la méthode par éléments finis de Lagrange de degré 1 comme ayant un ordre de convergence de degré 1, offrant une précision suffisante pour nos besoins tout en étant relativement peu coûteuse en termes de calcul. Après comparaison des temps de calcul et des performances des différentes combinaisons, le couple solveur CG et préconditionneur LU, fournis par PetsC, s'est avéré être le plus efficace pour résoudre l'équation de Helmholtz dans le domaine fréquentiel.

De plus, l'objectif était de modéliser la propagation du champ de pompe sur une dalle de mortier, et d'analyser l'homogénéité de la déformation engendrée par ce champ. Les simulations ont montré que le tenseur de déformation dans la dalle avait un champ de déformation plus homogène lorsque la modulation est effectuée avec des pompes décalées plutôt que dans la configuration de pompes symétriques. Ces résultats répondent à la première problématique du stage.



## 4 Étude de la contribution non linéaire du mortier en réponse à la propagation des ondes de Rayleigh

**Motivations** L'expérience consiste à envoyer une onde relativement monochromatique, à forte amplitude, sur la surface d'un échantillon de béton, afin d'étudier le phénomène de génération d'harmoniques. Ce phénomène est important car il permet de renseigner sur la non-linéarité du milieu inspecté. Une méthode pour générer une onde de Rayleigh à forte amplitude est d'utiliser un sabot, un dispositif qui convertit une onde de volume en onde de surface (figure 3). Cependant, l'interprétation du signal de seconde harmonique mesuré n'est pas aisée, ce qui motive l'utilisation de la modélisation.

Le sabot a une composition connue : fabriquée en teflon, et, pour un angle précisément déterminé auparavant ( $35^\circ$ ), il permet de convertir une onde de volume, envoyée sur le côté du sabot, en onde de surface (ici de Rayleigh). L'angle est déterminé notamment grâce à la loi de Snell-Descartes (figure 21) :

$$n_1 \sin(\theta_1) = n_2 \sin(\theta_2),$$

où  $n_1$  et  $n_2$  sont les indices de réfraction des milieux dépendant respectivement de la composition du sabot et de la composition de la dalle en mortier.

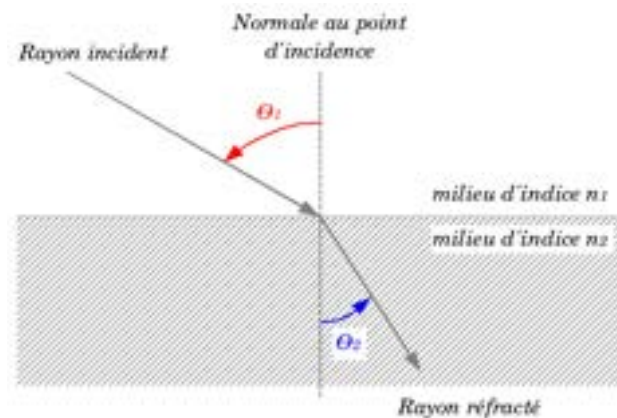


FIGURE 21 – Illustration de la loi de Snell-Descartes

Les difficultés rencontrées incluent la séparation entre la contribution de la non-linéarité du matériau et celle de la pollution harmonique (causée par des artefacts ou des non-linéarités de l'appareillage). De plus, la variabilité des résultats expérimentaux et l'amplitude réduite de la seconde harmonique compliquent l'analyse.

Les travaux de Torello et al. [10] proposent une méthodologie pour mesurer et interpréter correctement un signal de seconde harmonique. Elle consiste à prendre des mesures à la fréquence fondamentale ( $f$ ) et à la seconde harmonique ( $2f$ ) en plusieurs points. Un modèle réaliste de l'évolution des signaux à  $f$  et  $2f$  est utilisé pour ajuster les données et séparer les contributions dues à la pollution et celles dues à la non-linéarité du matériau.

La prochaine étape du stage est d'étudier ce modèle pour connaître la contribution non linéaire du mortier en réponse à la propagation d'une onde de Rayleigh.

**Objectifs** Le but est, dans un premier temps, de connaître le champ de déplacement généré par l'onde de Rayleigh convertie et diffusée par le sabot. On étudiera le champ généré par l'onde à la fréquence fondamentale mais aussi celui généré par sa seconde harmonique qui pourrait "polluer" les résultats. Pour se faire on étudie l'équation scalaire des ondes avec un terme source ayant pour fréquence  $f_1 = 50$  kHz (la fondamentale) et on comparera l'amplitude du déplacement particulière causé par cette onde lorsqu'elle se propage dans l'échantillon avec le déplacement particulière causé par l'onde ayant pour source une onde de fréquence  $2f_1 = 100$  kHz (la seconde harmonique).

Dans un second temps, on cherche à connaître la contribution non-linéaire du matériau en réponse à l'onde de Rayleigh. Pour se faire on étudiera l'équation de Helmholtz modélisée avec un tenseur de contrainte non-linéaire. Cette étude nous permet de passer sur un système dit "quasi-linéaire" et c'est la résolution de ce système qui nous donnera la contribution du matériau en réponse à l'onde de surface émise par le sabot.

**Paramètres** L'étude se fait toujours sur la même dalle avec les mêmes dimensions, mais cette fois ci nous prenons une vue de dessus dans le plan ( $x=50$  cm,  $Z=25$  cm). Le sabot est éloigné de 3 cm du bord de la dalle et il a les dimensions suivantes : Longueur = 10 cm, largeur = 9 cm (cf figure 22).

#### 4.1 Étude de l'équation scalaire linéaire

Soient  $\Omega = [0, L_1 = 50] \times [0, l_1 = 25]$  représentant la dalle en 2D vue du dessus et  $I = [0, L_2 = 10] \times [0, l_2 = 9]$  représentant le sabot posé sur la dalle lors de l'expérience tel que  $I \subset \Omega$  (figure 22). On cherche à résoudre l'équation scalaire de propagation dans le domaine harmonique telle que :

$$-\nabla^2 u - \frac{\omega^2}{c^2} u = \begin{cases} 0 & \text{dans } \Omega \setminus I \text{ (maillage vert)} \\ e^{i\frac{\omega}{c}x} & \text{dans } I \text{ (maillage orange)} \end{cases}$$

qui est équivalent à

$$-\rho c^2 \nabla^2 u - \rho \omega^2 u = \begin{cases} 0 & \text{dans } \Omega \setminus I \text{ (maillage vert)} \\ \rho c^2 e^{i\frac{\omega}{c}x} & \text{dans } I \text{ (maillage orange)} \end{cases}$$

On aimerait ajouter à cela la condition de bords transparents et pour se faire on utilisera la notion d'impédance  $Z = \rho c$ , c'est à dire :

$$\frac{\partial u}{\partial n} = i\omega Z u = i\omega \rho c u \text{ sur } \partial\Omega$$

Cette condition est approchée et n'est bonne que si l'onde arrivant sur le bord ressemble à une onde plane en incidence normale.

##### 4.1.1 Formulation faible

On pose  $q(x) = \begin{cases} 0 & \text{dans } \Omega \setminus I \\ \rho c^2 e^{i\frac{\omega}{c}x} & \text{dans } I \end{cases}$ . Soit  $v \in H^1(\Omega)$  une fonction test que nous multiplions à l'équation définie ci-dessus et que nous intégrons sur le domaine  $\Omega$  :

$$-\rho c^2 \int_{\Omega} \nabla^2 u v \, dx - \rho \int_{\Omega} \omega^2 u v \, dx = \int_{\Omega} q v \, dx.$$

En appliquant la formule de Green sur le premier terme, on obtient :

$$\rho c^2 \int_{\Omega} \nabla u \cdot \nabla v \, dx - \rho c^2 \int_{\partial\Omega} v \frac{\partial u}{\partial n} \, ds - \rho \int_{\Omega} \omega^2 u v \, dx = \int_{\Omega} q v \, dx.$$

En considérant la condition de bords transparents, et comme  $\mu = \rho c^2$ , l'égalité se réécrit de la façon suivante :

$$\mu \left( \int_{\Omega} \nabla u \cdot \nabla v \, dx - i\rho\omega c \int_{\partial\Omega} v u \, ds \right) - \rho\omega^2 \int_{\Omega} u v \, dx = \int_{\Omega} q v \, dx.$$

Le problème variationnel s'écrit alors : Trouver  $u \in H^1(\Omega)$  tel que

$$a(u, v) = L(v) \text{ pour tout } v \in H^1(\Omega) \quad (11)$$

avec

$$\begin{aligned} a(u, v) &= \mu \left( \int_{\Omega} \nabla u \cdot \nabla v \, dx - i\rho\omega c \int_{\partial\Omega} v u \, ds \right) - \rho\omega^2 \int_{\Omega} u v \, dx, \\ L(v) &= \int_{\Omega} q v \, dx. \end{aligned}$$

##### 4.1.2 Premiers résultats numériques

Afin de modéliser et de résoudre cette équation, on commence par générer un domaine correspondant à la géométrie de la dalle en vue de dessus (figure 22) et on y inclut l'empreinte du sabot qui sera la source de l'onde de Rayleigh. On lui donne un tag particulier (ici 201) pour pouvoir écrire la fonction  $L$  du problème variationnel.

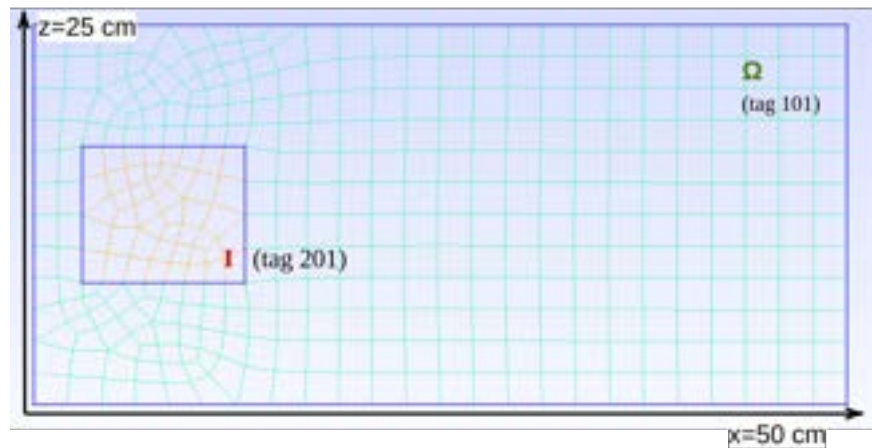


FIGURE 22 – Maillage de la dalle vue de dessus avec l'empreinte du sabot

Voici une partie du code permettant de résoudre l'équation pour une fréquence  $F = 50$  kHz et l'équation pour la seconde harmonique c'est-à-dire  $F_2 = 2F = 100$  kHz.

