UNIVERSIDADE FEDERAL FLUMINENSE

VICTOR ABI-RAMIA ANTONIO RACHIDE

A pixel-based finite element implementation to estimate effective wave velocity in heterogeneous media

NITERÓI

<2023>

UNIVERSIDADE FEDERAL FLUMINENSE

VICTOR ABI-RAMIA ANTONIO RACHIDE

A pixel-based finite element implementation to estimate effective wave velocity in heterogeneous media

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Computação da Universidade Federal Fluminense como requisito parcial para a obtenção do Grau de Mestre em Computação. Área de concentração: Ciência da Computação

Orientador:
RICARDO LEIDERMAN

Co-orientador:
ANDRE MAUÉS BRABO PEREIRA

NITERÓI

<2023>

Ficha catalográfica automática - SDC/BEE
Gerada com informações fornecidas pelo autor

R119p Rachide, Victor Abi-Ramia Antonio
A pixel-based finite element implementation to estimate effective wave velocity in heterogeneous media / Victor Abi-Ramia Antonio Rachide. - 2023.
58 f.

Orientador: RICARDO LEIDERMAN.
Coorientador: ANDRE PEREIRA.
Dissertação (mestrado)-Universidade Federal Fluminense, Instituto de Computação, Niterói, 2023.

1. Método dos elementos finitos. 2. Propagação de ondas. 3. Produção intelectual. I. LEIDERMAN, RICARDO, orientador. II. PEREIRA, ANDRE, coorientador. III. Universidade Federal Fluminense. Instituto de Computação.IV. Título.

CDD - XXX

Bibliotecário responsável: Debora do Nascimento - CRB7/6368

Victor Abi-Ramia Antonio Rachide

A pixel-based finite element implementation to estimate effective wave velocity in heterogeneous media

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Computação da Universidade Federal Fluminense como requisito parcial para a obtenção do Grau de Mestre em Computação. Área de concentração: Ciência da Computação

Aprovada em novembro de 2023.

BANCA EXAMINADORA

Prof. Dr. Ricardo Leiderman - Orientador, UFF

Prof. Dr. Andre Maues Brabo Pereira, - Coorientador, UFF

Prof. Dr. Anselmo Antunes Montenegro, UFF

Prof. Dr. Daniel Alves Castello, UFRJ

Niterói

2023

Resumo

Estimamos as propriedades elásticas de amostras digitais de rochas através de simulações de elementos finitos de experimentos de transmissão de propagação de ondas. Usamos uma estratégia na qual não montamos matriz de rigidez, explorando a malha estruturada proveniente de imagens, diminuindo a memória necessária para armazenar os cálculos. Além disso, empregamos uma técnica de matriz diagonal de massa concentrada para superar as dificuldades relacionadas à inversão e armazenamento da matriz de massa. Nós adotamos a integração Leapfrog, que é conhecida por sua combinação de estabilidade, precisão e eficiência. Testamos nossa implementação em relação às previsões teóricas e resultados analíticos obtidos com o auxílio do método clássico de teoria de propagação de ondas. Por fim, simulamos o teste de transmissão em um modelo digital obtido através de uma microtomografia e comparamos as propriedades elásticas obtidas com os resultados de homogeneização estática.

Palavras-chave:Propriedade efetiva, Caracterização, MEF, Onda Elástica.

Abstract

We estimate the elastic properties of digital rock samples through finite element simulations of wave propagation transmission experiment. We use an assembly-free approach for the stiffness matrix that explores the pixel-based structured mesh to decrease the memory required to store the computations. Further, we employ a diagonal Lumped-Mass matrix technique to overcome the difficulties related to the inversion and storage of the mass matrix. We adopt the Leapfrog integration, which is known for its combination of stability, precision, and efficiency. We have tested our implementation against theoretical predictions and analytical results obtained with the aid of the classical wave propagation theory. Lastly, we simulate the transmission test in a digital model attained via a micro-tomography and compare the results with the corresponding counterpart acquired via static finite element homogenization formulation.

Keywords: Effective property, Characterization, FEM, Elastic Wave.

List of Figures

List of Tables

Contents

Chapter 1

Introduction

The effective medium theory entails procedures that describes heterogeneous medium as an equivalent homogeneous medium. Those equivalent medium properties are called effective properties. Their importance embraces a large scope of areas since they link microstructure and the overall behavior of heterogeneous media. The effective elastic velocity, for example, is related to grain size, geometrical and elastic properties of the material's microconstituintes. This property can be obtained experimentally via laboratory ultrasonic tests as well as, in the geophysical context, from the processing of seismic data. Accordingly, this effective property plays an important role in applications such as geophysics (oil and water reservoir characterization and exploration and seismology)(UYANIK et al., 2019), metamaterials (SRIVASTAVA, 2015), and composites (ZIMMER; COST, 1970). In the case of geophysics, elastic velocities are used to identify and estimate petrophysical properties of rock formations such as porosity, saturation, grain densities, and subsurface fluid motion (4D seismic), for example (UYANIK et al., 2019). In laboratory ultrasonic tests small samples of heterogeneous material are employed. The equipment used in such test are expansive, the preparation of the samples themselves can be equally expansive and arduous task, as, for example, the saturation and desaturation of rock samples that can take weeks to take place, and the application of either unusual boundary conditions or extreme service conditions, as high oil reservoir pressure and temperature. In that sense, parallel to the laboratory tests, complementary analytical and numerical methods to evaluate the effective elastic wave velocity have been developed and exploited by the Industry as well as the Academy.

The pioneering analytical models focused on approximating effective elastic constants and elastic velocities of composites were provided by Voigt (VOIGT, 1889) and Reuss (REUSS, 1929). Those models use volume fractions and different assumptions to set the

upper and lower boundary for the effective property value. Nowadays, those bounds are used more in a framework for validating and evaluating approximations obtained via more sophisticated methods. Later, the Hashin-Shtrikman bounds (HASHIN; SHTRIKMAN, 1963) were proposed using variational approach, which are tighter bounds compared to Voigt and Reuss approximations.

Multiple static homogenization schemes were proposed in order to approximate effective properties for elastic porous media and composites based on assumptions of the geometries of inclusions (HILL, 1965) (ESHELBY, 1957)(MORI; TANAKA, 1973). For a review and comparison of those schemes, please, refere to (MILED; SAB; LE ROY, 2011). Technics of elastodynamic homogenization for composites were developed as well. For example, an asymptotic elastodynamic homogenization were presented in (BOUTIN; AURIAULT, 1993) and (AURIAULT; BOUTIN, 2012). Other authors, in turn, used another theory, known as Willis exact elastodynamic homogenization theory to estimate the effective properties (WILLIS, 2009)(WILLIS, 2011)(WILLIS, 2012). (NASSAR; HE; AUFFRAY, 2016) proved that both theories are correlated. From a ramification of this approach, frequency-dependant dynamic constitutive relations of composites were developed in (NEMAT-NASSER; SRIVASTAVA, 2011). Further, the dynamic approach were also used to state the upper and lower bounds for these dynamic constitutive relations properties(NEMAT-NASSER; SRIVASTAVA, 2013). In a classical reference (BIOT, 1956), Biot developed a model for propagation of elastic waves in a fluid-saturated porous medium and its effective elastic wave velocity. In it's model, the effective elastic wave velocity is frequency-dependant characterizing a dispersive medium. For the low-frequency limit, Biot's theory and effective wave velocity becomes the Gaussmann theory and effective wave velocity, respectively. In addition, computational homogenization approaches were developed (MICHEL; MOULINEC; SUQUET, 1999)(GARBOCZI; DAY, 1995) for static and dynamic homogenization (PHAM; KOUZNETSOVA; GEERS, 2013) allowing the characterization of microstructures with more complex geometries.

The field of digital rock physics provides tools for obtaining EEP coupling microscopic imaging and numerical simulations to evaluate physical properties of rocks (ANDRÄ et al., 2013a; SAENGER, E.; KRÜGER, O.; SHAPIRO, S., 2004; LOPES, Pedro Cortez Fetter et al., 2022a; LOPES, P. C. et al., 2023; BEZ et al., 2023). Different approaches have been explored, as discussed in (SAENGER; ENZMANN et al., 2011; FAISAL et al., 2019; ANDRÄ et al., 2013b; SRIVASTAVA, 2015). In the literature, most authors employ either the Finite Element Method (FEM) for static homogenization (ANDRÄ et al., 2013b; ARNS et al., 2002) or the Finite Difference Method (FDM) for transient analyses in the

microstructure, where the computational models are usually straightforwardly provided by micro-computed tomography (μCT) (SAENGER; ENZMANN et al., 2011; SAENGER, Erik H; KRÜGER, Oliver S; SHAPIRO, Serge A, 2004; SAENGER; GOLD; SHAPIRO, 2000; SAENGER; CIZ et al., 2007; ZHU; SHAN et al., 2017).

Here, we propose an explicit finite element scheme (VAN PAMEL et al., 2017; FREHNER et al., 2008; ABDULLE; GROTE; JECKER, 2018) to simulate the wave propagation transmission experiment from (SAENGER, 2008; ZHU; SHAN, 2016) in order to estimate effective elastic properties of image-based models, such as synthetic images and rocks images. The effective elastic velocities are obtained with time required for a Gaussian planar pulse to travel through the heterogeneous microscopic domain. Our objective is to implement a memory efficient, parallel-computed, voxel-based 3D version of the method to integrate with the workflow of rocks digital characterization following the methodology of works such as (LOPES, Pedro Cortez Fetter et al., 2022a; LOPES, P. C. et al., 2023; BEZ et al., 2023; VIANNA et al., 2020; FERREIRA et al., 2023). On those works, our methodology was applied for different context such as heat transfer, elasticity, viscoelasticity, permeability and nuclear magnetic resonance. We aim to achieve massive computations of Gigavoxel images, with 10^9 degrees of freedom, utilizing domestic GPU in a reasonable time expenditure for different physical phenomena. In that regard, this present work is a continuation of those efforts.

In order to achieve this goal, we have developed and thoroughly tested a Leap-frog solver for 2D pixel-based FEM problems. As input, a neutral file describing simulation parameters and a binary file containing an 8-bit raster image are provided. The program then computes the transient response for the displacement field in the domain and keeps track of the displacement history at specific points or subdomains where receivers are placed. These are used to calculate the effective wave velocity of the medium. Our implementation employs the Diagonal Lumped-Mass strategy (WU, 2006) that allows to store the global mass matrix as a vector and an assembly-free approach for the global stiffness and mass matrix, so it is not stored in memory. Our implementation was coded in julia language, exploiting it's high performance throught vectorization.

This work is organized in five sections, including this introduction. In section 2, the admitted numerical modeling for the wave propagation problem is presented. In section 3, some key aspects of the proposed implementation and its features are addressed, including the lumped-mass formulation and the assembly-free approach. In section 4, numerical results are compared to analytical predictions, benchmark and static analysis. In section

5, our conclusions are presented.

Chapter 2

Numerical modeling

2.1 Governing Equations

A linear elastodynamics model is assumed to govern the wave propagation simulations. We consider that the following equations of motion in Cartesian coordinate from must hold true:

$$\sigma_{ij,j} = \rho \ddot{u}_i \quad \forall (x_i, t) \in \Omega \times]0, \tau], \tag{2.1}$$

where σ_{ij} is the stress tensor, ρ is the density of the material, and u_i is the displacement vector field. Ω is the spatial domain and τ is the final instant of the considered time interval. Notice that no body forces acting in Ω are considered and any influence of damping is disregarded. These equations are solved for the displacement field, which is defined as a function of space x_i and time t, i.e. $u_i = u_i(x_i, t)$. Hooke's law for linear elasticity is employed as the constitutive equation, as in

$$\sigma_{ij} = C_{ijkl} \varepsilon_{kl}, \tag{2.2}$$

where C_{ijkl} is the fourth-order stiffness tensor, and ε_{kl} is the strain tensor. Also, as usual, strain is defined in terms of the displacement field, via compatibility equations:

$$\varepsilon_{ij} = \frac{1}{2} \left(u_{i,j} + u_{j,i} \right). \tag{2.3}$$

Equations 2.2 and 2.3 are combined and rearranged to admit a matrix representation of the constitutive tensor, employing Voigt notation. In doing so, a constitutive law in terms of displacement is defined as

$$\boldsymbol{\sigma} = \mathbf{C} \overline{\boldsymbol{\nabla}} \boldsymbol{u}, \tag{2.4}$$

where $\boldsymbol{\sigma}$ is a pseudo-vector that represents the stress tensor such that $\boldsymbol{\sigma} \equiv \sigma_{ij}$, $\nabla \cdot \boldsymbol{\sigma} \equiv \sigma_{ij,j}$, $\boldsymbol{u} \equiv u_i$, and, in a two-dimensional space,

$$\mathbf{C}\overline{\boldsymbol{\nabla}}\boldsymbol{u} = \begin{bmatrix} C_{1111} & C_{1122} & \frac{1}{2}C_{1112} \\ & C_{2222} & \frac{1}{2}C_{2212} \\ \text{sym.} & & \frac{1}{2}C_{1212} \end{bmatrix} \begin{bmatrix} \partial/\partial x_1 & 0 \\ 0 & \partial/\partial x_2 \\ \partial/\partial x_2 & \partial/\partial x_1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}. \tag{2.5}$$

Merging equations 2.1 and 2.4 yields

$$\nabla \cdot \left(\mathbf{C}\overline{\boldsymbol{\nabla}}\boldsymbol{u}\right) = \rho\ddot{\boldsymbol{u}} \quad \forall (x_i, t) \in \Omega \times]0, \tau], \tag{2.6}$$

which is analogous to the Navier-Cauchy equations for elastodynamics, without body forces.

Equation 2.6 presents both boundary and initial value problems. In that sense, boundary conditions throughout the observed time interval and an initial state must be determined for a solution to be obtained. Respectively, these can be defined as

$$\begin{aligned} \sigma_{ij} n_j = t_i \quad & \forall (x_i, t) \in \Gamma_q \times [0, \tau], \\ u_i = u_i^{\Gamma} \quad & \forall (x_i, t) \in \Gamma_u \times [0, \tau], \end{aligned} \tag{2.7}$$

and

$$\begin{cases} u_i = u_i^0 \\ \dot{u}_i = \dot{u}_i^0 \end{cases} \quad \forall (x_i, 0) \in \Omega, \tag{2.8}$$

where q_i represents the surface traction and u_i^{Γ} depicts the prescribed displacement at the boundary $\Gamma = \partial\Omega$. Γ_q and Γ_u are segments of the boundary contained in Γ where t_i and u_i^{Γ} are known, respectively. n_j is the unit outward normal vector with respect to Γ. u_i^0 is the initial displacement field and $\dot{u}_i^0$ is the initial velocity field, both assumed to be known in all of Ω at $t = 0$.