---

```
# Définition de l'espace géométrique, des éléments finis et leur degré de liberté
V1 = fem.functionspace(domain2, ("Lagrange", 1, (gdim, )))

# Définition des fonctions sources
def f_1(x):
    return rho*cR**2 * np.exp(1j * (k) * x[0])

def f_2(x):
    return rho*cR**2 * np.exp(1j * (2*k) * x[0])
f1 = fem.Function(V1)
f1.interpolate(f_1)
f2_input = fem.Function(V1)
f2_input.interpolate(f_2)

# Définition des espaces de fonctions mathématiques
u1 = ufl.TrialFunction(V1)
v1 = ufl.TestFunction(V1)

dx = ufl.Measure("dx", domain=domain2, subdomain_data=cell_tags2)
ds = ufl.Measure("ds", domain=domain2, subdomain_data=facet_tags2)

# Définition des formes bilinéaires et linéaires
a_1 = (
    mu * inner(grad(u1), grad(v1)) * dx
    - mu * rho * 1j * omega * cR * inner(u1, v1) * ufl.ds
    - rho * omega**2 * inner(u1, v1) * dx
)
L_1 = inner(f1, v1) * dx(201)

a_2_input = (
    mu * inner(grad(u1), grad(v1)) * dx
    - mu * 1j * 2*omega * cR * rho * inner(u1, v1) * ds
    - rho * 4* omega**2 * inner(u1, v1) * dx
)
L_2_input = inner(f2_input, v1) * dx(201)
```

---

```

# Définition des options du solveur
petsc_options = {
    "ksp_type": "cg",      # Utilisation du solveur GMRES
    "pc_type": "lu"        # Type de matrice par défaut
}

# Création du problème linéaire
problem1 = LinearProblem(a_1, L_1, bcs=[], petsc_options=petsc_options)
problem2 = LinearProblem(a_2_input, L_2_input, bcs=[], petsc_options=petsc_options)

# Résolution du problème
uh_1 = problem1.solve()
uh_2_input = problem2.solve()

```

On résout le problème variationnel par éléments finis de Lagrange de degré 1 et on observe sur la figure 23 la solution à l'équation pour une source ayant une fréquence de 50 kHz. On observe également sur la figure 24 le déplacement particulière dû à la seconde harmonique de l'onde de Rayleigh (source ayant une fréquence de 100 kHz) qui nous intéresse particulièrement car c'est une éventuelle contribution polluant l'étude des paramètres non linéaires du matériau.

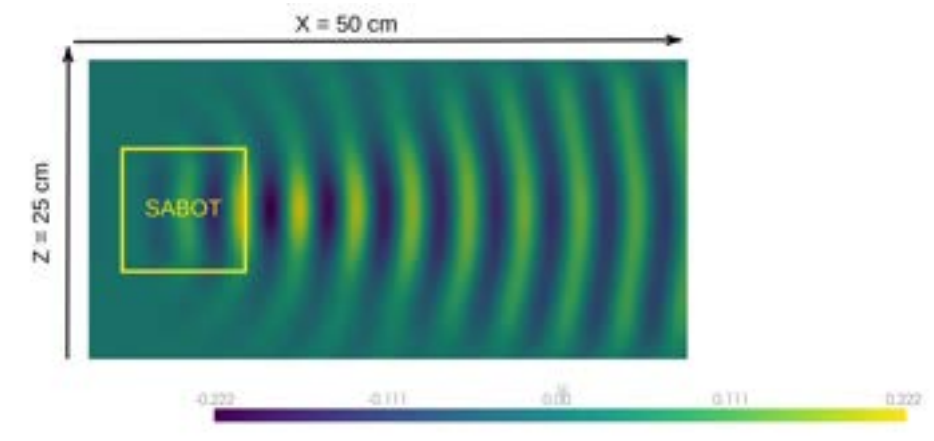


FIGURE 23 – Visualisation pyvista du déplacement particulier (mm), solution de l'équation pour une source ayant une fréquence de 50 kHz

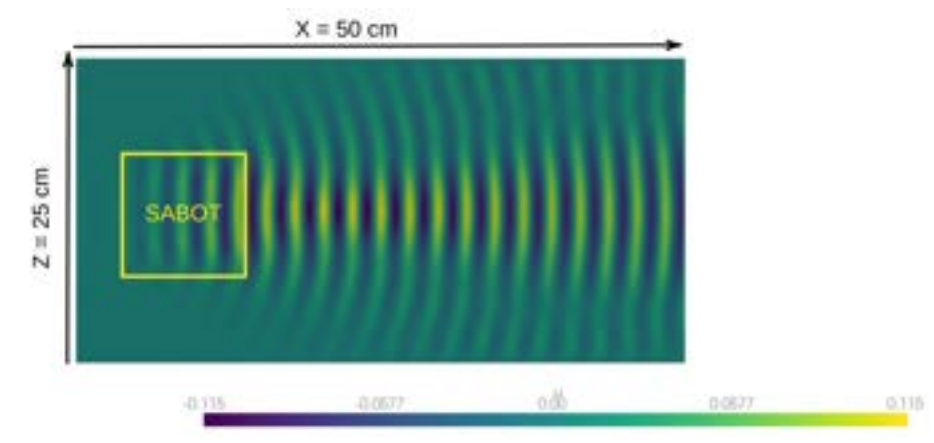


FIGURE 24 – Visualisation pyvista du déplacement particulier (mm), solution de l'équation pour une source ayant une fréquence de 100 kHz

Afin de mieux connaître le comportement de l'onde de surface sur le matériau, on trace sur la figure 25 les solutions de l'équation pour des sources ayant des fréquences de 50 et 100 kHz, évaluées en  $y=12$  cm (correspondant à la moitié de la largeur du domaine) et tracées en fonction de  $x$ .

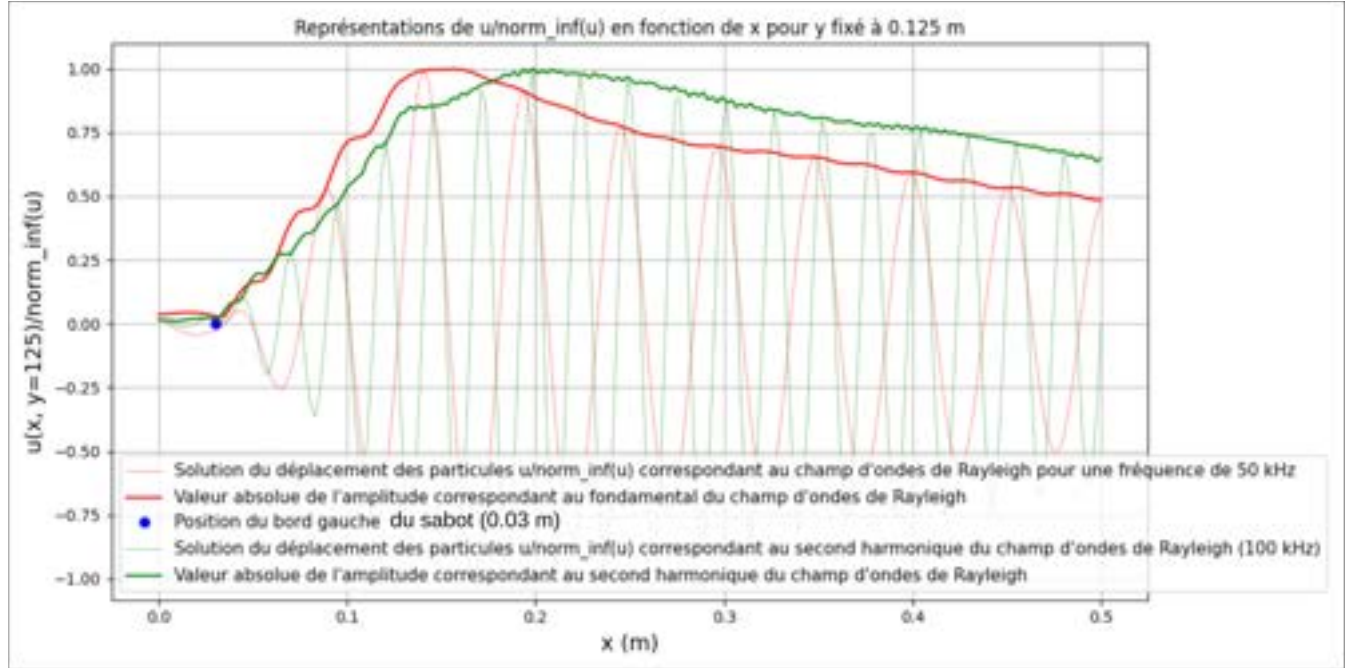


FIGURE 25 – Amplitude des déplacements particuliers normalisé résultant des première et seconde harmoniques de l'onde de Rayleigh, évaluée en  $Z = l/2 = 12.5$  cm

## 4.2 Système d'équations quasi-linéaire pour l'étude de la contribution non linéaire du matériau

### 4.2.1 Étude du système quasi-linéaire

Maintenant que l'on connaît la forme du champ de propagation des première et seconde harmoniques de l'onde en surface de la dalle, on peut s'intéresser à la contribution non linéaire du mortier en réponse à cette onde. Pour se faire nous étudions le système quasi-linéaire modélisant la non-linéarité d'une onde longitudinale dans le sabot, étudiée dans l'article de E. A. Zabolotskaya et al. publié dans Acoustical Society of America en 1993 [15]. En posant le nombre d'onde  $k = \frac{\omega}{c}$  et  $A_i$ ,  $i \in [1, 2]$  le coefficient d'atténuation de l'onde de Rayleigh à la fréquence  $i\omega$ , le système quasi-linéaire est le suivant :

$$(\nabla^2 + k^2 + ikA_1)v_1 = q(x)/\rho c^2, \quad (12)$$

$$(\nabla^2 + 4k^2 + 2ikA_2)v_2 = E\beta v_1^2, \quad (13)$$

avec  $q(x) = \begin{cases} 0 & \text{dans } \Omega \setminus I \\ \rho c^2 e^{i\frac{\omega}{c}x} & \text{dans } I \end{cases}$  et  $\beta$  est le coefficient de non-linéarité du matériau défini dans (1).

Une explication de l'obtention du système quasi-linéaire est placée en annexe 7.5 On veut retrouver une équation ressemblant à celle du problème variationnel 11, on va donc multiplier par  $\rho c^2 = \mu$  de chaque côtés et simplifier le terme de droite de l'égalité (13) en posant  $E\beta = 1$  pour la modélisation. On associe à ce système la condition de bords transparents  $\frac{\partial u}{\partial n} = i\omega Z u = i\omega \rho c u$  sur  $\partial\Omega$  et on en déduit la formulation faible suivante : Trouver  $v_1, v_2 \in V$  tels que ;

$$\rho c^2 \int_{\Omega} \nabla v_1 \nabla \phi \, dx - i\rho\mu\omega c \int_{\partial\Omega} v_1 \phi \, ds - \rho(i\omega c A_1 + \omega^2) \int_{\Omega} v_1 \phi \, dx = \int_{\Omega} q \phi \, dx, \quad \forall \phi \in V, \quad (14)$$

$$\rho c^2 \int_{\Omega} \nabla v_2 \nabla \phi \, dx - i\rho\mu\omega c \int_{\partial\Omega} v_2 \phi \, ds - \rho(2i\omega c A_2 + 4\omega^2) \int_{\Omega} v_2 \phi \, dx = \mu \int_{\Omega} v_1^2 \phi \, dx, \quad \forall \phi \in V. \quad (15)$$

Les coefficients d'atténuations suivent une courbe que l'on peut voir sur la figure 26. Nous choisissons de les prendre nuls pour la suite de l'étude. Le travail sur ces coefficients est une idée d'étude pour la suite mais ne fera pas l'objet d'étude dans ce rapport.