A solution for equation 2.6 can be approximated via numerical methods. Here, we employ the FEM to solve the partial differential equations in a discretized spatial domain. An explicit Leap-frog integration scheme is adopted to march in the time domain. The application of these methods is explained over the next two sections.

2.2 Finite Element Method

In order to solve our governing equations, we opted to use finite element method (FEM) to approximate the spacial derivatives in our governing equation. The FEM is known for transforming differential equations associated with boundary value problems into systems of algebraic equations. When dealing with transient phenomena, it usually is employed to numerically resolve the spatial derivatives, leading to matrix ordinary differential equations over time.

In order to develop our FEM implementation, first, we need a weak formulation that can be derived from multiple strategies, we chose the weighted residual method. Then, since our goal is to simulate models obtained by images, our discretization strategy uses the pixels as elements, originating a structured mesh which decreases the computational cost of the element mapping and connectivity. Naturally, our elements are squares and we also adopted bilinear basis functions approximation for our field variables. With the weak formulation and the element shape function provided by the discretization step, we derive the element equations. Since our method is matrix-free, we avoid the assembly step usually employed in FEM. Also, we employ the lumped-mass matrix strategy (WU, 2006) in order to decrease the computational cost related to matrix inversion of the consistent mass matrix and it's assembly. The lumped-mass matrix is a diagonal matrix, then it's inversion is trivial compared to the consistent mass matrix option. After the FEM formulation, we are going to use finite differences to approximate the time derivatives contriving our Leap-frog explicit FEM scheme. Lastly, we study the convergence and error of the method. The latter is one of our main focus and is accounted for in chapter 4.

2.2.1 Weak formulation

The weak formulation is the integral equation used to formulate the finite element method. This formulation allows the approximation of the variable of interest in the governing equations by a chosen function, called trial functions. To derive the weak formulation, we opted to use the weighted residual method 2.10, which is an approximation technique that forces the integral of the multiplication of the residual function and a weight function to be zero. The weight function is an arbitrary function used to force whatever it multiplies to zero value. This feature will be explored in the development of the element equation. But this arbitrariness is not absolute. During the development of FEM some restrictions

are imposed for the convenience of the method and are going to be introduced through the text. As it is possible to induce by analysing equation 2.10, the weight function must present requirements about integrability such as boundedness and piece-wise continuity.

$$R_i = \sigma_{ij,j} - \rho\ddot{u}_i \tag{2.9}$$

$$\int_\Omega w_i R_i d\Omega = 0 \tag{2.10}$$

where R_i are the residual functions. Equation 2.10 is the expression of the weighted residual method, where w_n are the weighted functions. Substituting the equation 2.9 in 2.10 we get:

$$\int_\Omega w_i \rho\ddot{u}_i d\Omega = \int_\Omega w_i \sigma_{ij,j} d\Omega \tag{2.11}$$

Equation 2.11 is the weighted residual integral form of the governing equations. Our aim is to approximate the displacement function to solve this equation, when we analyse the constitutive equation 2.2 and compatibility equation 2.3, it is concluded that $\sigma_{ij,j}$ is proportional to $u_{j,jj}$. For this reason, any approximation for the displacement field of equation 2.11 must be twice differentiable, and this requirement excludes the bilinear approximation used in this text. Hence, we are going to use a method to decrease the degree of differentiability of $\sigma_{ij,j}$ by one, shifting the differential operator to the weight function, using the divergence theorem. Another reason for this choice is related with the resulting stiffness and consistent mass element matrices in the element equation formulation step. This way they will be symmetric, and this is important when solving the algebraic linear system thought numerical methods from computational linear algebra as symmetry is a condition for positive definiteness.

This process weakens the differentiability requirement of the approximation of the method, and that's what weak formulation nomenclature refers to. Also, this step imposes another requirement on the weight function, it must be now at least once differentiable.

To produce the necessary term for the divergence theorem, first we apply the chain rule 2.12, getting equation 2.13.

$$w_i \sigma_{ij,j} = (w_i \sigma_{ij})_{,j} - w_{i,j}\sigma_{ij} \tag{2.12}$$

$$\int_{\Omega} w_i \rho \ddot{u}_i d\Omega = \int_{\Omega} (w_i \sigma_{ij})_{,j} d\Omega - \int_{\Omega} w_{i.j} \sigma_{ij} d\Omega \tag{2.13}$$

Now we can apply the divergence theorem, and equation 2.13, becomes:

$$\int_{\Omega} w_i \rho \ddot{u}_i d\Omega + \int_{\Omega} w_{i,j} \sigma_{ij} d\Omega = \int_{\Gamma} w_i \sigma_{ij} n_j d\Gamma \tag{2.14}$$

The product $\sigma_{ij} n_j$ in equation 2.14 is the same of the Cauchy stress formula used on the boundary 2.14. But, the integral on the left side of equation 2.7 is over the entire boundary Γ, not just Γ_q, where t_i is applied. So, we are going to exploit the weight functions arbitrariness, imposing that w_i presents zero values at the boundary with prescribed displacements Γ_u. Then, the equation 2.14 can be written in terms of the traction vector arriving at equation 2.15. That's the reason that tractions boundary condition are called natural boundary condition, since they are naturally incorporated in the formulation. Differently to the displacement boundary conditions, which are directly incorporated in the formulation and has to be satisfied first before derivation starts, or the process will fail (S. S. QUEK, 2003), hence they are called essential boundary conditions.

$$\int_{\Omega} w_i \rho \ddot{u}_i d\Omega + \int_{\Omega} w_{i,j} \sigma_{ij} d\Omega = \int_{\Gamma_q} w_i t_i d\Gamma \tag{2.15}$$

Analysing equation 2.15, the unknown variables are $\ddot{u}_i$ and σ_{ij}. In order to arrive at a formulation where only the unknown variable is displacement field, the Hooke's law 2.2 and the compatibility equation 2.3 are applied by finally deriving the weak formulation used in this work.

$$\int_{\Omega} w_i \rho \ddot{u}_i d\Omega + \int_{\Omega} w_{i,j} C_{ijkl} u_{k,l} d\Omega = \int_{\Gamma_q} w_i t_i d\Gamma \tag{2.16}$$

2.2.2 Interpolation functions

In the last subsection we derived the weak formulation used in this text. This formulation admits a trial function that approximates the variable of interest, in this case, the displacement field. We are going to use the elements and nodes created in the discretization of the domain to generate approximations of the variable of interest called element shape functions. Then, we will choose the trial function as a product of an weighted element shape function matrix and the nodal displacements of those elements.

Those element shape functions matrices are constructed using an approximation function of the variable of interest and it's nodal values. Choosing these functions is an important step, because it is related with the convergence of the FEM. It's not going to be proven in this text, but loosely speaking, continuity and completeness are the necessary conditions for FEM convergence (JACOB FISH, 2007). Where the continuity requirement was basically imposed on the development of the weak formulation, it must be piecewise continuously differentiable. And completeness is related with the property of the function to approximate a smooth function with arbitrary accuracy. Both are respected by the bilinear approximation.

The domains that were used with this text are obtained from images, hence they are pixel based. That's the reason why it's intuitive to use the pixel structure as base for the discretization and mapping. Henceforth, the elements are four-node quadrilateral, more specific, they are squares of the same geometry and the mesh is structured.

This allows the bilinear approximation to be used as base for our weighted element shape function matrices. Then, the trial functions are going to be approximated in each element as bilinear functions Φ of spatial coordinates x and y as follows:

$$\Phi^e(x,y) = a_0^e + a_1^e x + a_2^e y + a_3^e xy \tag{2.17}$$

where a_i^e are arbitrary parameters, the subscript e in $\Phi^e(x,y)$ and a_i^e refers to the element. This can be written in matrix form:

$$\Phi^e(x,y) = \begin{bmatrix} 1 & x & y & xy \end{bmatrix} \begin{bmatrix} a_0^e \\ a_1^e \\ a_2^e \\ a_3^e \end{bmatrix} \tag{2.18}$$

where

$$\mathbf{p} = \begin{bmatrix} 1 & x & y & xy \end{bmatrix} \tag{2.19}$$

$$\mathbf{a^e} = \begin{bmatrix} a_0^e \\ a_1^e \\ a_2^e \\ a_3^e \end{bmatrix} \tag{2.20}$$

or

$$\Phi^e = \mathbf{pa^e} \tag{2.21}$$

where the row vector of equation 2.18 is defined as $\mathbf{p}$ and the column vector of the equation 2.20 is defined as $\mathbf{a^e}$. We are aiming to write the parameters a_i^e in terms of the element nodal displacements to help to ensure the continuity between elements in further the development. In order to isolate $\mathbf{a^e}$, we write the approximation 2.17 for four knowable points, which are the element nodes numbered counterclockwise. Defining $d_n^e(x_n, y_n)$ as the nodal displacements, where (x_n, y_n) are the coordinates of node n in the element coordination system, and writing in matrix form:

$$\begin{bmatrix} d_0^e \\ d_1^e \\ d_2^e \\ d_3^e \end{bmatrix} = \begin{bmatrix} 1 & x_0^e & y_0^e & x_0^e y_0^e \\ 1 & x_1^e & y_1^e & x_1^e y_1^e \\ 1 & x_2^e & y_2^e & x_2^e y_2^e \\ 1 & x_3^e & y_3^e & x_3^e y_3^e \end{bmatrix} \begin{bmatrix} a_0^e \\ a_1^e \\ a_2^e \\ a_3^e \end{bmatrix} \tag{2.22}$$

$\mathbf{d^e}$ is the first column matrix , $\mathbf{P}$ is the 4x4 matrix in equation 2.22. Finally, we can express $\mathbf{a^e}$:

$$\mathbf{a^e} = \mathbf{P}^{-1}\mathbf{d^e} \tag{2.23}$$

and:

$$\Phi^e = \mathbf{N}(x, y)^e \mathbf{d^e} \tag{2.24}$$

$$\mathbf{N}(x, y)^e = \mathbf{pP}^{-1} \tag{2.25}$$

$\mathbf{N}(x, y)^e$ is the the element shape function matrix. The $N(x, y)^e$ components are the element shape functions. Representing N^e in the natural coordinate system with coordinates (r, s):

$$\mathbf{N}(r, s)^e = \begin{bmatrix} N_0^e(r, s) & N_1^e(r, s) & N_2^e(r, s) & N_3^e(r, s) \end{bmatrix} \tag{2.26}$$

$$\begin{aligned} N_0^e(r,s) &= \frac{1}{4}(1-r)(1-s) \\ N_1^e(r,s) &= \frac{1}{4}(1+r)(1-s) \\ N_2^e(r,s) &= \frac{1}{4}(1+r)(1+s) \\ N_3^e(r,s) &= \frac{1}{4}(1-r)(1+s) \end{aligned} \tag{2.27}$$

Since their derivatives are necessary in the development of FEM, we also are going to present their derivatives in the s and r direction:

$$\begin{bmatrix} \frac{\partial N_0}{\partial r} = \frac{-1+s}{4} & \frac{\partial N_1}{\partial r} = \frac{1-s}{4} & \frac{\partial N_2}{\partial r} = \frac{1+s}{4} & \frac{\partial N_3}{\partial r} = \frac{-1-s}{4} \\ \frac{\partial N_0}{\partial s} = \frac{-1+r}{4} & \frac{\partial N_1}{\partial s} = \frac{-1-r}{4} & \frac{\partial N_2}{\partial s} = \frac{1+r}{4} & \frac{\partial N_3}{\partial s} = \frac{1-r}{4} \end{bmatrix} \tag{2.28}$$

2.2.3 Galerkin Method

In this subchapter, we are going to employ both the weak formulation and the element shape functions to derive our element equation. Equation 2.16 with the the boundary conditions governs elasticity in the entire domain, including the elements created through its discreatization. For this purpose we are going to need to define the element nodal displacement vector $\mathbf{d}^e$. Where the superscript e refers to the element. Since, this is a 2D formulation and we are using a quadrilateral element, there are two degrees of freedom per node and there are eight components $\mathbf{d}^e$.

$$\mathbf{d}^e = \begin{bmatrix} u_{1x} & u_{1y} & u_{2x} & u_{2y} & u_{3x} & u_{3y} & u_{4x} & u_{4y} \end{bmatrix}^T \tag{2.29}$$

As explained in the last chapter, we are going to use the element shape function matrices to approximate displacement field, that estimates the displacement fields, u_x^e and u_y^e using the nodal displacements $\mathbf{d}^e$, hence:

$$\begin{bmatrix} u_x \\ u_y \end{bmatrix}^e = \begin{bmatrix} N_0^e & 0 & N_1^e & 0 & N_2^e & 0 & N_3^e & 0 \\ 0 & N_0^e & 0 & N_1^e & 0 & N_2^e & 0 & N_3^e \end{bmatrix} \mathbf{d}^e \tag{2.30}$$

where the strain–displacement matrix $\mathbf{B}^e$ is defined as:

$$\mathbf{B}^e = \begin{bmatrix} \frac{\partial N_0}{\partial x} & 0 & \frac{\partial N_1}{\partial x} & 0 & \frac{\partial N_2}{\partial x} & 0 & \frac{\partial N_3}{\partial x} & 0 \\ 0 & \frac{\partial N_0}{\partial y} & 0 & \frac{\partial N_1}{\partial y} & 0 & \frac{\partial N_2}{\partial y} & 0 & \frac{\partial N_3}{\partial y} \\ \frac{\partial N_0}{\partial y} & \frac{\partial N_0}{\partial x} & \frac{\partial N_1}{\partial y} & \frac{\partial N_1}{\partial x} & \frac{\partial N_2}{\partial y} & \frac{\partial N_2}{\partial x} & \frac{\partial N_3}{\partial y} & \frac{\partial N_3}{\partial x} \end{bmatrix} \tag{2.31}$$

$$\varepsilon^e = \begin{bmatrix} \varepsilon_{xx} \\ \varepsilon_{yy} \\ \gamma_{xy} \end{bmatrix}^e = \mathbf{B}^e \mathbf{d}^e \tag{2.32}$$

We are, also, going to use the element shape functions to approximate weight functions as well as the trial functions. This step defines this method as a method that belongs to the Galerkin class. So, we are going to write the displacement field and weight functions in terms of nodal values. In our case, the nodal displacement are time-dependent that's the reason this is a system of semidiscretization, where is spacially discretitazed but temporally continous.

$$u(x, y, t) \approx \mathbf{N}(x, y)\mathbf{d}(t)^e \tag{2.33}$$

$$w(x, y)^T \approx \mathbf{w}^{eT}\mathbf{N}(x, y)^T \tag{2.34}$$

Applying the approximations of displacement, weight at the weak formulation and rewriting 2.16 in matrix from for the element domain and writing C_{ijkl} as $\mathbf{C}$, where $\mathbf{C}$ can be $\mathbf{C}_{PlaneStress}$ and $\mathbf{C}_{PlaneStrain}$ which are respectively the 2D constitutive relations for the plane stress and plane strain conditions, we get:

$$\mathbf{C}_{PlaneStress} = \frac{E}{1-\nu^2} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & (1-\nu)/2) \end{bmatrix} \tag{2.35}$$

$$\mathbf{C}_{PlaneStrain} = \frac{E}{(1+\nu)(1-2\nu)} \begin{bmatrix} 1-\nu & \nu & 0 \\ \nu & 1-\nu & 0 \\ 0 & 0 & (1-2\nu)/2) \end{bmatrix} \tag{2.36}$$

$$\int_{\Omega^e} \mathbf{w}^{eT}\mathbf{N}^{eT}\rho\mathbf{N}^e\ddot{\mathbf{d}}^e d\Omega + \int_{\Omega^e} \mathbf{w}^{eT}\mathbf{B}^{eT}\mathbf{C}\mathbf{B}^e\mathbf{d}^e d\Omega = \int_{\Gamma_u^e} \mathbf{w}^{eT}\mathbf{N}^{eT}\boldsymbol{t}^e d\Gamma \tag{2.37}$$

$$\mathbf{w}^{eT}\left(\int_{\Omega^e}\mathbf{N}^{eT}\rho\mathbf{N}^e\ddot{\mathbf{d}}^e d\Omega^e+\int_{\Omega^e}\mathbf{B}^{eT}\mathbf{C}\mathbf{B}^e\mathbf{d}^e d\Omega^e-\int_{\Gamma^e}\mathbf{N}^{eT}\boldsymbol{q}^e d\Gamma)\right)=0 \tag{2.38}$$

Since $\mathbf{w}^{eT}$ is an vector with arbitrary parameters, then we get element equation:

$$\int_{\Omega^{eT}}\mathbf{N}^{eT}\rho\mathbf{N}^e\ddot{\mathbf{d}}^e d\Omega^e+\int_{\Omega^e}\mathbf{B}^{eT}\mathbf{C}\mathbf{B}^e\mathbf{d}^e d\Omega^e-\int_{\Gamma^e}\mathbf{N}^{eT}\boldsymbol{q}^e d\Gamma=0 \tag{2.39}$$

Then we can define the following:

$$\mathbf{K}_e=\int_{\Omega_e}\mathbf{B}^{Te}\mathbf{C}\mathbf{B}^e d\Omega=\int_{\Gamma_e}\int_{h_e}\mathbf{B}^{Te}\mathbf{C}\mathbf{B}^e dh d\Gamma=h\int_{\Gamma_e}\mathbf{B}^{Te}\mathbf{D}\mathbf{B}^e d\Gamma \tag{2.40}$$

$$\mathbf{M}_e=\int_{\Omega_e}\mathbf{N}^{Te}\rho\mathbf{N}^e d\Omega \tag{2.41}$$

$$\mathbf{F}_e=\int_{\Gamma^e}\mathbf{N}^{eT}\boldsymbol{q}^e d\Gamma \tag{2.42}$$

$\mathbf{K}_e$ is the element stiffness matrix, where h is the element thickness, which is constant along the element. And $\mathbf{M}_e$ is the element consistent mass matrix, this nomenclature comes from the fact that this matrix is the one originally derived from FEM. In this text we are going to approximate the matrix $\mathbf{M}_e$ as the matrix $\mathbf{M}_{Lumped-e}$ 2.43, the lumped-mass matrix (WU, 2006) where m_e is the element mass. This strategy allows the approximation of the mass element as a vector, estimating each element nodal mass as the average of the element mass distributed on the element nodes. This step is important to avoid an inversion of the global mass matrix on a further step, an computational expensive task. Instead, the global mass matrix from the lumped mass matrix strategy can be represented as a vector and it's inversion is calculated with a inversion of it's terms, decreasing the computational resources necessary for the process. Lastly, $\mathbf{F}_e$ is the element force matrix.

$$\mathbf{M}_{Lumped-e}=\frac{1}{4}\mathbf{I}_4\int_{\Omega_e}\rho\, d\Omega=\frac{1}{4}\mathbf{I}_4 m_e=\frac{1}{4}\begin{bmatrix}m_e & 0 & 0 & 0\\ 0 & m_e & 0 & 0\\ 0 & 0 & m_e & 0\\ 0 & 0 & 0 & m_e\end{bmatrix} \tag{2.43}$$

With those definitions, we can rewrite equation 2.39 as:

$$\mathbf{M}_{Lumped-e}\ddot{\mathbf{d}}^{e} + \mathbf{K}_{e}\mathbf{d}^{e} = \mathbf{F}_{e} \tag{2.44}$$

The $\mathbf{M}_{Lumped-e}$ term of 2.44 it is calculated with the element mass, $\mathbf{F}_e$ can be directly evaluated imposing a force value at each node of the element. But, $\mathbf{K}_e$ cannot be integrated analytically, therefore rises the necessity of numerical methods to accomplish this task. Leading to the next subsection.

2.2.4 Gauss Quadrature

In order to numerically integrate $\mathbf{K}_e$ there are lots of techniques available. The chosen one was Gauss Quadrature, more specific the Legendre-Gauss quadrature, because is one of the most efficient techniques for functions that are polynomials or nearly polynomials. The Gaussian quadrature approximates the integral as a weighted sum of function values at specified points within the domain of integration, although the quadrature formulas are given by the domain [-1,1]. Which means a mapping is going to be needed, that's going to be developed using the determinant of jacobian of the matrix of the shape functions. In the case of two point Gauss quadrature, the weights and the function values are obtained by approximating the function as the sum of polynomials 1, x, $\mathrm{x}^{\mathbf{2}}$, $\mathrm{x}^{\mathbf{3}}$ weighted by w^e_{g1} and w^e_{g2}, respectively. Leading to the following linear system:

$$\begin{aligned} w_{g1} + w_{g2} &= 2 \\ x_1 w_{g1} + x_2 w_{g2} &= 0 \\ x_1^2 w_{g1} + x_2^2 w_{g2} &= 2/3 \\ x_1^3 w_{g1} + x_2^3 w_{g2} &= 0 \end{aligned} \tag{2.45}$$

Solving this linear system 2.45, we get:

$$w_{g1} = w_{g2} = 1, x_1 = -\frac{1}{\sqrt{3}}, x_2 = \frac{1}{\sqrt{3}} \tag{2.46}$$

Then, we can write an approximation for $\mathbf{K}_e$ 2.48 using the following definitions 2.45:

$$\mathbf{W}^e = \begin{bmatrix} w^e_{g1} \\ w^e_{g2} \end{bmatrix}, \mathbf{J^e} = \begin{bmatrix} \frac{\partial x}{\partial r} & \frac{\partial y}{\partial r} \\ \frac{\partial x}{\partial s} & \frac{\partial y}{\partial r} \end{bmatrix} \tag{2.47}$$

Where $\mathbf{W}^e$ is the vector with the weigths of the gaussian aproximations and $\mathbf{J^e}$ is the jacobian of the change of variables from the local coordinate system to the natural coordinate system.

$$\begin{aligned} \mathbf{K}_{GaussQuadrature-e} &= h\int_{-1}^{1}\int_{-1}^{1}\mathbf{B}(r,s)^{Te}\mathbf{D}\mathbf{B}(r,s)^{e}|\mathbf{J^e}(\mathbf{r},\mathbf{s})| = drds \\ \mathbf{K}_{GaussQuadrature-e} &= h\sum_{i=1}^{2}\sum_{j=1}^{2}\mathbf{W_i}^{e}\mathbf{W_j}^{e}\mathbf{B}(r_i,s_j)^{Te}\mathbf{D}\mathbf{B}(r_i,s_j)|\mathbf{J^e}(r_i,s_j)| \end{aligned} \tag{2.48}$$

2.3 Leap-frog Explicit Time Integration

The Leap-frog method is an explicit numerical integration scheme, commonly used to tackle transient problems. It often provides satisfactory results for wave propagation simulations, provided that stability criteria are met, as it is conditionally stable. Also, it requires significantly less computations per time step in comparison to usual implicit schemes, such as the Newmark method. In light of this, it is our direct integration method of choice for this work.

The formulation of the Leap-frog scheme is derived from the truncated Taylor's expansion over the central point on the time domain, leading to second order accuracy. The idea is to march in whole steps for displacement and acceleration, and in half steps for velocity. That is, the vectors $\mathbf{u}^{(i)}$ and $\ddot{\mathbf{u}}^{(i)}$ are evaluated at each step $i \in \{1, 2, \ldots, n_\tau\}$, while $\dot{\mathbf{u}}^{(j)}$ is computed at each step $j \in \{1+\frac{1}{2}, 2+\frac{1}{2}, \ldots, n_\tau-\frac{1}{2}\}$, where n_τ is the number of time steps. A first $\dot{\mathbf{u}}^{(\frac{1}{2})}$ can be evaluated via a half step with the Euler method, for example. Then, at each step i, the following expressions must be computed:

$$\mathbf{u}^{(i)} = \mathbf{u}^{(i-1)} + \dot{\mathbf{u}}^{(i-\frac{1}{2})}\Delta_\tau, \tag{2.49}$$

$$\ddot{\mathbf{u}}^{(i)} = \mathbf{M}^{-1}(\mathbf{F}^{(i)} - \mathbf{K}\mathbf{u}^{(i)}), \tag{2.50}$$

$$\dot{\mathbf{u}}^{(i+\frac{1}{2})} = \dot{\mathbf{u}}^{(i-\frac{1}{2})} + \ddot{\mathbf{u}}^{(i)}\Delta_\tau, \tag{2.51}$$

where Δ_τ is the time increment at each step. Equations 2.49 and 2.51 are FDM approximations for the velocity and displacement vectors, respectively. Equation 2.50 is simply a rearrangement of the version of equation 2.44 for the global matrices where $\mathbf{F}^{(i)}$ and $\mathbf{K}$ are the global force and stiffness matrices, respectively. The computation of of those global matrices, using the elements matrices is explored on chapter 3. It is crucial to observe that $\mathbf{M}$ is a diagonal matrix, due to the consideration of lumped mass. This allows for

the matrix-vector product with its inverse to be trivially computed as a term-by-term vector division. No dense inverse matrix must be stored, nor must a system of algebraic equations be solved at each time step.

2.4 Stability Criteria

As stated earlier, the explicit formulation is conditionally stable. The condition used as proposed by (COURANT; FRIEDRICHS; LEWY, 1928) is

$$\Delta_\tau \leq 2/\omega, \tag{2.52}$$

where Δ_τ is the size of the time step and ω is the natural frequency of the system. Estimating ω is computationally expensive. It can be proven that the maximum natural frequency of the system is lesser than or equal to the natural frequency of its elements (HUGHES; LIU, 1978), applying the latter frequency in equation 2.52 results in a time step lesser than or equal to the critical time step, which is an conservative approach. The natural frequency of an element e is estimated with

$$\omega_e = 2v_p/\Delta_x, \tag{2.53}$$

where v_p is the primary elastic velocity of the material and Δ_x is the spatial step size, i.e. the pixel size in our context. The time increment for our simulations is then chosen as

$$\Delta_\tau \leq \min\{2/\omega_e \mid e \in E\}. \tag{2.54}$$

Note that, as all elements share the same Δ_x, it is sufficient to evaluate ω_e for each material's element sample, instead of repeating this process for each element in the mesh.

Chapter 3

Implementation

3.1 Algorithm

The programs devised in this work are focused on performing the Leap-frog integration for the explicit FEM formulation described in the previous section. The implemented numerical solver can be conceptually synthesized as in algorithm 1.

Algorithm 1 Explicit FEM solver with Leap-frog integration

Input: $\mathbf{M}$, $\mathbf{K}$, $\mathbf{F}$, Δ_τ, n_τ, $\mathbf{u}^{(0)}$, $\dot{\mathbf{u}}^{(0)}$, $\overline{\mathbf{u}}_\Gamma$
Output: $\mathbf{u}, \dot{\mathbf{u}}$
$\ddot{\mathbf{u}}^{(0)} \leftarrow \mathbf{M}^{-1}(\mathbf{F}^{(0)} - \mathbf{K}\mathbf{u}^{(0)})$ {compute initial acceleration}
$\dot{\mathbf{u}}^{(\frac{1}{2})} \leftarrow \dot{\mathbf{u}}^{(0)} + \ddot{\mathbf{u}}^{(0)}\frac{\Delta_\tau}{2}$ {Euler method for velocity at $t = \Delta_\tau/2$}
for each i in $\{1, 2, \ldots, n_\tau\}$ **do**
    $\mathbf{u}^{(i)} \leftarrow \mathbf{u}^{(i-1)} + \dot{\mathbf{u}}^{(i-\frac{1}{2})}\Delta_\tau$ {update displacement}
    $\mathbf{u}_\Gamma^{(i)} \leftarrow \overline{\mathbf{u}}_\Gamma^{(i)}$ {impose Dirichlet boundary conditions}
    $\ddot{\mathbf{u}}^{(i)} \leftarrow \mathbf{M}^{-1}(\mathbf{F}^{(i)} - \mathbf{K}\mathbf{u}^{(i)})$ {update acceleration}
    $\dot{\mathbf{u}}^{(i+\frac{1}{2})} \leftarrow \dot{\mathbf{u}}^{(i-\frac{1}{2})} + \ddot{\mathbf{u}}^{(i)}\Delta_\tau$ {update velocity}
end for

There are several different ways of implementing a computational procedure for the problem at hand. A usual and relatively straightforward strategy would be to make use of sparse linear algebra libraries to handle the demanded matrix and vector operations, allocating the global matrices in memory. Although computationally efficient, such an approach can often lead to significantly high memory requirements as meshes grow larger. Given that we aim to deal with domains characterized by images, simulation sizes grow rapidly with increasing pixel count, either due to improvements in resolution or large image dimensions. On the other hand, this sort of mesh is structured as a regular grid, so connectivity information and degree of freedom (DOF) mapping can be inferred on-

the-fly, and matrix-free (or assembly-free) schemes can become extremely lightweight, as shown in (LOPES, Pedro Cortez Fetter et al., 2022b). As the end goal of our efforts in this context is simulating three-dimensional wave propagation in μCT samples with hundreds of millions, or even billions, of voxels with accessible hardware, the pursuit for space efficiency is an important factor in our developments.