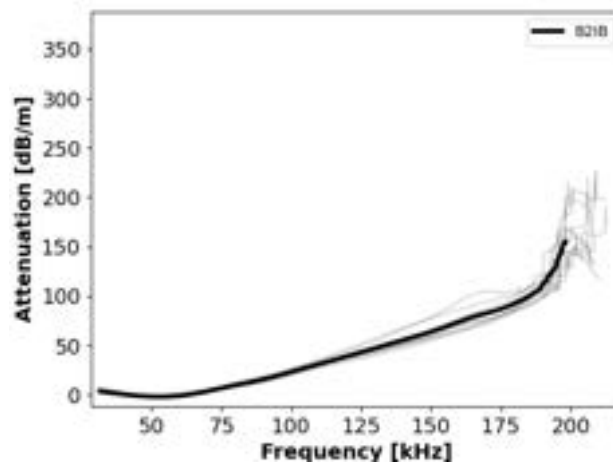


FIGURE 26 – Courbe d'atténuation en fonction de la fréquence de l'onde source dans une dalle de béton

#### 4.2.2 Résultats numériques

On résout les deux équations linéaires par éléments finis de Lagrange quadrangulaires de degré 1. On observe la contribution non linéaire du mortier en réponse à l'onde de Rayleigh sur la figure 27. La figure 28 nous permet d'observer l'atténuation de la fréquence fondamentale et de la seconde harmonique de l'onde de Rayleigh en fonction de la distance, ainsi que la contribution du matériau correspondant à la réponse en seconde harmonique de l'onde de Rayleigh.

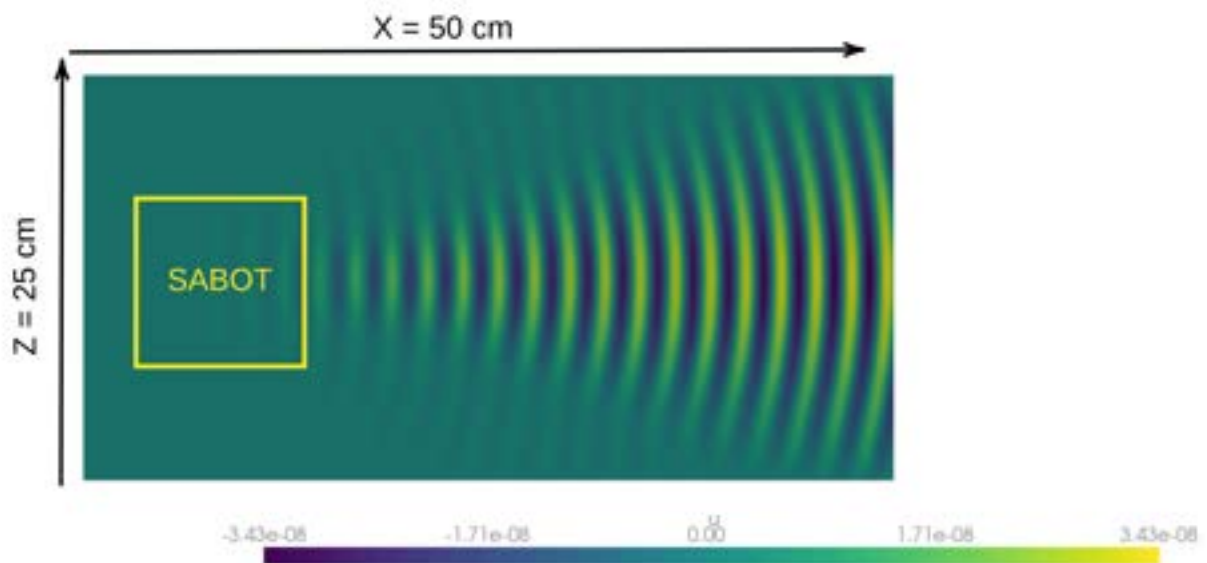


FIGURE 27 – Déplacement particulière (mm) résultant de la contribution non linéaire du matériau en réponse à l'onde de Rayleigh pour une fréquence de 100 kHz



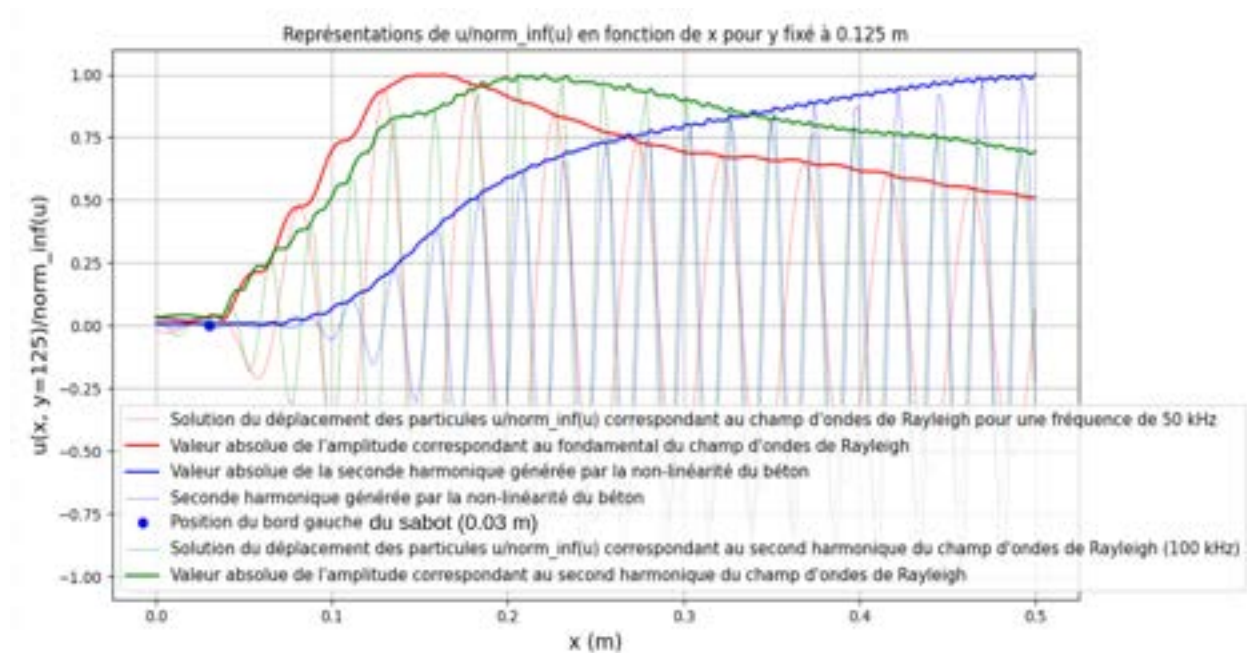


FIGURE 28 – Amplitude du déplacement particulaire émis par l'onde de Rayleigh (rouge pour la fondamentale et verte pour la seconde harmonique) et du déplacement particulaire émis par la contribution non linéaire du matériau (bleue)

### 4.3 Conclusions de fin de section

Dans cette partie du travail, nous avons étudié la propagation de ces ondes dans une dalle de mortier à l'aide du modèle sabot/dalle, utilisé dans les expériences de Massina. Le sabot, dont la composition est connue, permet de convertir une onde de volume (émise par un transducteur piézoélectrique) en onde de surface grâce à un angle spécifique défini par la loi de Snell-Descartes (figure 29).

L'objectif principal était d'analyser à la fois la contribution linéaire et non linéaire du matériau en réponse à la propagation des ondes de Rayleigh. Une première étude a porté sur la modélisation du champ de l'onde fondamentale de fréquence 50 kHz émise par le sabot, ainsi que sur la génération du second harmonique à 100 kHz. La comparaison des amplitudes du déplacement particulaire des ondes fondamentales et de leur harmonique a permis d'évaluer la possibilité de "pollution" des résultats par des effets harmoniques indésirables.

Ensuite, l'étude a porté sur la contribution non linéaire du matériau en modélisant l'équation de Helmholtz avec un tenseur de contrainte non-linéaire. Cette modélisation nous a permis de passer à un système quasi-linéaire dont la résolution a fourni une première estimation de la réponse non linéaire du mortier sous l'effet des ondes de surface. Ces résultats ouvrent la voie à une meilleure compréhension des phénomènes non linéaires dans la propagation des ondes de Rayleigh, essentiels pour caractériser la réponse des matériaux soumis à des dégradations ou fissures.

Afin d'observer des résultats se rapprochant des études sur le béton, nous allons maintenant procéder à la modélisation des granulats à l'intérieur de la dalle.



FIGURE 29 – Photo du dispositif expérimental de génération d'ondes de surface



## 5 Modélisation du squelette granulaire du béton et résolution du système quasi-linéaire sur la dalle de béton

Afin de modéliser au mieux les expériences réalisées par Massina, l'objectif principal de cette section est de générer un maillage de la dalle en y ajoutant des granulats, chacun représenté par une ellipse. Le second objectif sera de résoudre les équations étudiées précédemment, cette fois sur une dalle de béton, en assignant des paramètres matériau différents aux ellipses.

Un code sera utilisé exclusivement pour générer une liste d'ellipses (petit rayon, grand rayon, coordonnées du centre, angle), qui servira ensuite dans Gmsh pour créer un maillage. Ce maillage sera ensuite importé dans le code principal pour la résolution des équations.

### 5.1 Étude granulométrique et génération d'ellipses

Les ellipses générées doivent respecter une distribution granulométrique en termes de taille, ainsi qu'un certain ratio de surface par rapport à la surface totale.

Nous devons d'abord effectuer une étude granulométrique afin de respecter les dimensions des granulats et leur répartition sur la dalle en deux dimensions. Dans le programme, la distribution des granulats est modélisée à partir de données réelles issues des fiches techniques des échantillons de béton fabriqués pour les expériences : figures 30 et 31.

	20	1.40	D	D/1.4	d	d/2	f	FI	LA
Tamis (mm)	25	18	12.5	9	4	2			
maxi			97.0	74.5	4.3	2.4	1.33	14	26
Xf+1.25 Sf			96.2	69.4	3.4	1.7	0.81		25.2
moyenne Xf	100.0	100.0	93.6	61.3	2.4	1.1	0.54	12.8	23.7
Xf - 1.25 Sf	100.0	100.0	91.0	53.3					
mini	100.0	100.0	87.0	42.4					
Ecart type Sf	0.00	0.00	2.08	6.46	0.79	0.42	0.220		1.20
nb. valeurs	51	51	51	51	51	51	51	10	24

FIGURE 30 – Proportion du diamètre des graviers dans le béton fournie par le fournisseur des échantillons de béton étudiés



FIGURE 31 – Courbe granulométrique théorique fournie par le fournisseur des échantillons de béton étudiés

Une interpolation par spline cubique de type "clamped", provenant de la bibliothèque scipy, est utilisée pour obtenir une courbe lisse représentant cette distribution. Cela permet de générer des courbes granulométriques correspondant aux spécifications des échantillons de béton que l'on peut observer sur la figure 32.

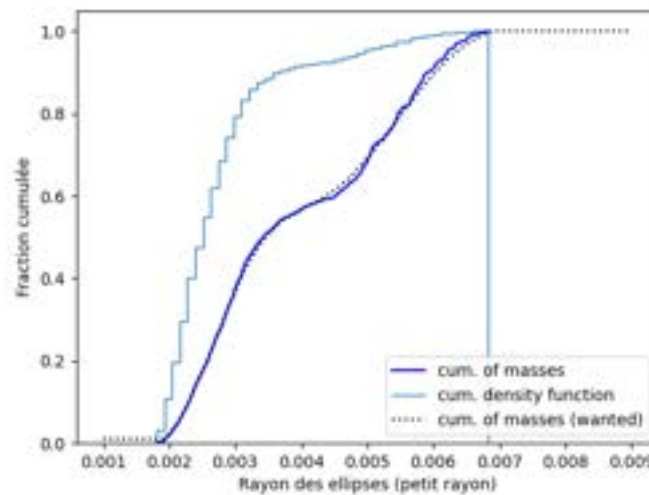


FIGURE 32 – Courbes des cumulées de masse (cum. of masses) et de densité granulométrique (cum. density function) et courbe de masse granulométrique que l'on veut approcher (cum. of masses (wanted))

L'objectif est de remplir une surface donnée (une dalle de béton en 2D) avec des ellipses représentant des granulats, jusqu'à ce que la somme des surfaces de ces ellipses atteigne un ratio prédéfini par rapport à la surface totale de la dalle (ici 50%). Les ellipses sont générées aléatoirement en respectant une distribution granulométrique, avec des tailles, formes et orientations variables.