The code was implemented in the Julia language, which makes use of "just ahead of time" compilation to leverage script-like syntax with performance on par to that of compiled languages. A node-by-node (NbN) approach (MARTÍNEZ-FRUTOS; HERRERO-PÉREZ, 2015; KIRAN; GAUTAM; SHARMA, 2020) was adopted for the computation of acceleration at each time step. Both the lumped mass coefficients and the resulting entries of the vector $(\mathbf{F} - \mathbf{Ku})$ are computed without any global matrix or extra vector being assembled. Furthermore, it is possible to avoid allocating the acceleration vector itself, by adding its would-be resulting entries, scaled by the time increment, directly into the velocity vector. This can be seen as a merging of lines 8 and 9 (also 3 and 4), in algorithm 1. In that sense, we have devised a memory efficient matrix-free routine for updating the velocity at each time step in image-based simulations, which is described by algorithm 2.

Algorithm 2 Pixel-based NbN velocity update

1: **Input:** image, ρ, $\mathbf{K}_e$, $f(x,t)$, Δ_τ, i, $\mathbf{u}^{(i)}$, $\dot{\mathbf{u}}^{(i-\frac{1}{2})}$
2: **Output:** $\dot{\mathbf{u}}^{(i+\frac{1}{2})}$
3: **for** each n in N **do**
4: $\quad m \leftarrow \sum_{e \in E_n} \frac{1}{4} \int_{\Omega_e} \rho \, d\Omega$ {mass, E_n = set of neighbor elements}
5: $\quad$ **for** each d in D_n **do**
6: $\quad\quad \ddot{u} \leftarrow \frac{1}{m} \left(f(d, i\Delta_\tau) - \sum_{e \in E_n} \mathbf{K}_{e(d)} \mathbf{u}_e^{(i)} \right)$ {acceleration}
7: $\quad\quad \dot{\mathbf{u}}_d^{(i+\frac{1}{2})} \leftarrow \dot{\mathbf{u}}_d^{(i-\frac{1}{2})} + \ddot{u}\Delta_\tau$ {update entry d of the velocity vector}
8: $\quad$ **end for**
9: **end for**

In algorithm 2, no global matrix is employed. The diagonal lumped mass matrix coefficients are recomputed on demand in terms of the material's density ρ within Ω_e, for each neighbor element e. Alternatively, they can be stored in a node-indexed array. Likewise, the matrix-vector product $\mathbf{Ku}$ is broken down to a sum of products with local matrices. The term $\mathbf{K}_{e(d)}$ is used to represent the line of local stiffness matrix $\mathbf{K}_e$ related to the DOF d. Recall that, although $\mathbf{K}_e$ and $\int_{\Omega_e} \rho \, d\Omega$ are associated with individual elements, they can be computed and stored for each material phase, as all elements share the same geometry. The image is employed as a material map, with each distinct color value depicting a material in the heterogeneous domain. As described in (LOPES, Pedro

Cortez Fetter et al., 2022b), it is possible to store the required pixel information through nodal color keys. Also, it is important to remark that no incidence matrix is required to obtain $\mathbf{u}_e$, as the structured nature of the mesh allows for the proper indexes of $\mathbf{u}$ to be accessed without any pre-computed connectivity map. For the sake of generality, it is assumed that a lambda function $f(x,t)$ computes the force components due to surface traction at each time step for any given position, herein mapped via a DOF index. There is considerable room for various sorts of load modeling in the proposed scheme, but here we keep to either null or constant forces in our simulations. Anyhow, the idea is to avoid allocating a full-sized global force vector, or even worse, a matrix of force vectors over time.

3.2 Code structure

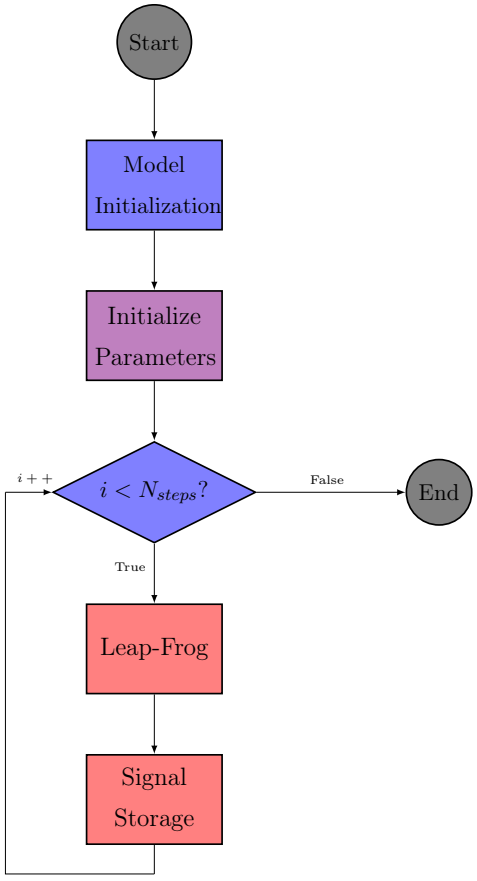


Figure 3.1: Code structure: Time-integration and signal storage. N_{steps} is the number of time steps.

Four different implementations were employed to reproduce different tests:

- Version 1 was used in test 1.

- Version 2 was used in tests 2, 3, 4 and 5;
- Version 3 in tests 6, 7, 8 and 9;
- Version 4 was used in test 10 and 11.

All those implementations share the following code structure steps:

3.2.1 Model and parameters initialization.

Our model construction is executed through the conversion of a segmented microtomography image to a binary map which is used to identify the pixel's material. The parameters of the simulation are read from a neutral file. Those parameters are the time of the simulation, the time step, the pixel size and each material's elastic properties and density. Which are used to calculate and store the element stiffness and lumped-mass matrices for each material. On version 3 of the implementation, we also read a file with the displacement values that are going to be applied on the upper face of the model. Lastly, we start the leap-frog method with one time step performed with Euler method using the initial condition.

3.2.2 Time integration.

Our leap-frog method is performed with a node-by-node strategy. Since our mesh is pixel-based, it is structured. Hence, the neighboring elements of each node can be inferred with the nodal index as well as their respective DOF's. For each node and at each time step on the time integration, we calculate the element map and the DOF maps. Each version of the code have modifications in order to reproduce different experiments:

- Version 1: We check on if the node is on the left face of the model and apply the corresponding force at each node.
- Version 2: On each time step, we calculate the Gaussian curve displacement value. Then we apply this value to all nodes on the left face of the model instead of performing the time integration for those nodes.
- Version 3: On each time step, we apply the values read from the file on all nodes on the upper face of the model instead of performing the time integration for those nodes.

- Version 4: On each time step, we calculate the Gaussian curve displacement value. Then we apply this value to all nodes on the left face of the model instead of performing the time integration for those nodes. Also, we calculate element map and the DOF maps respecting the periodic boundary condition.

3.2.3 Signal storage.

The results presented in chapter 4 are obtained with this step, which is when the displacement value is stored.

- Version 1: The mean value of the displacements on each node of the left face of the model on the x direction are stored.
- Version 2: All displacement values on each node as in 4.6 on the x direction are stored.
- Version 3: All displacement values on y direction are stored.
- Version 4: The mean value of the displacements on each node of the receiver position as in 4.21 on the x direction are stored.

3.3 Memory usage

With the implementation proposed in algorithm 2, memory requirements are significantly reduced from solutions where global matrices are allocated, even in sparse format. Only two n-sized arrays of double precision floating point variables must be stored, one for $\mathbf{u}$, another for $\dot{\mathbf{u}}$, where n is the number of DOFs. A vector of nodal mass may be stored, if advantageous, adding an $(n/2)$-sized array to the requirements. And, of course, the image itself must be loaded in memory, which in our case amounts to either 8bit or 16bit unsigned variables per pixel or per node, respectively. Additional data stored in RAM comprehends Dirichlet boundary conditions and one local matrix per material, demanding significantly less memory than the aforementioned arrays. In a nutshell, our program requires roughly

$$\text{memory} \approx 2 \times 2n_p \times 8 + n_p \times 8 + n_p \times 2 = 42n_p \text{ Bytes} \tag{3.1}$$

to carry out a simulation with an image that has n_p pixels, which means that models with 10000×10000 pixels (200 million DOFs) can be studied with about 4.2 GB allocated in

RAM. However, it is crucial to perceive that the low memory demand comes at a trade-off in execution time. The NbN velocity update procedure essentially recomputes the coefficients of both global matrices at each time step, so the process does become more computationally expensive.

Chapter 4

Results and Discussion

Eleven tests were made to validate our implementation, the tests were conceived in a way to verify the quality of different aspects of our implementation. Test 1 compares our simulation to a benchmark from (MAZZOCHI; SUAREZ; ROSSI, 2020), where the accuracy and convergence of the method is attested. Test 2 studies essentially 1D propagation of a Gaussian pulse in unbounded media. In this test, possible errors including numerical errors and dispersion were investigated. Test 3 to 5 reproduces a Gaussian pulse propagating through an interface between two semi-infinite half-spaces. In this test, reflection and transmission phenomena was evaluated. Test 6 to 9 simulates the propagation of normal and oblique Gaussian wave beam impinging an interface between two semi-infinite half spaces. This is an essentially 2D scenario, and we qualitatively compare the obtained results with analytical results obtained in the frequency domain. In these tests, we investigate the reflection and transmitted phenomena as well as evanescent waves. In tests 10, we simulate the wave propagation in laminated medium and compare the obtained effective longitudinal wave speed (V_p) to the one obtained via the Backus average. Finally, in test 11 we compute the effective V_p for a microtomografic image (Berea Stripe - BS) using the developed computational tools. We yet compare this obtained effective V_p with the one calculated using the effective elastic parameters obtained via an in house developed elastostatic voxel-based finite element implementation (LOPES, Pedro C. F. et al., 2022).

4.1 Test 1 - Forced vibration: bar subjected to impulsive loading.

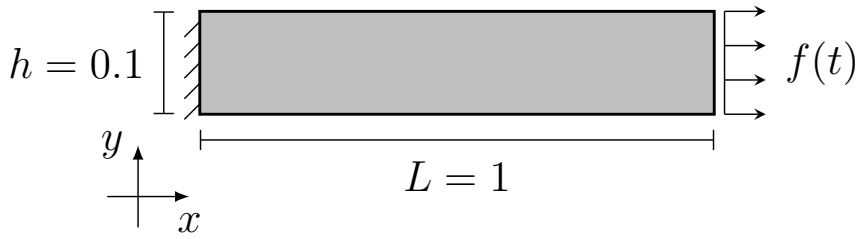


Figure 4.1: Representation of the problem presented in (MAZZOCHI; SUAREZ; ROSSI, 2020), where a bar with clamped-free extremities is subjected to an impulsive load described by Heaviside's function applied in its free extremity. We reproduced this result and compared to the analytical response and presented the results in figures 4.2 4.3.

The first numerical simulation aimed to reproduce results from the benchmark presented in (MAZZOCHI; SUAREZ; ROSSI, 2020), where a clamped-free bar is subjected to an impulsive load described by Heaviside's function applied in its free extremity. The material elastic constants are $E = 1$ Pa, $\nu = 0$, $\rho = 1$ Kg/m^3. Plane stress was assumed. The geometry of the model consists of width $L = 1$ m between the clamped extremity and the free extremity and height $h = 0.1$ m. The distributed load $f(t)$, as in figure 4.1, with value 1 N/m was applied from the beginning of the simulation and lasted 16 seconds. Such a distributed load was applied in our FEM implementation as a concentrated load at the corresponding nodes of the free extremity.

The way this test was conceived, with choices such as $\nu = 0$ and the unidirectional load in the x direction creates a displacement pattern in time and along the model where there are no expansions and contractions in the y direction. Therefore, this problem is essentially a one-dimensional problem that was studied with our two-dimensional explicit FEM implementation.

Different discretizations were used to study the convergence and precision of the implementation as presented. They are presented in table 4.1. In Figure 4.2 we plot the analytical result and the numerical results fol all the considered discreatizations. Figure 4.3 is a magnified view of the the region marked in red in the Figure 4.2. In the Figure 4.2 we see the excellent agreement between the analytical and numerical results (all the curves are indistinguishable in the figure), validating our formulation and implementation. Further, all numerical results were accurate regardless the considered discreatization. This is due to the fact that we chose all the discreatization respecting the criteria from (COU-

RANT; FRIEDRICHS; LEWY, 1928). In Figure 4.2 we see that, in fact, the better the discreatization the closer the numerical results are to the analytical results, confirming the implementation robustness and convergence. This stability robustness and convergence were observed in all other tests that we show in the present work as well, as we chose the discreatizations accordingly to the criteria (**{courant1928partiellen}**).

Test	element mesh	voxel size(mm)	time steps	time step size (ms)
Test 1.1	1x10	100	1600	10
Test 1.2	2x20	50	1600	10
Test 1.3	4x40	25	1600	10
Test 1.4	16x160	6.25	3200	5

Table 4.1: Element meshes, voxel size in mm, time steps and time steps size in ms of tests 1.1,1.2,1.3,1.4.

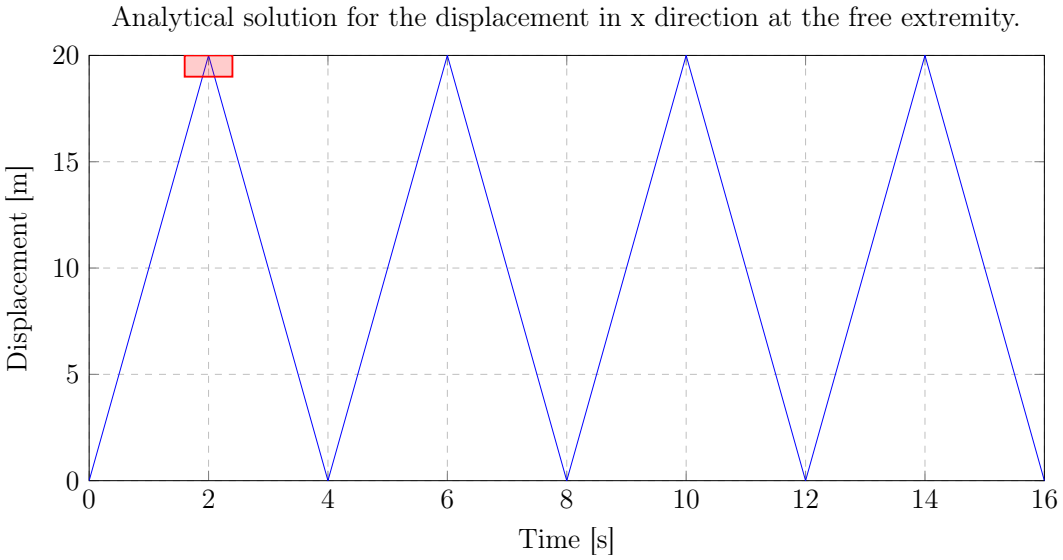


Figure 4.2: Analytical result of the problem in (MAZZOCHI; SUAREZ; ROSSI, 2020) in black and Test 1.4 in blue. Since the numerical results and the analytical results are close, their difference are better represented in figure 4.3, which is the zoom of the region marked in red. Where the convergence of the method can be observed as the mesh becomes thinner and the time step decreases confirming the implementation robustness.

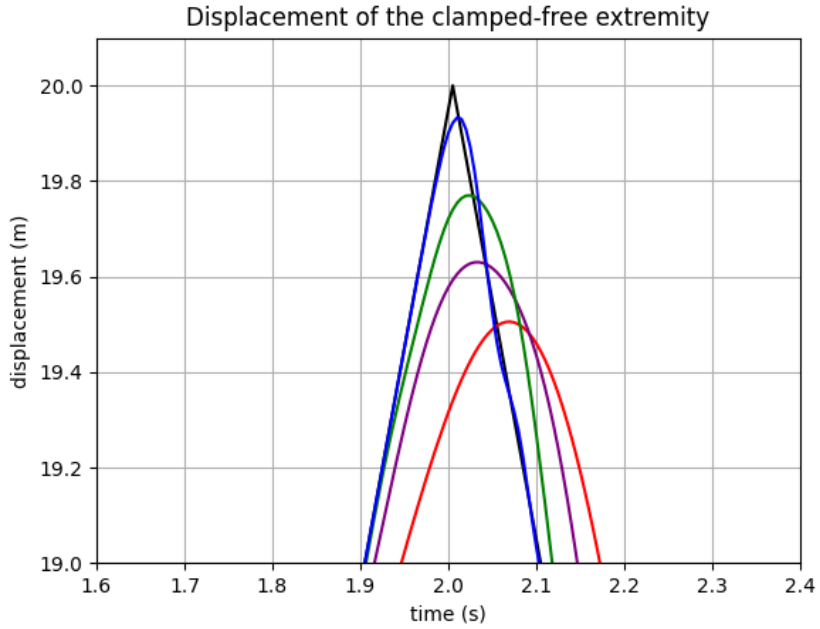


Figure 4.3: Analytical result of the problem in (MAZZOCHI; SUAREZ; ROSSI, 2020), results of tests 1.1 in red, 1.2 in purple, 1.3 in green and 1.4 in blue for x between 1.6 and 2.4 seconds and y between 19.0 and 20.1 meters.

4.1.1 Test 2 - P Gaussian pulse travelling in a unbounded media.

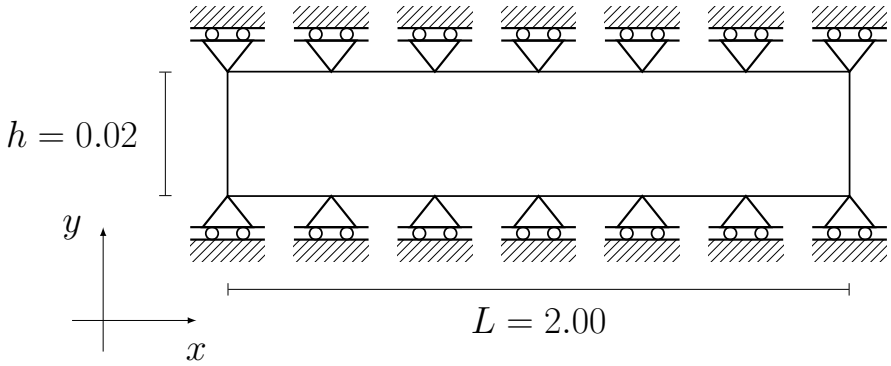


Figure 4.4: Illustration of the geometrical model and constrains used test 2, where h and L are the height and length used at test 2, respectively. Those constrains are called symmetry boundary condition that imposes zero displacement on the direction of the pulse propagation at the boundaries that are lateral to the pulse propagation direction. In test 2 a Gaussian pulse travelling in a homogeneous media is reproduced by imposing a prescribed displacement boundary condition on the left face of the boundary changing over time with values of a Gaussian curve.

In Test 2, a P wave Gaussian planar pulse travelling in a unbounded media is reproduced. We prescribed the displacement on the left face of the model represented in 4.4 changing

over the time following the values of a Gaussian curve with amplitude of 1 meter, center of peack at 0.25 seconds after the beginning of the pulse and deviation with modulus of 0.05, also the time simulated did not allow the Gaussian pulse to interact with the right face of the model. The models heigth, h, is 0.02m and length, L, is 2m as in 4.4. On the faces lateral to the pulse propagation direction, symmetry boundary conditions were applied imposing zero values on the transverse direction to pulse propagation direction. The elastic constants, density and parameters of the simulation are represented in table 4.2.

Input	Test 2
V_p[m/s]	1
V_s[m/s]	0.5
ρ[Kg/m^3]	2
model discretization[elements^2]	2000x2
model dimensions[m^2]	2x0.002
Time[s]	2
Time Steps	20000

Table 4.2: Informations of Test 2. V_p and V_s are the elastic wave velocity with polarity p and s, respectively. ρ is the density of the material. Model discretization represents the number of elements in x and y directions,respectively. The simulation reproduced the pulse travelling for 2 seconds using 20000 time steps.

This simulation results represents the propagation of a planar Gaussian pulse with p polarity. We assume that the media is infinite in the perpendicular directions of the propagation using constrains as represented in figure 4.4 and plane strain was asusmed. The constrains represented in figure 4.4 inhibits contractions and expansions of the displacement field in the y direction, which is a behavior found in wavefronts of 2D planar waves. Since we also assumed plane strain and all nodes on the face where the Gaussian pulse is prescribed have the same displacement, we were able to approximate a 3D planar wave propagation in an infinite media, an essentially 1D phenomenon.

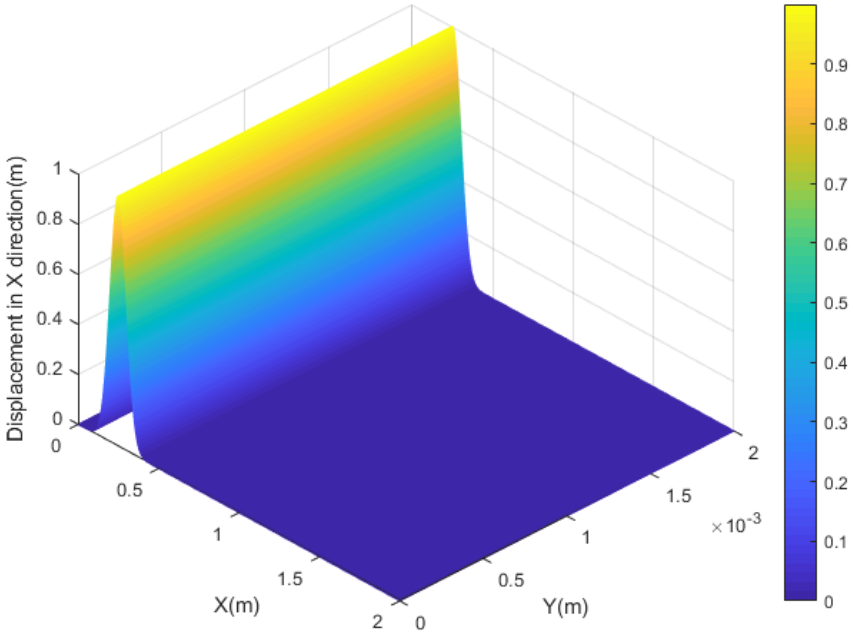


Figure 4.5: Displacement in the x direction at the entire domain at 0.5 seconds showing that the displacement only changes in the x direction, which is typical for a planar wave.

The purpose of this test was to evaluate the quality of the measure of wave velocity in a homogeneous media and to visually inspect the wave distortion. So, we've plotted the displacement on the marked nodes as in 4.6 at times 0.5s, 0.75s and 2 seconds and plotted all the displacements on the entire model at 0.5s 4.5. The elastic wave velocity was estimated with the position were the peak of the Gaussian pulse was located at times 0.5s and 0.75. The peak moved 0.25 meters from 0.25m to 0.50m as in figure 4.7 during 0.25 seconds. The obtained value of the elastic wave velocity was 1m/s which corresponds to the elastic property V_p form 4.2. At figure 4.5 no significant change of the displacement field in y direction was observed, only in x direction, which is typical for a planar wave. It was not possible to see significant distortion on the wave's propagation comparing the wave at 0.5s, 0.75s and 2s as expected from a non-dispersive medium, also their peak values agree to six decimal places. Hence, no significant numerical dispersion and dissipation were observed. We have also tried to increase the time step with values higher than the limited proposed by equation 2.54, which led to instability on the response of the program, the displacement pattern related to wave propagation is replaced with values that seems to alternate from negative infinity to positive infinity.

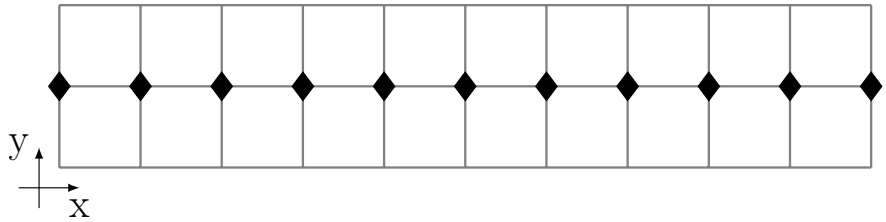


Figure 4.6: Representation the discretization used in the tests 2, 3, 4 and 5. The actual model is a hundred times larger in the x direction as in table 4.2 and 4.3. The marked nodes are the ones where the displacement is recorded and represented in figures 4.4, 4.9, 4.10, 4.11

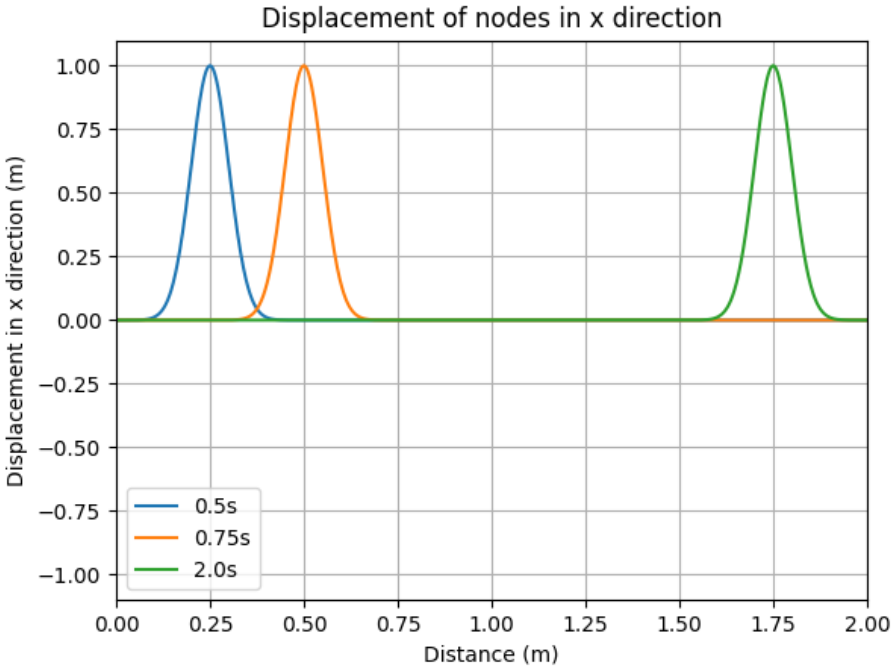


Figure 4.7: Simulation results of the displacement in x direction of all the central nodes in y direction 4.6. In blue, orange and green are represented the times 0.5s, 0.75s and 2s, respectively.

4.1.2 Test 3, 4 and 5 - P Gaussian pulse impinging an interface between two semi-infinite half-space.

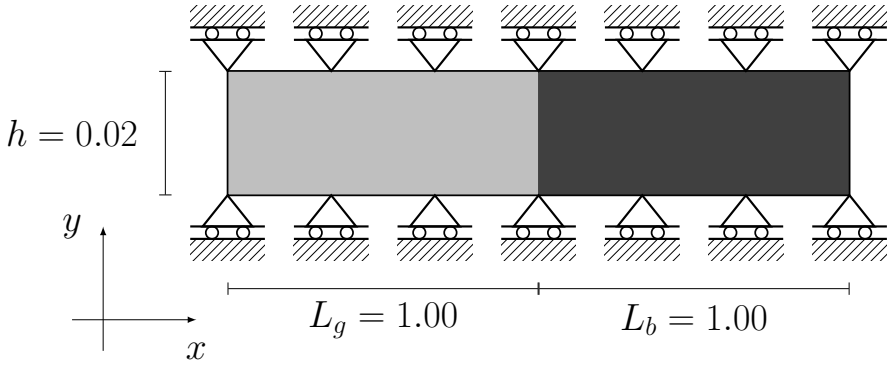


Figure 4.8: Illustration of the geometrical model and constrains used test 3. The gray and black colors represents two different media. The gray and black materials properties are V_{p1}, V_{s1}, ρ_1, Z_1 and V_{p2}, V_{s2} ,ρ_2, Z_2, respectively, as in table 4.3. The heigth used in test 3, h is the same as the test 2 with value 0.02m. L_g and L_b are the lengths of the gray and black medias, both with value 1m. In this test we also used symmetry boundary conditions and prescribed the Gaussian pulse the same way as test 2.