#### Fonctionnement de l'algorithme :

- **Initialisation :**
  - Fixer un rapport surfacique cible pour les ellipses par rapport à la dalle (ici, 50%).
  - Définir une courbe granulométrique qui spécifie la taille et la forme des ellipses à générer.
- **Génération d'une ellipse :**
  - Générer aléatoirement une ellipse en suivant la courbe granulométrique. Les paramètres générés incluent le grand et le petit diamètre, ainsi que l'angle d'inclinaison.
  - Tirer aléatoirement une position  $(x, y)$  pour son centre dans la surface de la dalle.
- **Vérification de chevauchement :**
  - Vérifier si la nouvelle ellipse chevauche celles déjà placées :
    - **Test rapide :** Vérifier avec les ellipses ayant des grands rayons proches.
    - **Test détaillé :** Si la nouvelle ellipse est proche d'une autre, effectuer un test de chevauchement plus précis en vérifiant les premiers voisins.
  - Si chevauchement :
    - Si le nombre d'échecs dépasse une limite prédéfinie, l'algorithme abandonne la génération de l'ellipse et passe à la suivante.
    - Sinon, générer une nouvelle position aléatoire et réessayer.
- **Itération :**
  - Répéter l'opération jusqu'à atteindre le rapport surfacique cible ou jusqu'à ce que le nombre maximal de tentatives soit atteint.

**Pseudocode de l'algorithme :**

Initialiser le ratio surfacique cible

Tant que (surface totale des ellipses < ratio cible) :

    Générer une nouvelle ellipse (taille, forme, orientation)

    Tirer une position (x, y) pour son centre

    Tant que (tentatives < limite d'échecs) :

        Si chevauchement avec une ellipse existante :

            Effectuer un test rapide de non-chevauchement (grands rayons)

            Si nécessaire, effectuer un test plus précis (voisins proches)

            Si chevauchement confirmé :

                Retirer une nouvelle position

                Incrémenter le compteur de tentatives

        Sinon :

            Ajouter l'ellipse à la liste des ellipses

            Break

Fin Tant que

Enfin, une analyse du nombre d'itérations nécessaires, observable sur la figure 33, pour atteindre un certain ratio de surface montre qu'au-delà d'un ratio de 0.5, le nombre d'itérations augmente de manière exponentielle. Il n'est donc pas nécessaire de dépasser ce ratio pour modéliser la dalle de béton de manière réaliste.

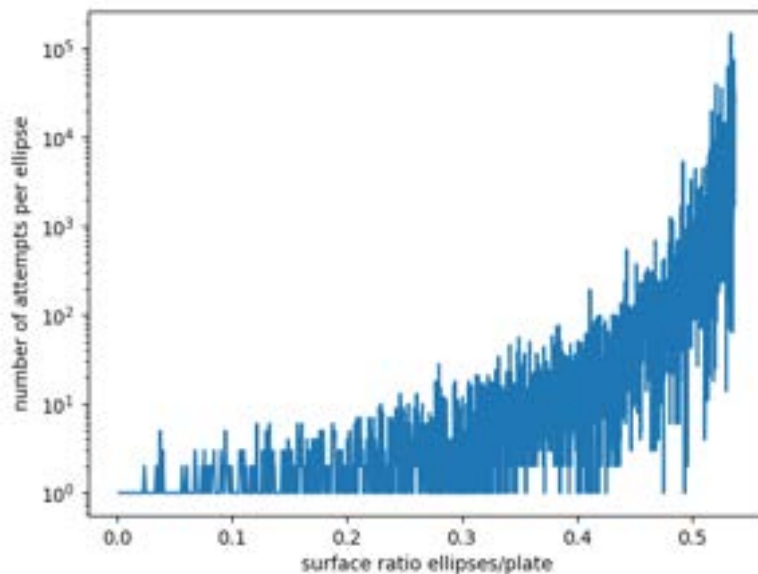


FIGURE 33 – Nombre d'itérations nécessaires pour couvrir un ratio surfacique par les ellipses

Le maillage final obtenu présente un ratio de 50 % de surface recouverte par les ellipses. Ce résultat est ensuite exporté sous forme de fichier CSV, prêt à être utilisé pour la génération du maillage final.

Une illustration du maillage obtenu à la fin est observable sur la figure 34.

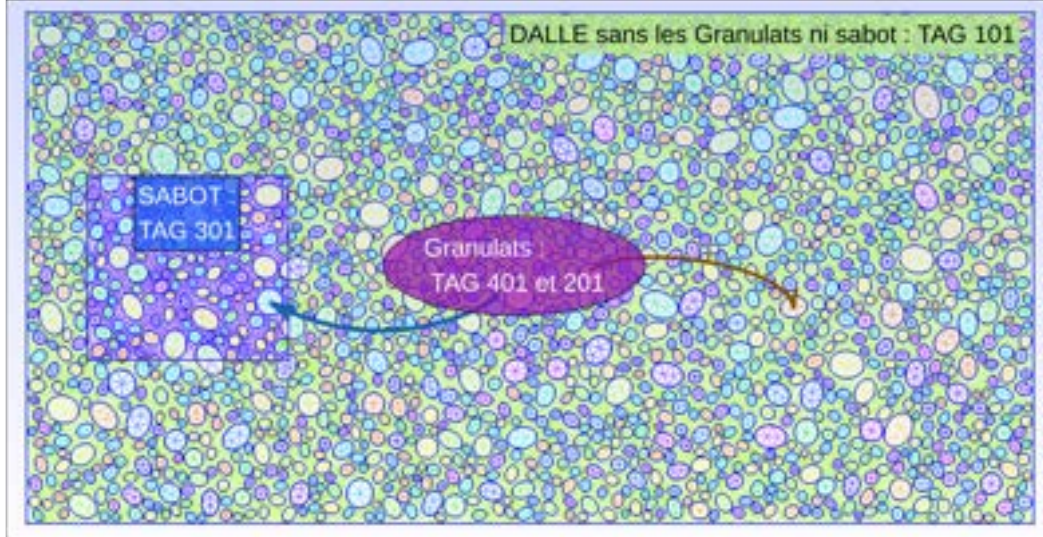


FIGURE 34 – Maillage de la dalle de béton avec un ratio surfacique recouvert par les ellipses de 50%

## 5.2 Étude de la contribution non linéaire du béton en réponse à la propagation des ondes de Rayleigh

Après avoir généré 50 listes de répartitions d'ellipses différentes pour la géométrie 25x50 pour modéliser 50 répartitions d'ellipses différentes, nous générons 50 maillages à l'aide de Gmsh et Fenicsx, puis nous résolvons le système quasi-linéaire sur chacun des domaines. Dans un premier temps, nous considérons une atténuation nulle et un équilibre homogène entre les granulats et le ciment, afin de vérifier les résultats en les comparant à ceux obtenus dans la section 4.2.2.

Pour modéliser les nouveaux paramètres matériau correspondant aux granulats, nous attribuons un poids (le ratio surfacique ici) aux paramètres de la matrice de ciment et aux granulats, afin de les différencier dans la suite des calculs. Pour cela, nous fixons  $\alpha_1 = 0.5$  et  $\alpha_2 = 0.5$ , qui correspondent aux poids attribués respectivement au ciment et aux granulats. Les paramètres matériau choisis sont les suivants :

$$\begin{aligned}\rho_1 &= \bar{\rho}(1 - p\alpha_2), \\ \rho_2 &= \bar{\rho}(1 + p\alpha_1), \\ \mu_1 &= \bar{\mu}(1 - p\alpha_2), \\ \mu_2 &= \bar{\mu}(1 + p\alpha_1),\end{aligned}$$

où  $\bar{\rho} = \rho = 2395 \text{ kg/m}^3$  est la masse volumique du béton et  $\bar{\mu} = c^2 \bar{\rho}$ , avec  $c$  la vitesse de l'onde de Rayleigh dans le béton. On note également que le module de Young du béton a une valeur de  $E = 38 \text{ GPa}$ . Ces valeurs sont prises dans le but de représenter au mieux le milieu homogénéisé, on veut les garder fixes tout en faisant varier le paramètre de contraste  $p$  entre les milieux.

Le paramètre de contraste  $p$  permet de gérer les paramètres matériaux ; on veut  $p < \frac{1}{\alpha_2}$  et  $\mu_1 = \mu_2$ , si  $p = 0$ , cela correspond à un modèle de dalle homogène. Le premier milieu (mortier) sera moins dense et plus souple tandis que le second (granulats) sera plus dense et plus rigide. On choisit des coefficients d'atténuations nuls.

Soient  $\Omega_{(101,301)}$ ,  $\Omega_{(201,401)}$ ,  $\Omega_{(301,401)}$  définissant respectivement les parties du domaine comprenant l'empreinte du sabot (tag 301) et la dalle entière sans les granulats (tag 101), les granulats dans la dalle (tag 201) et dans l'empreinte du sabot (tag 401) (figure 34). On peut alors réécrire le système quasi-linéaire, prenant en compte ces nouveaux paramètres pour chaque matériau et en posant la fonction

$$q(x) = \begin{cases} 0 & \text{dans } \Omega \setminus \text{SABOT}, \\ \bar{\rho} c^2 e^{i \frac{\omega}{c} x} & \text{dans SABOT}, \end{cases} \quad (\text{similaire à celle de l'équation 14}). \text{ On a :}$$



$$\begin{aligned}
& \rho_1 c^2 \int_{\Omega_{(101,301)}} \nabla v_1 \cdot \nabla \phi \, dx + \rho_2 c^2 \int_{\Omega_{(201,401)}} \nabla v_1 \cdot \nabla \phi \, dx \\
& - i \rho_1 \mu_1 \omega c \int_{\partial \Omega} v_1 \phi \, ds \\
& - \rho_1 (i \omega c A_1 + \omega^2) \int_{\Omega_{(101,301)}} v_1 \phi \, dx \\
& - \rho_2 (i \omega c A_1 + \omega^2) \int_{\Omega_{(201,401)}} v_1 \phi \, dx = \int_{\Omega} q \phi \, dx, \quad \forall \phi \in V, \quad (13)
\end{aligned}$$

$$\begin{aligned}
& \rho_1 c^2 \int_{\Omega_{(101,301)}} \nabla v_2 \cdot \nabla \phi \, dx + \rho_2 c^2 \int_{\Omega_{(201,401)}} \nabla v_2 \cdot \nabla \phi \, dx \\
& - i \rho_1 \mu_1 \omega c \int_{\partial \Omega} v_2 \phi \, ds \\
& - \rho_1 (2i \omega c A_2 + 4\omega^2) \int_{\Omega_{(101,301)}} v_2 \phi \, dx \\
& - \rho_2 (2i \omega c A_2 + 4\omega^2) \int_{\Omega_{(201,401)}} v_2 \phi \, dx = \bar{\rho} c^2 \int_{\Omega} v_1^2 \phi \, dx, \quad \forall \phi \in V. \quad (14)
\end{aligned}$$

On observe sur la figure 35 les amplitudes des solutions du système variationnel (13) et (14) résolu en éléments finis de Lagrange de degré 1, sur 50 domaines modélisés avec des répartitions d'ellipses différentes, avec des paramètres matériaux homogènes ( $p = 0$ ).