In tests 3, 4 and 5 a P Wave Gaussian pulse impinging an interface between two semi-infinite half-space was reproduced. Those tests are similar to test 2, the same boundary conditions were applied including the prescribed displacement representing the Gaussian pulse and the symmetry boundary condition as in figure 4.8, plane strain was assumed. The difference is that different material were used, the time simulated and the number of time steps as in 4.3. This change allows the study of the implementation behavior when dealing with transmitted and reflected waves. The choice of the materials properties was based on the relation of impedance between those materials which leads to different reflection phenomena.

Input	Test 3	Test 4	Test 5
V_{p1}[m/s]	1	2	0.5
V_{s1}[m/s]	0.5	1	0.25
ρ_1[Kg/m^3]	2	2	8
Z_1 [Kg/m^{2}s]	2	4	4
V_{p2}[m/s]	2	1	2
V_{s2}[m/s]	1	0.5	1
ρ_2[Kg/m^3]	2	2	1
Z_2 [Kg/m^{2}s]	4	2	4
model discretization	2000x2	2000x2	2000x2
model dimensions[m^2]	2 x 0.002	2 x 0.002	2 x 0.002
Time[s]	4	4	4
Time Steps	40000	40000	40000

Table 4.3: Informations of Test 3, 4 and 5. V_p and V_s are the elastic wave velocity with polarity p and s, respectively. ρ is the density of the material, Z is the impedance of the material calculated with the product of ρ and V_p. Model discretization represents the number of elements in x and y directions, respectively. The simulations on test 3, 4, 5 reproduced the pulse travelling for 4 seconds using 40000 time steps. V_{p1}, V_{s1}, ρ_1 and Z_1 are properties of the gray media 4.8 and V_{p2},V_{s2},ρ_2 and Z_2 are properties of the black media 4.8.

$$Z = \rho V_p \tag{4.1}$$

$$R = \frac{Z_1 - Z_2}{Z_1 + Z_2} \tag{4.2}$$

$$T = \frac{2Z_1}{Z_1 + Z_2} \tag{4.3}$$

Equation 4.1 show the relation between impedance, density and primary velocity. Equations 4.2 and 4.3 predict the relation between the incident wave amplitude and their reflected and transmitted waves, respectively. When $Z_1 > Z_2$, the amplitude of the reflected wave is positive. $Z_1 < Z_2$ leads to negative amplitude of the reflected wave. Lastly, $Z_1 = Z_2$ does not produce reflected waves by theory prediction.

Amplitude	Test 3	Test 4	Test 5
R	-0.33333	0.33333	0.0
Measured R	-0.33335	0.333349	0.0
T	0.66666	1.33333	1.0
Measured T	0.66664	1.33334	0.99987

Table 4.4: Results of tests 3, 4 and 5. R and T are the prediction calculated as in 4.2 and 4.3, respectively. Measured R and Measured T are the reflected and transmitted amplitudes measured on the simulation.

The amplitude of the simulated waves are compared to theory prediction in table 4.4. The difference between the prediction and the measured amplitudes are less than 0.1%, indicating the simulation accuracy when dealing with transmission and reflection caused by the wave passing through different media. The reflected and transmitted waves used to acquire table 4.4, are shown in figures 4.9, 4.10 and 4.11. From those figures, we can see that the reflected and transmitted waves are also Gaussian curves. This is the expected result, since the reflection coefficient is frequency-independent. Hence, all the different waves with their respected frequencies that composes the Gaussian pulse suffers the same difference in amplitude showing that the change in impedance is responsible for the scaling of amplitude of the Gaussian wave. In test 5, the impedance is constant showing that the relation of the velocities between those media is responsible for the scaling in the x direction of the Gaussian pulse.

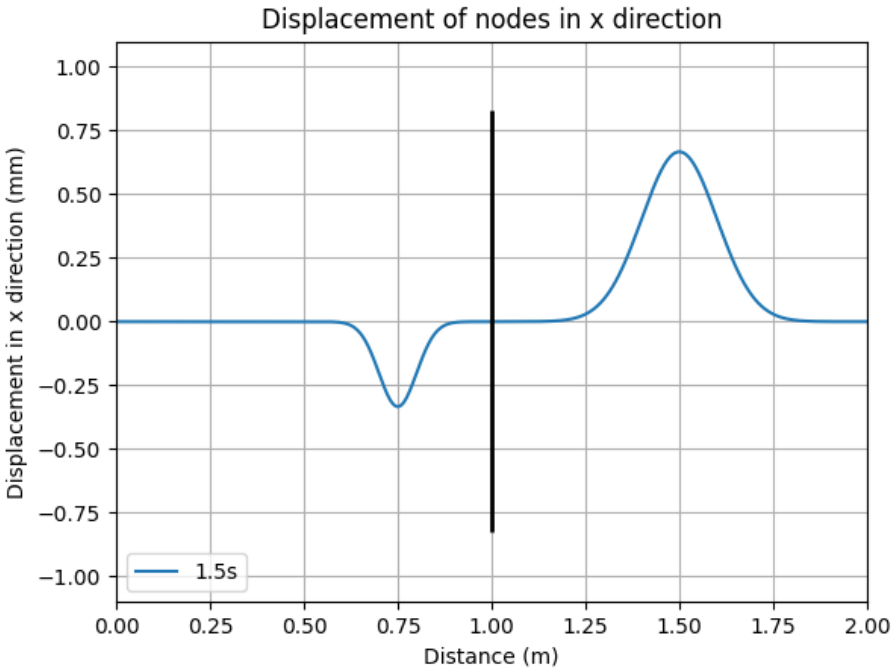


Figure 4.9: Test 3 results. Y axis represents the displacement in x direction of all centralized nodes in y direction as in 4.6 at 1.5 seconds. X axis represents the length of the model used to reproduce the interface between two semi-infinite half-space. The incident Gaussian pulse is prescribed on the left face of the model. As expected from the theory, the reflected wave, in the left, has negative amplitude and the transmitted wave, in the right, has positive amplitude. The vertical line in black represents the interface.

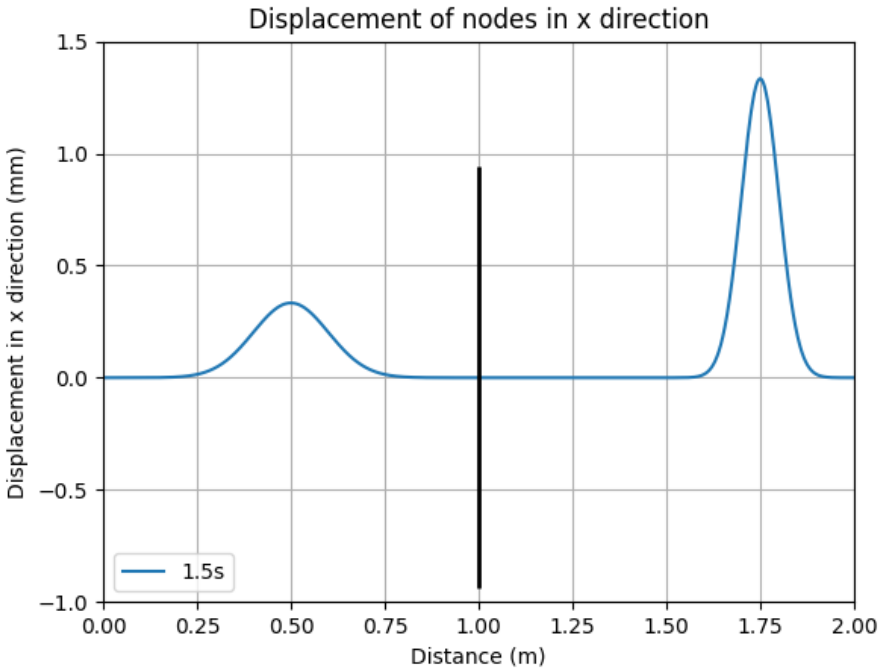


Figure 4.10: Test 4 results. Y axis represents the displacement in x direction of all centralized nodes in y direction as in 4.6 at 1.5 seconds. X axis represents the length of the model used to reproduce the interface between two semi-infinite half-space. The incident Gaussian pulse is prescribed on the left face of the model. As expected from the theory, the reflected wave, in the left, has positive amplitude and the transmitted wave, in the right, also has positive amplitude. The vertical line in black represents the interface.

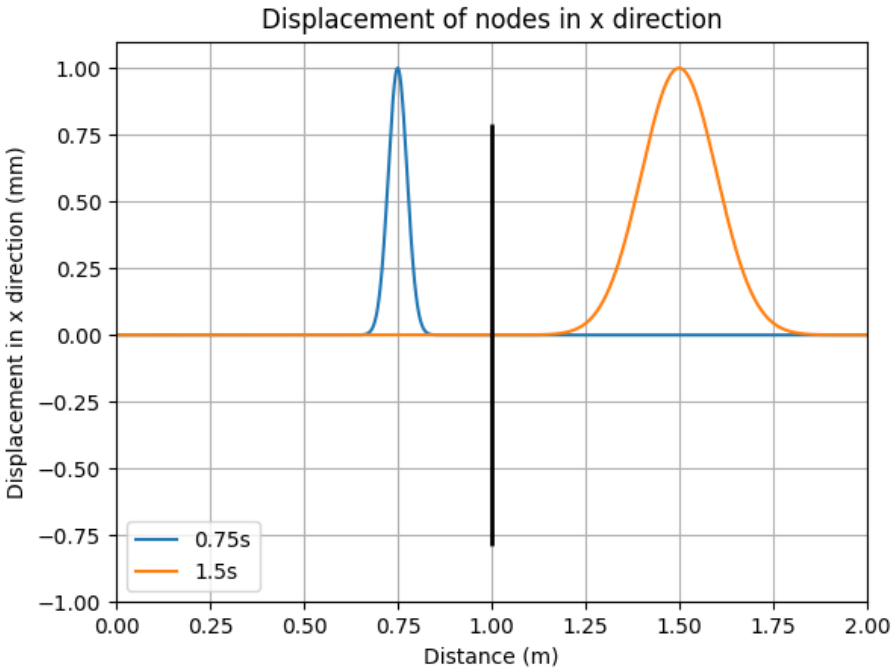


Figure 4.11: Test 5 results. Y axis represents the displacement in x direction of all centralized in y direction as in 4.6 at 1.5 seconds in blue and at 2.5 seconds in orange. X axis represents the length of the model used to reproduce the interface between two semi-infinite half-space. The incident Gaussian pulse is prescribed on the left face of the model. As expected from the theory, there are no reflected waves and only one wave can be seen at 2.5 seconds, which corresponds to the transmitted wave.

4.1.3 Test 6, 7, 8 and 9. - P Wave Gaussian Beam travelling through a interface between two semi-spaces.

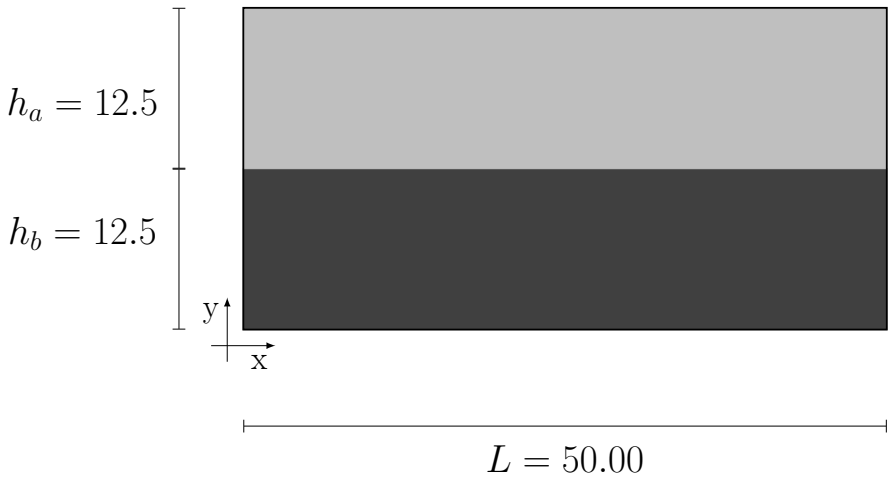


Figure 4.12: Representation of the model used in the tests 6, 7, 8 and 9 where the material properties used are available in table 4.5, where the gray material has properties V_{p1}, V_{s1}, ρ and the black material has properties V_{p2}, V_{s2} and ρ.

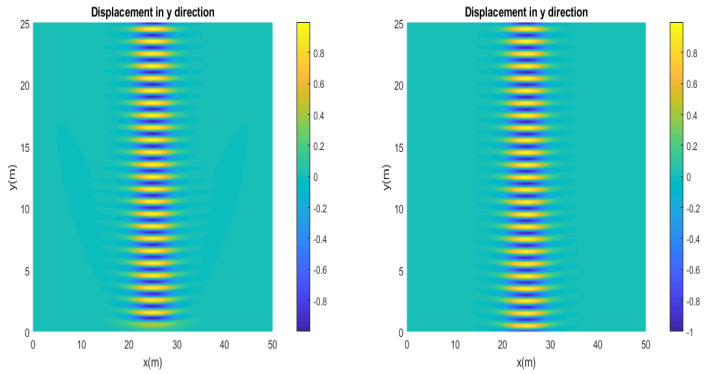
The numerical results from this subsection studies infinity media and adjoining semi-infinite medias subjected to a Gaussian beam. Four tests are presented, were the two latter ones presents a interface between different materials to study the reflection and transmission phenomena with information presented in table 4.5. The geometry of the model, the center of the Gaussian beam and the time were the experiment ends were chosen such that the beam, or it's reflections, does not interacts with the lateral borders of the model. Also, plane strain was assumed. The model consists of 50 m in x direction and 25 m in the y direction and the origin of the coordinate system is in the bottom left vertex of the geometry as in 4.13a. The Gaussian beam was reproduced as prescribed displacement field on the upper face of the model 4.12, it propagates in negative y direction with the y polarity and the other three faces have stress-free conditions.