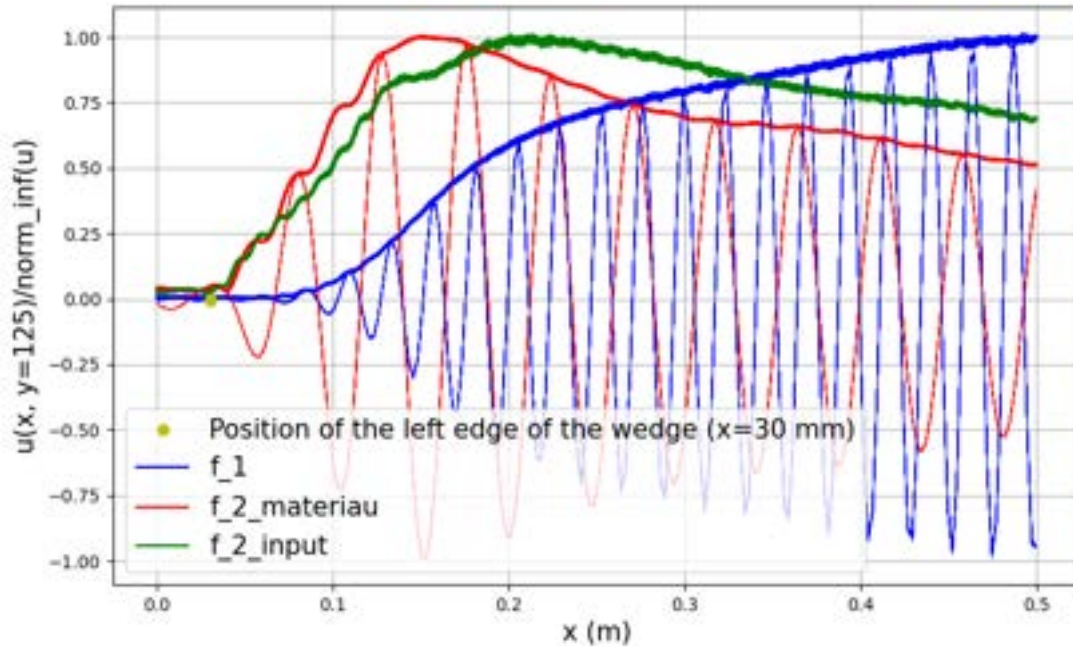


FIGURE 35 – Superpositions des solutions normalisées des équations résolues sur tous les domaines pour une onde de Rayleigh propagée à une fréquence de 50 kHz et une dalle totalement homogène ( $p = 0$ )

Les formes des courbes obtenues sont similaires à celles issues de l'étude sur une dalle homogène, ce qui nous permet désormais d'ajuster les paramètres du matériau en modélisant des dalles présentant des phénomènes de dispersion plus ou moins marqués, en fonction de la valeur de la variable  $p$ . Ainsi, une valeur de  $p = 0.14$  correspondra à une forte dispersion, tandis que  $p = 0.07$  représentera une dispersion modérée.

La figure 36 illustre la dispersion de l'onde liée à la contribution non linéaire du matériau pour une valeur de  $p = 0.14$ , avec une fréquence de réponse de 100 kHz. Une dispersion similaire est observée pour une fréquence de 160 kHz sur la figure 37.

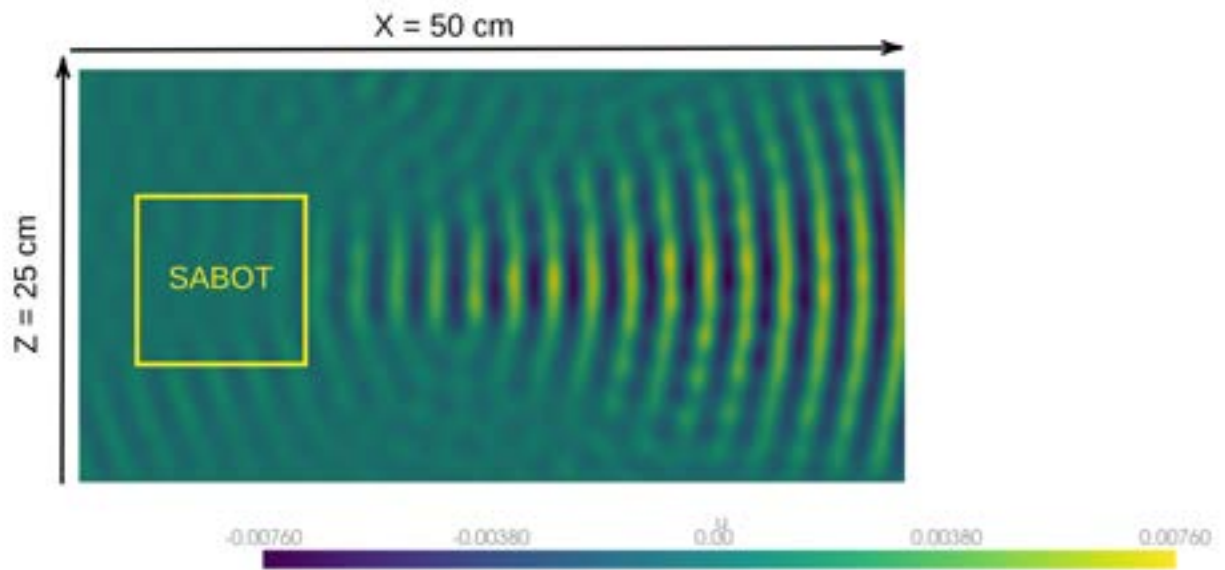


FIGURE 36 – Contribution du matériau non-linéaire lorsque la variable  $p = 0.14$  entraîne une dispersion de l'onde de Rayleigh sur la dalle à 100 kHz

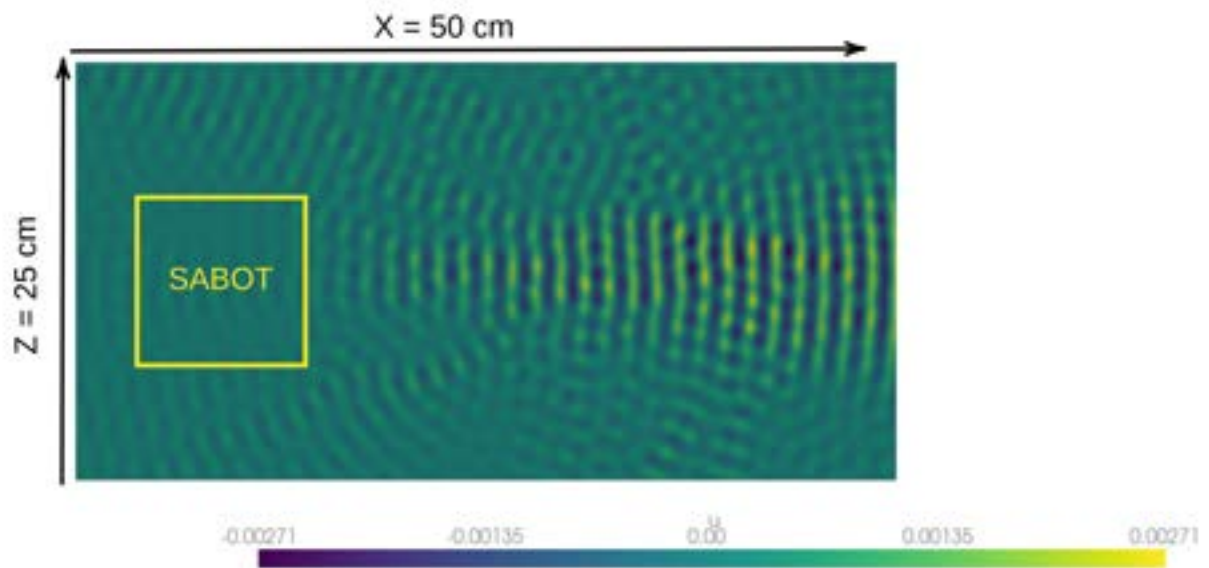
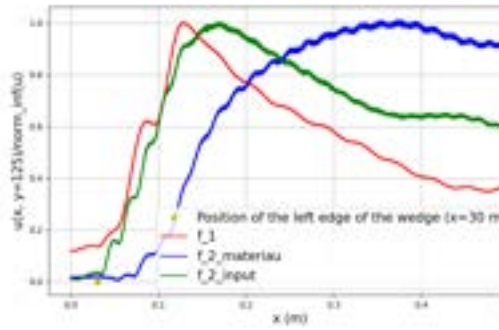
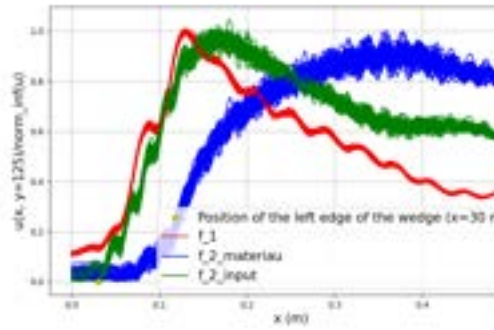
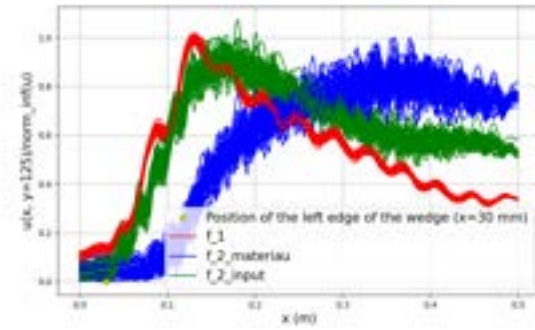
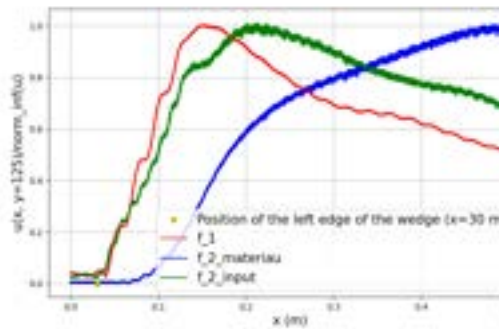
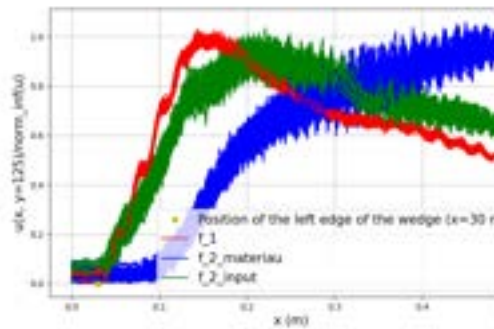
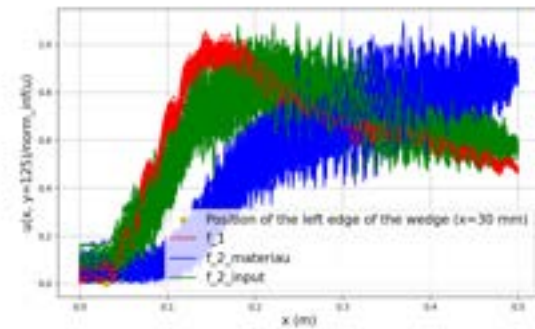
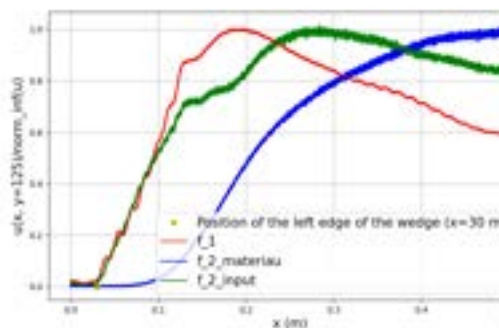
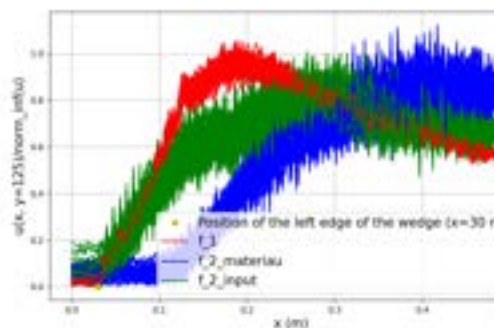
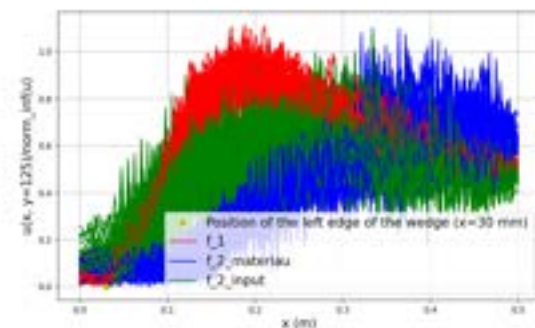


FIGURE 37 – Contribution du matériau non-linéaire lorsque la variable  $p = 0.14$  entraîne une dispersion de l'onde de Rayleigh sur la dalle à 160 kHz

Nous pouvons maintenant observer, sur les figures des tables 2, 3 et 4, le comportement des valeurs absolues des solutions normalisées, pour les trois valeurs attribuées au paramètre de contraste  $p$  et rendant plus ou moins visible les granulats dans l'équation, ce pour 3 valeurs de fréquences différentes : 30, 50 et 80 kHz.

(a)  $p = 0.0$  (homogène)(b)  $p = 0.07$  (peu hétérogène)(c)  $p = 0.14$  (très hétérogène)TABLE 2 – Solutions à 30 kHz pour différentes valeurs de  $p$ .(a)  $p = 0.0$  (homogène)(b)  $p = 0.07$  (peu hétérogène)(c)  $p = 0.14$  (très hétérogène)TABLE 3 – Solutions à 50 kHz pour différentes valeurs de  $p$ .(a)  $p = 0.0$  (homogène)(b)  $p = 0.07$  (peu hétérogène)(c)  $p = 0.14$  (très hétérogène)TABLE 4 – Solutions à 80 kHz pour différentes valeurs de  $p$ .



### 5.3 Conclusions de fin de section

Cette étude a permis de développer un modèle numérique capable d'intégrer la présence de granulats dans une dalle de béton à travers l'utilisation d'ellipses, représentant leur répartition réelle. La génération d'un maillage adapté en intégrant les granulats représentés par des ellipses a permis de modéliser avec plus de vraisemblance la dalle de béton utilisée dans les expériences de Massina. Le maillage, associé à une résolution des équations quasi-linéaires sur plusieurs configurations, a permis de valider la capacité du modèle à capturer la propagation des ondes de Rayleigh dans des milieux hétérogènes.

Les résultats obtenus avec des paramètres homogènes, sans ellipses/granulats ( $p = 0$ ), ont montré une bonne cohérence avec ceux issus du modèle homogène, confirmant ainsi la fiabilité du code développé et des maillages utilisés.

De manière générale, cette étude permet aussi de constater les limites de la méthodologie habituelle lorsqu'il s'agit de mesurer un paramètre matériau à partir du signal à la fréquence harmonique  $2f_{matériau}$  dans un milieu diffusant. À partir d'un certain contraste ou d'une fréquence donnée, la variabilité devient considérable, et effectuer la mesure uniquement sur un profil en  $x$  ne suffit plus pour séparer les contributions de la pollution  $2f_{input}$  et de  $2f_{matériau}$  par ajustement. On pourrait améliorer cette approche en réalisant plusieurs mesures indépendantes et en moyennant les variations dues au désordre. Toutefois, la modélisation montre que cette convergence a ses propres limites, notamment dans des configurations avec des contrastes forts, où un nombre important de moyennes serait nécessaire.

Pour aller plus loin, les simulations futures pourront intégrer des coefficients d'atténuation non nuls afin de mieux représenter les phénomènes physiques réels dans la dalle de béton. Ces simulations permettront d'affiner la compréhension des interactions entre les ondes et les hétérogénéités du matériau, tout en tenant compte de l'impact des pertes d'énergie dues à l'atténuation.

## 6 Conclusions et perspectives

En conclusion, nous avons développé des modèles numériques avec la plateforme de calcul open-source FEniCSx. Ces modèles permettent de résoudre les équations de propagation des ondes de Rayleigh dans le béton et le mortier. Nous avons ainsi répondu à certaines problématiques physiques soulevées par les mesures expérimentales.

Nous avons d'abord abordé la résolution de l'équation de Helmholtz dans le domaine fréquentiel, avec une fonction source à 30 kHz, en modélisant une dalle de mortier en deux dimensions. Cette approche visait à évaluer la capacité du modèle numérique à représenter le comportement des ondes. Nous pouvions alors analyser la réponse de la dalle soumise à un champ de pompe, vérifiant ainsi si le champ moyen de pompe était homogène. Les résultats ont montré que le gradient de déformation était plus homogène lorsque les ondes pompes n'étaient pas symétriques. Ils ont également permis d'écarter la première idée d'expérience de modulation acoustoélastique avec les ondes pompes et de la remplacer par une modulation thermique.

La suite du travail a consisté à examiner les effets de la non-linéarité du matériau à l'aide d'un modèle quasi-linéaire et de granulats modélisés par des ellipses, ce qui a permis de rapprocher les résultats de la modélisation des conditions expérimentales réelles.

Les perspectives de travail se concentrent sur trois axes principaux. Premièrement, la résolution du modèle pompes/sondes dans le béton avec la modélisation des granulats. Ce qui offrirait un calcul de la déformation plus précis.

Deuxièmement, le passage à une modélisation en trois dimensions permettant de mieux représenter les conditions expérimentales réelles et d'étudier plus en détail les effets des hétérogénéités comme les fissures et les granulats. Cette extension nécessitera une réévaluation des paramètres de maillage et l'utilisation de la parallélisation via MPI pour gérer efficacement le volume de données et les calculs.

Enfin, la résolution en domaine temporel permettant de modéliser des lois non linéaires plus complexes, comme celles à seuil, qui sont difficiles à traiter en quasi-linéaire.

## 7 Annexes

### 7.1 Définitions, propriétés et théorèmes vers la résolution de l'équation de Helmholtz [11]

#### 7.1.1 Rappels sur les espaces $L^p$ , $W^{1,p}$ et espaces duaux

**Définition 1** (Espace  $L^1$ ). L'espace  $L^1(\Omega)$  est l'espace des fonctions définies sur  $\Omega$  à valeurs réelles dont la valeur absolue est intégrable au sens de Lebesgue.

**Définition 2** (Espaces  $L^p$ ). Soit  $p \in \mathbb{R}$  tel que  $1 < p < \infty$ ; on définit

$$L^p(\Omega) = \{f : \Omega \rightarrow \mathbb{R}; f \text{ est mesurable et } |f|^p \in L^1(\Omega)\}$$

avec pour norme

$$\|f\|_{L^p} = \left( \int_{\Omega} |f(x)|^p d\mu \right)^{\frac{1}{p}}.$$

**Définition 3** (Espaces de Sobolev  $W^{1,p}$ ). L'espace de Sobolev  $W^{1,p}(\Omega)$  est défini par

$$W^{1,p}(\Omega) = \{u \in L^p(\Omega) \mid \nabla u \in L^p(\Omega)\}$$

où  $\nabla u$  est le gradient au sens faible et avec

$$\|f\|_{W^{1,p}} = (\|f\|_{L^p}^p + \|\nabla f\|_{L^p}^p)^{\frac{1}{p}}.$$

De plus, on définit  $H^1 = W^{1,2}$  ainsi que  $W_0^{1,p} = \{u \in W^{1,p}(\Omega) \mid u = 0 \text{ sur } \partial\Omega\}$ .

**Définition 4** (Espace dual). L'espace dual  $E'$  de  $E$  est l'espace vectoriel formé des formes linéaires continues sur  $E$ .

Par exemple  $W^{-1,p'}(\Omega) = (W_0^{1,p})'$  avec  $1 = \frac{1}{p} + \frac{1}{p'}$ .

**Propriété 1.** Pour tout  $\sigma \in L^{p'}(\Omega)$ , on a  $\operatorname{div}(\sigma) \in W^{-1,p'}(\Omega)$ .

*Démonstration.*

$$\begin{aligned} \text{Soit } v \in W_0^{1,p}, \text{ on a } \langle \operatorname{div}(\sigma), v \rangle &= -\langle \sigma, \nabla v \rangle, \text{ d'après la formule des dérivées distributionnelles} \\ &= - \int_{\Omega} (\sigma : \nabla v) dx \\ &\leq \|\sigma\|_{L^{p'}} \|\nabla v\|_{L^p} \text{ par l'inégalité de Hölder} \\ &< \infty \text{ car } \sigma \in L^{p'} \text{ et } \nabla v \in L^p \\ \text{ainsi } \operatorname{div}(\sigma) &\in W^{-1,p'}. \end{aligned}$$

□

**Propriété 2** (Formule de Green). Soient  $w, v$  des fonctions  $\mathcal{C}^1(\Omega)$  à valeurs vectorielles, on a :

$$\int_{\Omega} \operatorname{div}(w)v dx = \int_{\partial\Omega} (w \cdot n) v d\mu - \int_{\Omega} w \cdot \nabla v dx,$$

où  $n$  est le vecteur normal extérieur à  $\partial\Omega$ .

De plus, pour des champs tensoriels  $\sigma$  de classe  $\mathcal{C}^1(\Omega)$  à valeurs dans les tenseurs d'ordre 2, on a :

$$\int_{\Omega} \nabla \cdot (\sigma v) dx = \int_{\partial\Omega} (\sigma v) \cdot n d\mu - \int_{\Omega} \sigma : \nabla v dx,$$

où  $\sigma : \nabla v$  désigne le produit contracté entre le tenseur  $\sigma$  et le gradient de  $v$ , et  $n$  est toujours le vecteur normal à  $\partial\Omega$ .

### 7.1.2 Existence et unicité d'une solution faible [14]

Soit  $V$  un espace de Hilbert réel ou complexe.

**Théorème 1** (Théorème de représentation de Riesz-Fréchet). Soit  $V$  un espace de Hilbert réel ou complexe muni de son produit scalaire  $\langle \cdot, \cdot \rangle$  et soit  $l \in V'$  une forme linéaire continue sur  $V$ . Alors il existe un unique  $y$  dans  $V$  tel que pour tout  $x$  de  $V$  on ait :  $l(x) = \langle y, x \rangle$ .

C'est-à-dire :

$$\exists! y \in V, \forall x \in V, l(x) = \langle y, x \rangle.$$

**Remarque :** Ce théorème s'étend aux formes bilinéaires en considérant une unique application  $A$  de  $V$  dans  $V$  et telle que, si  $a$  est une forme bilinéaire continue sur un espace de Hilbert  $V$ , et que pour tout  $(u, v) \in V \times V$  on ait  $a(u, v) = \langle Au, v \rangle$ . L'application  $A$  est linéaire et continue, de norme égale à celle de  $a$ . Ainsi :

$$\exists! A \in \mathcal{L}(V), \forall (u, v) \in V \times V, a(u, v) = \langle Au, v \rangle.$$

**Définition 5. Continuité d'une forme bilinéaire.** Une forme bilinéaire  $a$  est dite continue si elle vérifie :

$$\exists c > 0, \forall (u, v) \in V^2, |a(u, v)| \leq c \|u\|_V \|v\|_V.$$

**Définition 6. Coercivité d'une forme bilinéaire.** Une forme bilinéaire  $a$  est dite coercive si elle vérifie :

$$\exists \alpha > 0, \forall u \in V, a(u, u) \geq \alpha \|u\|_V^2.$$

**Théorème 2** (Théorème de Lax-Milgram). Soit  $V$  un espace de Hilbert muni d'un produit scalaire et d'une norme associée. Soit  $a(\cdot, \cdot)$  une forme bilinéaire, continue et coercive sur  $V \times V$  et soit  $l(\cdot)$  une application linéaire et continue sur  $V$ .