Input	Test 6	Test 7	Test 8	Test 9
V_p[m/s]	1	1	1	1
V_{s1}[m/s]	0.5	0.5	0.5	0.5
V_{p2}[m/s]	1	1	3	3
V_{s2}[m/s]	0.5	0.5	2	2
ρ[Kg/m^3]	1	1	1	1
Angle	0^o	30^o	15^o	40^o
Gaussian Beam Center[m]	25	25	5	8.3
Element mesh	999x499	999x499	999x499	999x499
Time[s]	24.75	24.75	15.75	29.75
Time Steps	24750	24750	15750	29750

Table 4.5: Information of tests 6, 7, 8 and 9. V_p and V_s are the elastic wave velocity with polarity p and s, respectively. ρ is the density of the material. Model discretization represents the number of elements in x and y directions, respectively. Angle is related to the incident angle of the gaussian beam related with the y axis. The Gausian beam center was chosen in order to reduce the possibility of the wave hitting the surfaces. Time is the simulated time that was reproduced using the number of time steps on the table. V_{p1}, V_{s1}, ρ_1 are properties of the gray media 4.8 and V_{p2}, V_{s2}, ρ_1 are properties of the black media.

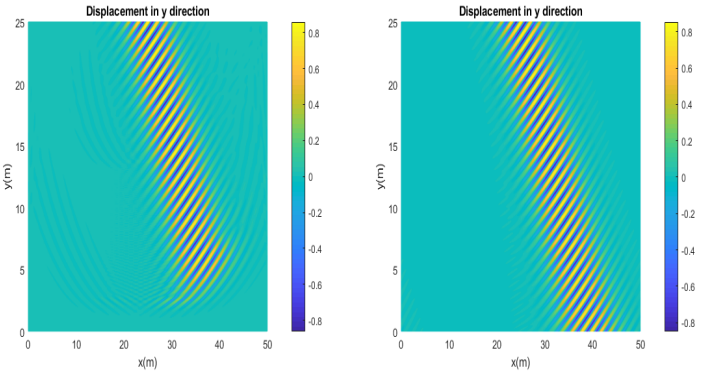
The objective of test 6 and 7 is to evaluate the quality of the representation of the Gaussian beam in a homogeneous media, where the Gaussian beam was applied with 0 and 30 degrees to the y direction, respectively. On those tests, the gray and black medias in 4.12 have the same properties as in 4.5. V_p was measured with the values of the position of the fourth peak of the beam at 10 and 11 seconds as represented in

figures in test 6. The obtained value was approximately 1.001 m/s which approximately reproduces the elastic property V_{p1} and V_{p2} used in the input, that is another indication of the formulation accuracy. Then, the overall quality of the result was studied with the values of the displacement at the entire model at time 24,75s.



(a) (b)

Figure 4.13: Results test 6 with properties and parameters as in table. (a) shows the displacement result of our simulation in the y direction at time 24.75s. (b) shows the analytical result of the displacement in y direction obtained from Fourier analysis. In this test was possible to see a traveling Gaussian beam in a homogeneous media.



(a) (b)

Figure 4.14: Results test 4.2 with properties and parameters as in table. (a) shows the displacement result of our simulation in the y direction at time 24.75s. (b) shows the analytical result of the displacement in y direction obtained from Fourier analysis. In this test was possible to see a traveling Gaussian beam in a homogeneous media.

In tests 8 and 9, the Gaussian beam was applied to a heterogeneous media and interface phenomena such as reflection, transmission, full reflection and evanescent waves were observed. The geometry of the model is the same as in tests 6 and 7, but the materials configuration is different. Above 12.5m in y direction as in figure 4.14a the input properties are V_{p1} and V_{s1} as in 4.5. Below this interface the properties used as

V_{p2} and V_{s2} of the same table. The critical angle between both medias as in table 4.5 was calculated with Snell's law and the value obtained was approximately 19.47 degrees. Henceforth, the test 8 and test 9 were performed with angles of 15 and 40 degrees in order to observe different phenomena. An incident angle lower than the critical angle allows the simulation to reproduce reflection and transmission as in figure 4.14a. An angle higher than the critical angle allows the observation of full reflection and evanescent waves as in figure 4.17a. Both results are in accordance to the theory strengthening the case for the simulation robustness.

The observed displacement pattern in figure 4.14a is caused by five different waves. Above the interface, at 12.5m in y direction, occurs the superposition of the incident P wave, the reflected P and Sv waves as in figure 4.15. Below this interface, occurs the superposition of the transmitted P and SV waves represented in the same figure.

In test 9, where full reflection is present, occurs the superposition of the incident P wave, the reflected P and Sv waves above the interface, as in test 4.3. Moreover, bellow the interface, evanescent waves are reproduced.

All tests in this subsection were compared to analytical result from Fourier analysis (LEIDERMAN; BRAGA; BARBONE, 2005). Through visual inspection we infer that the results are qualitatively close since both the direction and wave amplitudes are similar. Naturally, our results are obtained in time domain where wave front is present. We would have to wait a long time for the time domain solution to reproduce exactly the frequency domain solution, which is not possible in the model used in this test since the wave front would interact with the model borders.

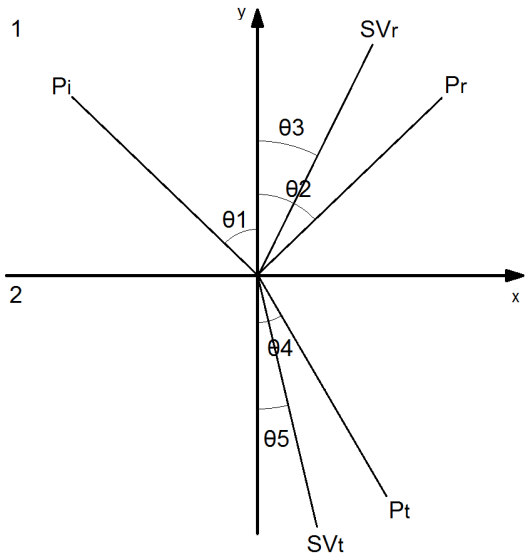
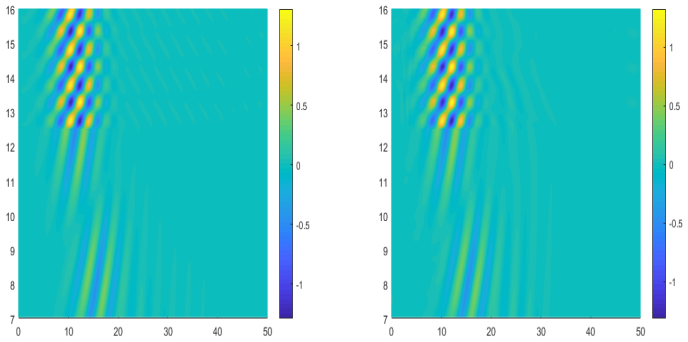
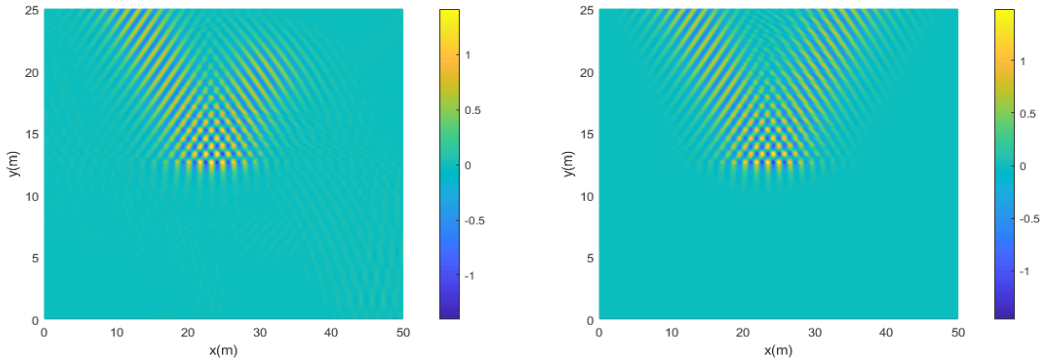


Figure 4.15: Representation of the physical problem and the waves caused by the incident P wave (Pi), as it travels through the interface between medias 1 and 2. Where SVr and Pr are the reflected SV and P waves, and the SVt and Pt are the transmitted SV and P waves.



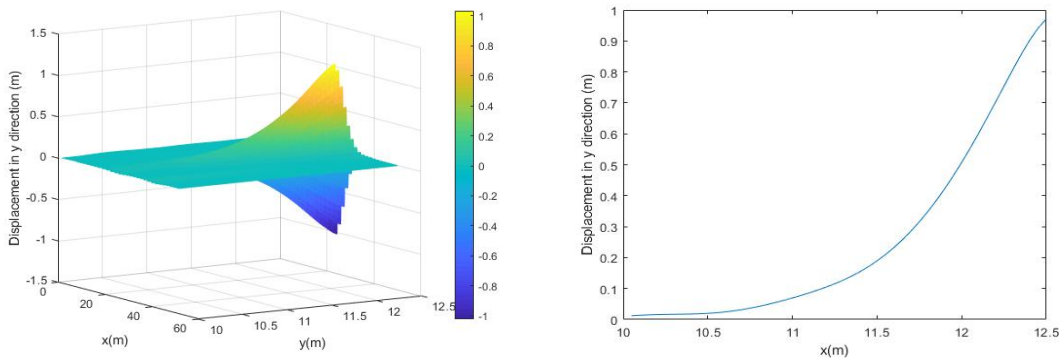
(a) (b)

Figure 4.16: Results test 4.3 with properties and parameters as in table. (a) shows the displacement result of our simulation in the y direction at time 15.75s. (b) shows the analytical result of the displacement in y direction obtained from Fourier analysis. In this test was possible to see the superposition of the incident P waves (Pi) reflected P waves (Pr) and reflected SV waves (SVr) in the upper half of the images. Also, the superposition of the transmited P waves (Pt) and the transmitted SV waves (SVt) on the bottom half of the images.



(a) (b)

Figure 4.17: Results test 4.4 with properties and parameters as in table. (a) shows the displacement result of our simulation in the y direction at time 29.75s. (b) shows the analytical result of the displacement in y direction obtained from Fourier analysis. In this test was possible to see the superposition of the incident P waves (Pi) reflected P waves (Pr) and reflected SV waves (SVr) in the upper half of the images, the full reflection phenomena. Also, we can see the evanescent waves in the region closer to the interface of both medias.



(a) (b)

Figure 4.18: Results test 4.4 below the interface to higlight the evanescent waves at 29.75s. (a) shows the displacement result of our simulation in the y direction with values of y between 10m and 12.5m. (b) Shows the displacement in y direction at 29.75 seconds for x roughly 30m, where the an approximately exponentially decaying displacement pattern typical of evanescent waves is detected.

4.1.4 Test 10 - Simulation in Layered media and effective wave speed.

In this last subchapter, the ultrasonic test from (ZHU; SHAN, 2016) was reproduced in two different models. In test 10, a model that represents an synthetic laminated material was subjected to the ultrassonic measurement whose results were compared to backus average prediction for a laminated medium composed of transversely isotropic materials.

A image obtained with the segmentation of a BS microtomography with resolution of 2μm was used for test 11 where the geometry of the pores and the matrix were used to simulate different contrasts using synthetic material.

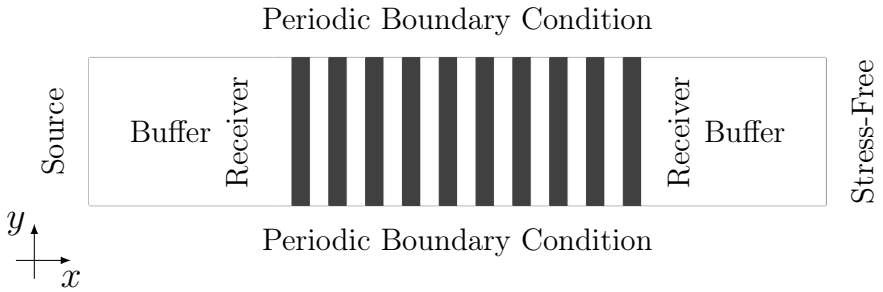


Figure 4.19: Representation of the model arrangement used in test 10. Since periodic boundary condition were applied in the upper and bottom faces as in figure, each square represents a layer infinitely large in y direction as in the image. The actual model is composed of 100 layers.

The ultrassonic transmisison test from (ZHU; SHAN, 2016) attains the effective elastic properties using the time required for a Gaussian pulse to cross the inspected media. This time is measured with two receivers that are positioned before and after the model in the wave propagation direction of interest as in 4.19, where t_1 and $t2$ are the times where the peack is recorded in the left and right receivers, respectively. Those receivers capture the mean value of the displacements at the faces of the model on each time step. And the time delay is calculated using the peak values of both receivers which were recorded as in 4.20. Wave velocity is estimated with the time delay and the length of the image, L, between receivers in the propagation direction of the wave 4.4, where E_{vp} is the effective V_p obtained in the transmission test. The Gaussian pulse is generated with prescribed displacement boundary condition at the left face of the model where the source is represented as in 4.19 and the right face of the model is in stress-free condition, on the upper and bottom faces to this direction on the 2D plane we applied periodic boundary conditions. This procedure can be replicated to any polarity of the wave changing the polarity of the Gaussian pulse and the direction that the mean displacement is recorded. In our tests, the Gaussian pulse displacement was applied in x direction, which as the same direction that the receivers measured the displacement. Those choices allowed the estimation of the model's primary velocity.

$$E_{vp} = L/(t_1 - t_2) \tag{4.4}$$

To minimize effects of border, two buffer zones with the size of the elastic wave length

after and before the receivers were implemented. The wavelengths used were at least ten times larger than approximated value of the characteristic length which was arbitrary chosen. The reason for this discrepancy is that we want to reproduce a result that respects the long-wavelength approximation (SRIVASTAVA, 2015) which allows the comparison with bakus average in test 10 and the static simulation in test 11. Wavelengths close to the characteristic length of the sample leads to a wave propagation phenomena where scattering phenomena starts to be more dominant, effective elastic velocities dependant on the wave frequency, which we want to avoid. The choice of this characteristic length value involved considerations of the models geometry and concern to adopt a conservative approach.