Alors le problème variationnel de la forme :

trouver  $u \in V$ , tel que  $a(u, v) = l(v)$ ,  $\forall v \in V$ , admet une unique solution.

*Démonstration.* La continuité et la coercivité de  $a$ , la forme bilinéaire, permettent de dire que  $a$  définit un produit scalaire  $(\cdot, \cdot)_a$  dont la norme associée  $\|\cdot\|_a$  est équivalente à la norme sur l'espace de Hilbert  $V$ . Le problème devient alors : Trouver  $u$  tel que  $\forall v \in V, (u, v)_a = l(v)$ .

Or  $l(v) = \langle l, v \rangle$  le produit de dualité entre  $V$  et  $V'$ . Le problème devient ainsi : trouver  $u$  tel que  $\forall v \in V, (u, v)_a = \langle l, v \rangle$ .

Le théorème de représentation de Riesz-Fréchet permet ensuite de donner l'existence et l'unicité.  $\square$

## 7.2 Rappels sur les éléments finis

La méthode des éléments finis se restreint à des approximations continues polynomiales par morceaux. On choisit une famille de parties compactes connexes pour recouvrir le domaine d'étude ouvert borné  $\Omega$ . On peut choisir par exemple des triangles ou des quadrangles en dimension 2 (création d'un maillage). On comparera les deux par la suite.

Soit  $K \subset \mathbb{R}^n$  une partie compacte, connexe et d'intérieur non vide. Soit  $\mathbb{P}$  un espace vectoriel de fonctions polynomiales de  $K$  dans  $\mathbb{R}$ . Soit  $\Sigma_K = \{\sigma_i : \mathbb{P} \rightarrow \mathbb{R}, 1 \leq i \leq M\}$  une famille de  $M$  formes linéaires sur  $\mathbb{P}$ , souvent appelées : degrés de liberté.

**Définition 7. Unisolvance.** Soit  $A = \{a_1, \dots, a_N\}$  un ensemble de  $N$  points distincts de  $\mathbb{R}^n$ , soit  $\mathbb{P}$  un espace vectoriel de dimension finie de fonctions de  $\mathbb{R}^n$  à valeurs dans  $\mathbb{R}$ . On dit que  $A$  est  $\mathbb{P}$ -unisolvant si et seulement si pour tous les réels  $\alpha_1, \dots, \alpha_N$ , il existe un unique élément  $p$  de  $\mathbb{P}$  tel que  $\forall i = 1, \dots, N, p(a_i) = \alpha_i$ . Cela veut dire que la fonction de  $\mathbb{P}$  dans  $\mathbb{R}^N$  qui à  $p$  fait correspondre  $(p(a_1), \dots, p(a_N)) = (\alpha_1, \dots, \alpha_N)$  est bijective.

**Définition 8.** On dit qu'un triplet  $(K, \mathbb{P}, \Sigma_K)$  est un **élément fini** si  $\Sigma_K$  est  $\mathbb{P}$ -unisolvante.

**Propriété 3.** Tout élément fini  $(K, \mathbb{P}, \Sigma_K)$  admet une base canonique. De plus, si  $\dim(\mathbb{P}) = \text{card}(\Sigma_K)$  et que le triplet  $(K, \mathbb{P}, \Sigma_K)$  admet une base canonique, alors c'est un élément fini.

*Démonstration.* Commençons par prouver que tout élément fini  $(K, \mathbb{P}, \Sigma_K)$  admet une base canonique. Soit  $\{\phi_i\}_{i=1}^{\dim(\mathbb{P})}$  une base de  $\mathbb{P}$ . Pour chaque degré de liberté  $\sigma_i \in \Sigma_K$ , on définit une fonction  $\lambda_i \in \mathbb{P}$  telle que :

$$\sigma_i(\lambda_j) = \delta_{ij}, \quad \text{pour tout } i, j,$$

où  $\delta_{ij}$  est le symbole de Kronecker. Cela signifie que chaque  $\lambda_i$  satisfait uniquement la condition  $\sigma_i$  tout en annulant les autres degrés de liberté. Ainsi, l'ensemble des fonctions  $\{\lambda_i\}_{i=1}^{\dim(\mathbb{P})}$  forme une base canonique de  $\mathbb{P}$ .

Ensuite, montrons que si  $\dim(\mathbb{P}) = \text{card}(\Sigma_K)$  et que le triplet admet une base canonique, alors  $(K, \mathbb{P}, \Sigma_K)$  est un élément fini.

Étant donné que  $\dim(\mathbb{P}) = \text{card}(\Sigma_K)$ , les degrés de liberté dans  $\Sigma_K$  définissent un système de contraintes linéaires indépendantes sur les fonctions de  $\mathbb{P}$ . Ainsi, chaque fonction de  $\mathbb{P}$  est entièrement déterminée par les valeurs prises sur les degrés de liberté définis par  $\Sigma_K$ .

Par conséquent, les conditions imposées par  $\Sigma_K$  forment un système complet qui caractérise de manière unique chaque élément de  $\mathbb{P}$ . Cela correspond à la définition de l'unisolvance, c'est-à-dire que  $(K, \mathbb{P}, \Sigma_K)$  est un élément fini.  $\square$

**Définition 9.** Un élément fini  $(K, \mathbb{P}, \Sigma_K)$  est dit **de Lagrange** si tous les degrés de liberté ( $\sigma_i$ ) sont de la forme  $\sigma_i(p) = p(s_i)$ ,  $s_i \in K, \forall i$ . On notera  $Q_k$  l'espace vectoriel des polynômes de degré inférieur ou égal à  $k$  par rapport à chaque variable. Un élément fini de Lagrange est dit de **degré**  $k$  pour  $\mathbb{P} = Q_k$ .

La méthode des éléments finis découle de la méthode de Galerkin. L'idée est de partir du problème variationnel et d'utiliser le fait que  $V$  est un espace de Hilbert séparable. Il admet alors une base hilbertienne que l'on peut tronquer de sorte à créer une base de dimension  $M$  engendrée par les  $M$  premières fonctions de base de l'espace  $V$ . On définit alors  $V_M$  l'espace engendré par cette base :  $V_M = \text{Vect}(v_1, \dots, v_M)$ .

Ainsi le problème variationnel (3) peut être approché par : Trouver  $u_h \in V_M$ ,

$$a(u_h, v_h) = L(v_h), \quad \forall v_h \in V_M, \quad (16)$$

ce qui peut s'écrire sous la forme d'un système linéaire de  $M$  équations  $AU=L$  avec :

$$A = \begin{bmatrix} a(v_1, v_1) & a(v_2, v_1) & \cdots & a(v_M, v_1) \\ a(v_1, v_2) & a(v_2, v_2) & \cdots & a(v_M, v_2) \\ \vdots & \vdots & \ddots & \vdots \\ a(v_1, v_M) & a(v_2, v_M) & \cdots & a(v_M, v_M) \end{bmatrix}, \quad L = \begin{bmatrix} l(v_1) \\ l(v_2) \\ \vdots \\ l(v_M) \end{bmatrix}$$

et notre inconnue

$$U = \begin{bmatrix} u_{h,1} \\ \vdots \\ u_{h,M} \end{bmatrix}.$$

On résout ensuite le système linéaire grâce aux solveurs disponibles dans la bibliothèque PetsC. Ils sont basés sur des méthodes de sous-espaces de Krylov (KSP) telles que la méthode du Gradient Conjugué et la méthode du Résidu minimal généralisé. Chaque solveur KSP est configuré avec des options spécifiques qui contrôlent son comportement et sa performance, comme le nombre maximum d'itérations, la tolérance de convergence, le préconditionneur à utiliser, etc.

Nous expliquons dans la section 7.3 les 2 solveurs que nous comparons dans le programme et dans la section 7.4 nous expliquons quels préconditionneurs sont utilisés.

## 7.3 Résolution numérique du système linéaire $AU=L$

### 1. La méthode du gradient conjugué

Cette méthode est basée sur la méthode du gradient à pas optimal et permet de résoudre les systèmes linéaires de la forme  $Au = b$  dont la matrice  $A$  est symétrique définie positive. La méthode à pas optimal consiste à partir d'un vecteur donné  $u^0$  et à déterminer à chaque étape un vecteur de descente  $p^k$  et un pas de descente  $r^k$ , qui sera optimal, permettant de calculer  $u^{k+1}$  à partir de  $u^k$  par l'égalité :

$$u^{k+1} = u^k + r^k p^k$$

où  $r^k$  est optimal, c'est-à-dire qu'il change à chaque itération  $k$  et  $p^k = \nabla J(u^k)$  avec  $J(u) = \frac{1}{2}(Au, u) - (b, u)$  la fonctionnelle de minimisation. Le choix de  $p^k$  est dû au fait que  $\nabla J(u) = Au - b$  et donc résoudre le système est équivalent à trouver la solution au problème  $J(v) = \min_{u \in V} J(u)$ .

**Théorème 3.** Si  $A \in M_n(\mathbb{R})$  est symétrique définie positive, alors la méthode du gradient à pas optimal converge. C'est-à-dire, que la suite générée par l'algorithme converge, pour tout choix de  $u^0$ , vers l'unique solution du système linéaire.

La particularité de la méthode du gradient conjugué est de prendre comme première direction de descente la direction  $p^0 = \nabla J(u^0)$  et de choisir à l'étape  $k+1$  une direction  $p^{k+1}$  telle que  $(p^{k+1}, Ap^k) = 0$  et un pas optimal  $r^k$  défini par

$$\forall t \in \mathbb{R}, J(u^k + r^k p^k) \leq J(u^k + t p^k).$$

Les directions de descente sont donc A-conjuguées ou orthogonales au sens du produit scalaire défini par la matrice symétrique définie positive A, ce qui donne le nom à la méthode.

**Propriété 4.** Pour  $u^k$  donné et pour tout choix de  $p^k \neq 0$ , le paramètre optimal  $r^k$  a pour expression :

$$r^k = -\frac{(g^k, p^k)}{(Ap^k, p^k)}$$

où  $g^k = Au^k - b$  est le résidu à l'itération k et on a les deux relations suivantes :

$$\forall k \in \mathbb{N}, g^{k+1} = g^k + r^k Ap^k,$$

$$\forall k \in \mathbb{N}, (p^k, g^{k+1}) = 0.$$

*Démonstration.* L'expression de  $r^k$  découle de l'algorithme de la méthode à pas optimal. De plus :

$$g^{k+1} = Au^{k+1} - b = A(u^k + r^k p^k) - b = g^k + r^k Ap^k,$$

et

$$(p^k, g^{k+1}) = (p^k, g^k) + r^k (Ap^k, p^k) = (p^k, g^k) - \frac{(p^k, g^k)}{(Ap^k, p^k)} (Ap^k, p^k) = 0.$$

□

Ainsi, l'algorithme du gradient conjugué est initialisé en choisissant un vecteur  $u^0$ . Si  $g^0 \neq 0$  on effectue :

$$\begin{cases} g^0 = p^0, \\ r^0 = -\frac{(g^0, g^0)}{(Ag^0, g^0)}, \\ u^1 = u^0 + r^0 g^0, \end{cases} \quad (17)$$

et pour  $k \geq 1$ , on a :

$$\begin{cases} g^k = Au^k - b, \\ \alpha^k = -\frac{(Ag^k, p^{k-1})}{(Ap^{k-1}, p^{k-1})}, \\ p^k = g^k + \alpha^k p^{k-1}, \\ r^k = -\frac{(g^k, p^k)}{(Ap^k, p^k)}, \\ u^{k+1} = u^k + r^k p^k. \end{cases} \quad (18)$$

Si  $g^k = 0$ , on arrête les calculs.

Dans la pratique, le critère d'arrêt tient compte de la précision de l'ordinateur et l'arrêt se fait si :

$$\frac{\|g^k\|}{\|b\|} < \varepsilon$$

pour un  $\varepsilon$  choisi. Quand cette relation est vérifiée, on a :

$$\frac{\|u^k - A^{-1}b\|}{\|u\|} < \varepsilon \text{Cond}(A)$$

où  $u$  est la solution et  $\text{Cond}(A) = \|A\| \|A^{-1}\|$  le conditionnement de la matrice A.

Dans notre étude numérique on utilise la factorisation LU en guise de préconditionnement.

- 2. La méthode GMRES** Cette méthode permet de résoudre les systèmes linéaires où A n'est pas nécessairement symétrique. La base de cette méthode est la minimisation des résidus acquis de façon successive. Elle est basée sur l'algorithme d'Arnoldi qui utilise la méthode de Gram-Schmidt pour calculer une base orthonormée  $(v_1, v_2, \dots, v_k)$  du sous-espace de Krylov  $K_k = \text{Vect}(u^1, Au^1, \dots, A^{k-1}u^1)$ .

Alors on obtient la matrice de Hessenberg  $\tilde{H}_k = (u^k)^T A u^k$ , en lui ajoutant une ligne supplémentaire dont le seul élément non nul est  $h_{k+1,k}$  à la position  $(k+1,k)$ . On obtient alors le système suivant :

$$A u^k = u^{k+1} H_k.$$

L'algorithme est le suivant :

après avoir choisi la donnée initiale  $u^0$  et calculé le premier résidu  $r^0 = b - A u^0$  ainsi que  $v_1 = \frac{r_0}{\|r_0\|}$ , on calcule :

pour  $j = 1, \dots, M$

$$\begin{cases} h_{i,j} = (A v_j, v_i), & i = 1, \dots, j \\ \tilde{v}_{j+1} = A v_j - \sum_{i=1}^j h_{i,j} v_i \\ h_{j+1,j} = \|\tilde{v}_{j+1}\|_2 \\ v_{j+1} = \frac{\tilde{v}_{j+1}}{h_{j+1,j}} \end{cases}$$

On établit ensuite la solution approchée de l'équation

$$u^M = u^0 + V^M y^M$$

où  $y^M$  minimise  $\|(b - A u^0) e_1 - H_M y\|_2$  avec  $e_1$  le premier vecteur de la base canonique et le terme  $V^M$  représente une matrice dont les colonnes sont les vecteurs  $(v_1, v_2, \dots, v_M)$ .

On calcule ensuite  $r^M = b - A u^M$  et si le résidu est assez petit on arrête, sinon on réinitialise  $x^0 = x^M$ ,  $r^0 = r^M$  et on recommence.

L'option de solveur PetsC par défaut utilise cette méthode accompagnée d'un préconditionnement utilisant la factorisation LU Incomplète.

## 7.4 Méthodes de Préconditionnement : LU et LU Incomplet

La méthode de préconditionnement LU est une technique utilisée pour améliorer la convergence des méthodes itératives pour la résolution de systèmes linéaires de grande taille. Le préconditionnement vise à transformer le système linéaire original en un système équivalent mais plus facile à résoudre par des méthodes itératives.

Soit le système linéaire

$$A x = b,$$

où  $A$  est une matrice carrée  $n \times n$ ,  $x$  est le vecteur des inconnues et  $b$  est le vecteur des termes constants. La décomposition LU de  $A$  permet d'écrire  $A$  comme le produit de deux matrices :

$$A = L U,$$

où  $L$  est une matrice triangulaire inférieure et  $U$  est une matrice triangulaire supérieure. En utilisant cette décomposition, nous pouvons reformuler le système initial comme suit :

$$L U x = b.$$

Nous pouvons résoudre ce système en deux étapes : résoudre  $L y = b$  pour  $y$ , puis résoudre  $U x = y$  pour  $x$ . Cette méthode est efficace lorsque la décomposition LU peut être obtenue rapidement et que la résolution des systèmes triangulaires est peu coûteuse.

Le préconditionnement LU incomplet (ou ILU pour Incomplete LU) est une variante de la méthode LU qui cherche à conserver la structure de la matrice tout en simplifiant la décomposition LU. Au lieu de calculer la décomposition LU exacte, ILU approximative la décomposition en conservant une partie de la structure des matrices  $L$  et  $U$ . Cette méthode est particulièrement utile pour les grandes matrices creuses, où une décomposition complète LU serait trop coûteuse en termes de mémoire et de temps. La décomposition ILU peut être exprimée comme :

$$A \approx L_{ILU} U_{ILU},$$

où  $L_{ILU}$  et  $U_{ILU}$  sont des approximations creuses de  $L$  et  $U$ , respectivement. La décomposition ILU est obtenue en introduisant des seuils ou en supprimant certains éléments de  $L$  et  $U$ , ce qui réduit la densité des matrices tout en essayant de conserver une approximation utile de la décomposition LU.



Le préconditionnement ILU est généralement réalisé en suivant ces étapes :

1. Initialiser  $L_{ILU}$  et  $U_{ILU}$  comme des matrices creuses, souvent en prenant  $L_{ILU}$  comme une matrice triangulaire inférieure avec des éléments unitaires sur la diagonale et  $U_{ILU}$  comme une matrice triangulaire supérieure.
2. Effectuer une décomposition LU partielle sur  $A$  tout en imposant des conditions de structure creuse.
3. Répéter les étapes jusqu'à ce que la décomposition converge à une approximation satisfaisante.

Les préconditionneurs ILU sont utilisés pour accélérer la convergence des méthodes itératives telles que le gradient conjugué ou GMRES. La clé est de choisir une approximation suffisamment précise pour améliorer la condition du système tout en restant computationalement efficace.

Ainsi, la méthode de préconditionnement ILU offre une approximation plus rapide et plus économique en termes de mémoire pour les grandes matrices creuses, mais avec une perte de précision par rapport à la décomposition LU exacte.

## 7.5 Système quasi-linéaire

**Equation de propagation des ondes :**

$$\rho \partial_t^2 u - \operatorname{div} \sigma = f \quad (19)$$

avec  $\sigma$  donnée par l'équation (1) et  $f$  un terme source donné. En réalité, on travaille ici avec le premier tenseur de Piola-Kirchhoff (coordonnées lagrangiennes), qu'il conviendrait de noter  $P$ ; mais on ne va pas multiplier les notations.

On suppose l'égalité suivante :  $u = u_1 + u_2$ , avec  $u_1$  "grand, mais pas trop", et  $u_2$  "petit devant  $u_1$ ". On développe  $\sigma$  :

$$\sigma = E(\varepsilon_1 + \varepsilon_2 + 2\beta\varepsilon_1\varepsilon_2 + \beta\varepsilon_1^2 + \beta\varepsilon_2^2). \quad (20)$$

**Début de l'approximation quasi-linéaire :** On écrit d'abord l'équation (20) en ne gardant que les termes dominants  $u \approx u_1$  et  $\sigma \approx \sigma_{\text{lin}}(\varepsilon_1) = E\varepsilon_1$  :

$$\rho \partial_t^2 u_1 - \operatorname{div} \sigma_{\text{lin}}(u_1) = f. \quad (21)$$

L'équation (21) permet de définir  $u_1$  : on suppose à présent que l'on connaît sa solution. Puis, on revient à l'équation (20) en ne gardant que les termes dominants pour  $\sigma \approx \sigma_{\text{lin}}(u_1 + u_2) + E\beta\varepsilon_1^2$ . En substituant l'équation (21), on obtient :

$$\rho \partial_t^2 u_2 - \operatorname{div} \sigma_{\text{lin}}(u_2) = f_2(u_1) \quad (22)$$

avec  $f_2 = E\beta \operatorname{div} \varepsilon_1^2$ . Dans le régime quasi-linéaire, le champ fondamental  $u_1$  vérifie l'équation d'onde linéaire, et le champ  $u_2$  vérifie l'équation d'onde linéaire avec un terme source dépendant de  $u_1$ .

Les équations sont linéaires : on peut passer en fréquentiel (équation de Helmholtz).

## Références

- [1] Ciment romain, [https://fr.wikipedia.org/wiki/Ciment\\_romain](https://fr.wikipedia.org/wiki/Ciment_romain)
- [2] La filière ciment-béton : un mur face à la transition ? <https://reseauactionclimat.org/la-filiere-ciment-beton-un-mur-face-a-la-transition/>, réseau action climat France
- [3] FEniCS, <https://fenicsproject.org/>
- [4] Anders Logg, Kent-Andre Mardal, Garth N. Wells, Automated Solution of Differential Equations by the Finite Element Method, The FEniCS Book, <http://launchpad.net/fenics-book/trunk/final/+download/fenics-book-2011-10-27-final.pdf>, 2012
- [5] Jørgen S. Dokken, The FEniCSx tutorial, <https://jsdokken.com/dolfinx-tutorial/>, Last updated on Sep 09, 2024.
- [6] FEniCS Project 2024, FEniCSx Documentation, <https://docs.fenicsproject.org/>
- [7] Jeremy Bleyer, Numerical Tours of Computational Mechanics with FEniCSx, <https://bleyerj.github.io/comet-fenicsx/>
- [8] Igor A. Baratta, Joseph P. Dean, Jørgen S. Dokken, Michal Habera, Jack S. Hale, Chris N. Richardson, Marie E. Rognes, Matthew W. Scroggs, Nathan Sime, Garth N. Wells DOLFINx : The next generation FEniCS problem solving environment, <https://docs.fenicsproject.org/dolfinx/v0.8.0/python/>
- [9] FEniCS Project, UFL : Unified Form Language, <https://docs.fenicsproject.org/ufl/2024.1.0.post1/index.html>
- [10] David Torello, Sebastian Thiele, Kathryn H. Matlack, Jin-Yeon Kim, Jianmin Qu, Laurence J. Jacobs, Diffraction, attenuation, and source corrections for nonlinear Rayleigh wave ultrasonic measurements, Ultrasonics, 2014
- [11] Haim Brezis, Functional Analysis, Sobolev Spaces and Partial Differential Equations, Rutgers University, Springer 2010.
- [12] Patrick Joly, Introduction à l'analyse mathématique de la propagation d'ondes en régime harmonique, Université Pierre et Marie Curie, 2006
- [13] Anaïs Crestetto, Modélisation pour les fluides incompressibles, Nantes Université, cours MACS 2024
- [14] Vincent Manet, Méthode des éléments finis, 2013
- [15] D. J. Shull, M. F. Hamilton, Yu. A. Il'insky, E. A. Zabolotskaya, Harmonic generation in plane and cylindrical nonlinear Rayleigh waves, Acoustical Society of America, 1993.
- [16] S. Zhang, X. Li, H. Jeong, H. Hu, Modeling nonlinear Rayleigh wave fields generated by angle beam wedge transducers - A theoretical study, Wave Motion, 2016.