In test 10, our laminated model has 100 layers with two different materials alternatively arranged as in 4.19, each layer has 0.02m, two buffer zones with width of 50 layers were added to the model. The Gaussian pulse width used to measure the effective V_p was 50 times larger than the width of the layers. The different synthetic materials used, the simulated V_p estimation and the Bakus average prediction are shown in table 4.6. V_{p1} , V_{s1} are properties of the black layer and V_{p2} , V_{s2} are properties of the white layer.

$$C = (f_1\rho_1 V_{p1}^2 + f_2\rho_2 V_{p2}^2)^{-1} \tag{4.5}$$

$$\rho_m = f_1/\rho_1 + f_2/\rho_2 \tag{4.6}$$

$$Vp_{Backus} = \sqrt{C/\rho_m} \tag{4.7}$$

Where C is the effective elastic parameter approximation from (BACKUS, 1962) in the normal direction to the layers surface, Vp_{Backus} is the V_p velocity prediction obtained from backus average in the direction and polarity normal to the layers surface. f_1 and f_2 are the volume fractions from the the black and white layers, respectively. ρ_m is the average density.

Test	V_{p1}	V_{s1}	V_{p2}	V_{s2}	E_{vp}	Backus	Ratio
Test 10.1	2	1	2	1	2.008032	2.0	1.004016
Test 10.2	1	0.5	2	1	1.265822	1.264911	1.000720
Test 10.3	0.5	0.25	2	1	0.683994	0.685994	0.997084
Test 10.4	0.2	0.1	2	1	0.244738	0.281439	0.869595
Test 10.5	0.04	0.02	2	1	0.051324	0.056557	0.907472
Test 10.6	0.02	0.01	2	1	0.025803	0.028282	0.912347
Test 10.7	0.002	0.001	2	1	0.002570	0.002828	0.908728

Table 4.6: Information of tests in m/s. E_{vp} is the effective V_p obtained from the simulation, Backus is the Backus average prediction for V_p .

In figure 4.20, some phenomena presented in our simulations becomes clearer. When the Gaussian Wave hits the sample, reflections began to occur. Those reflections are caused by waves impinging interfaces with different stiffness and when they encounter the source boundary, some of the reflected waves have opposite polarity, leading to the negative values on the first receiver observed in roughly 2 seconds. Another phenomena is the reflection of the Gaussian wave that happens on the stress-free boundary, leading to the second orange and blues peeks. By the time this phenomena happens, the measurement already took place, since it happened approximately, in 4.20, between 1 and 3 seconds. So, it doesn't affect the goal of the simulation. One of the reasons the bufferzone was used is exactly to isolate the detection of those reflected waves and the measurement of the elastic velocity.

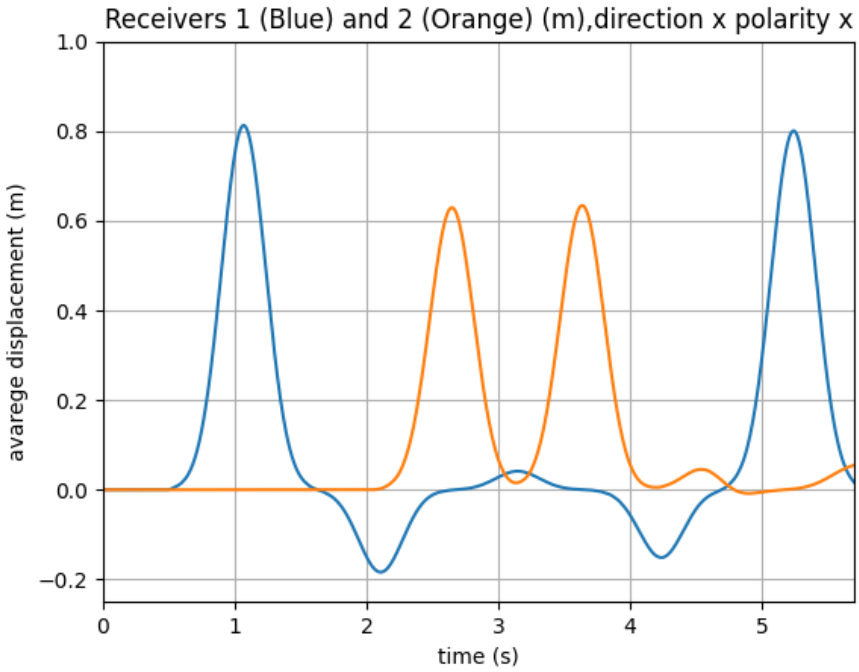


Figure 4.20: Average of displacements values in x direction with x polarity recorded on the receivers as in figure 4.20 to obtain the result of the the wave propagation transmission experiment in table 4.7. In blue it is represented the displacements on the left receiver and in orange the values on the right receiver.

4.1.5 Test 11 - Simulation in microtomography image and effective wave speed.

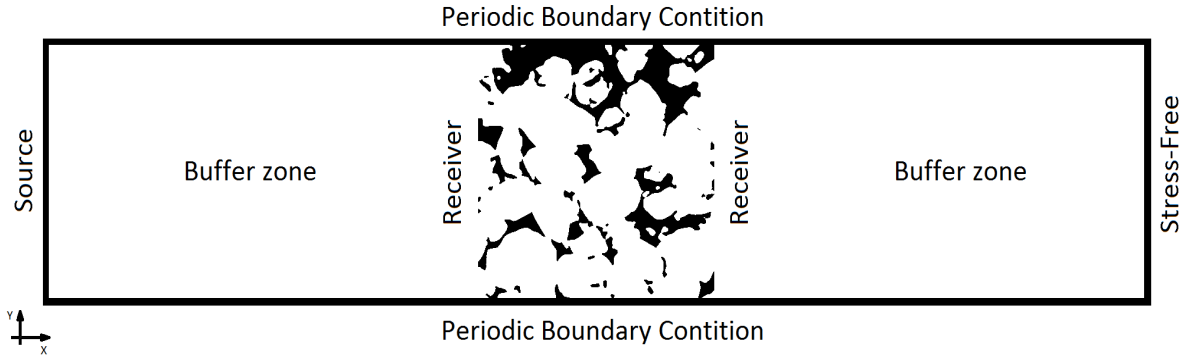


Figure 4.21: Image used to construct the model for the simulations in test 11. The source of the gaussian pulse where the displacement is prescribed is located on the left face of the domain, on the right face is in stress-free condition. On the upper and bottom faces, Periodic Boundary conditions are applied.

In test 11, different contrasts of synthetic material were tested using the geometry of a segmented image obtained from the microtomography of a Berea Stripe sample 4.22.

The results from the transmission test were compared to a static homogenizing program (LOPES, Pedro C. F. et al., 2022) and are presented in 4.7. Where V_{p1} , V_{s1} are properties of the material that uses the pore geometry from 4.22 and V_{p2} , V_{s2} are properties of the material that uses the matrix geometry from the same image.

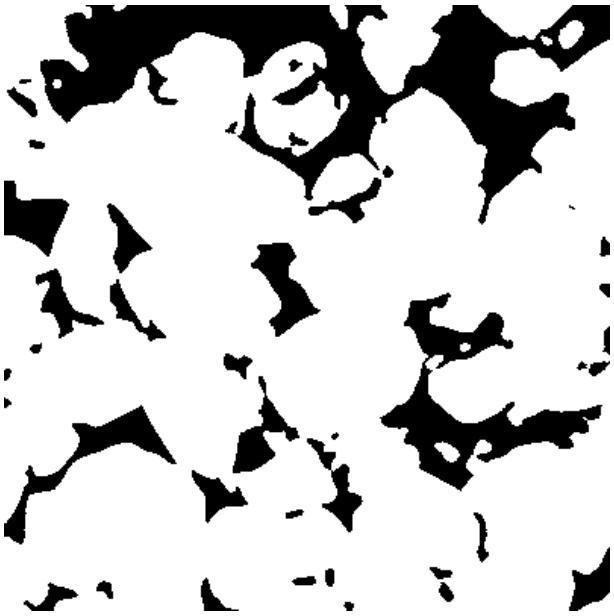


Figure 4.22: Image used to construct the model for the simulations in test 11.

Test	V_{p1}	V_{s1}	V_{p2}	V_{s2}	static E_{vp}	Transient E_{vp}	Accuracy
Test 11.1	2	1	2	1	1.999	2.002	99.8
Test 11.2	1	0.5	2	1	1.678	1.650	98.33
Test 11.3	0.4	0.2	2	1	1.268	1.169	92.19
Test 11.4	0.2	0.1	2	1	1.067	0.895	83.88
Test 11.5	0.04	0.02	2	1	0.929	0.351	37.78
Test 11.6	0.02	0.01	2	1	0.923	0.154	16.78

Table 4.7: Information of test 11.

On both tests, there is a tendency of higher accuracy of the simulation as the contrast of elastic properties decreases. Which implies that models with a phase representing a pore needs more attention because of it's high contrast between the phases. To solve this issue, our group proposes that, in future implementations, the pore phase is not modelled or discretized. This choice transforms the interfaces between the matrix and the pore in borders of the domain. The reason for this proposition is that FEM formulation

naturally resolves stress-free conditions on the border of the simulated domain, mitigating the contrast on the simulated phases, reducing accuracy loss.

Chapter 5

Conclusion

We've implemented a memory efficient 2D pixel-based explicit finite element method and thoroughly testing we've endorsed the simulation robustness. We've reproduced a benchmark result and studied the response of our simulation in different scenarios such as a travelling planar wave in a infinite media, in a interface of semi-infinite medias. Also, a Gaussian beam in a infinite media, in a interface of semi-infinite medias, allowing the reproduction of reflected and transmitted waves, including evanescent waves.

With this tool the transmission pulse test was reproduced using a simple model to emulate an inclusion embedded in a matrix and μCT image of a Berea Stripe sample, where we studied the values obtained for different contrast of elastic properties and compared to analytic results and static FEM formulation, respectively. Our group proposes that, in next studies, the pores of our models are not discretized exploiting FEM features and mitigating the effects of contrast in our simulations.

5.1 Next steps

This work is the first step in order to achieve an a memory efficient, voxel-based transient FEM that exploits parallel computing. Which will be used to reproduce transmission pulse test to acquire elastic properties from μCT rock samples. So, the first obvious step would be to reproduce those tests and simulations using a 3D formulation.

There are other possible improvements that can be explored. We could use elements based on quadratic approximations used on the element formulation and increase the number of Gauss points applied on the numerical integration to verify the possible gains on accuracy. There is an alternative approach to deal with porous media, we can exploit

our formulation property of easily reproducing boundary phenomena of the model in a way to better reproduce pores. Without modeling the pores, their boundary becomes the boundary of model reproducing reflection on a stress-free condition. There are several alternatives to decrease the computational cost related to the buffer zone at the ultrassonic test, such as discretitizing elements that models more than one pixel on the principal direction of the test, we can precomputate part of the pulse movement on the first buffezone, we can implement boundary conditions in the y direction in order to use only one buffer zone provided that's the principal pulse only goes in one direction inside the sample, we could try to use the derivative of the gaussian as the stress boundary condition on the source.

References

ABDULLE, Assyr; GROTE, Marcus J; JECKER, Orane. Finite element heterogeneous multiscale method for elastic waves in heterogeneous media. **Computer Methods in Applied Mechanics and Engineering**, Elsevier, v. 335, p. 1–23, 2018.

ANDRÄ, Heiko et al. Digital rock physics benchmarks Part I: Imaging and segmentation. **Computers & Geosciences**, Elsevier, v. 50, p. 25–32, 2013.

_____. Digital rock physics benchmarks Part II: Computing effective properties. **Computers & Geosciences**, Elsevier, v. 50, p. 33–43, 2013.

ARNS, Christoph H et al. Computation of linear elastic properties from microtomographic images: Methodology and agreement between theory and experiment. **Geophysics**, Society of Exploration Geophysicists, v. 67, n. 5, p. 1396–1405, 2002.

AURIAULT, J-L; BOUTIN, Claude. Long wavelength inner-resonance cut-off frequencies in elastic composite materials. **International Journal of Solids and Structures**, Elsevier, v. 49, n. 23-24, p. 3269–3281, 2012.

BACKUS, George E. Long-wave elastic anisotropy produced by horizontal layering. **Journal of Geophysical Research**, Wiley Online Library, v. 67, n. 11, p. 4427–4440, 1962.

BEZ, Luiz F. et al. An Image-Based Explicit Matrix-Free FEM Implementation with Lumped Mass for NMR Simulations. **Transport in Porous Media**, v. 147, n. 01, p. 35–57, 2023. DOI: 10.1007/s11242-022-01894-1.

BIOT, Maurice A. Theory of propagation of elastic waves in a fluid-saturated porous solid. II. Higher frequency range. **The Journal of the acoustical Society of america**, Acoustical Society of America, v. 28, n. 2, p. 179–191, 1956.

BOUTIN, C; AURIAULT, JL. Rayleigh scattering in elastic composite materials. **International journal of engineering science**, Elsevier, v. 31, n. 12, p. 1669–1689, 1993.

COURANT, Richard; FRIEDRICHS, Kurt; LEWY, Hans. Über die partiellen Differenzengleichungen der mathematischen Physik. **Mathematische annalen**, Springer, v. 100, n. 1, p. 32–74, 1928.

ESHELBY, John Douglas. The determination of the elastic field of an ellipsoidal inclusion, and related problems. **Proceedings of the royal society of London. Series A. Mathematical and physical sciences**, The Royal Society London, v. 241, n. 1226, p. 376–396, 1957.

FAISAL, Titly Farhana et al. Numerical prediction of carbonate elastic properties based on multi-scale imaging. **Geomechanics for Energy and the Environment**, Elsevier, v. 20, p. 100125, 2019.

FERREIRA, Lucas Prata et al. An image-based numerical homogenization strategy for the characterization of viscoelastic composites. **International Journal of Solids and Structures**, Elsevier, v. 267, p. 112142, 2023.

FREHNER, Marcel et al. Comparison of finite difference and finite element methods for simulating two-dimensional scattering of elastic waves. **Physics of the earth and planetary interiors**, Elsevier, v. 171, n. 1-4, p. 112–121, 2008.

GARBOCZI, E.J.; DAY, A.R. An algorithm for computing the effective linear elastic properties of heterogeneous materials: Three-dimensional results for composites with equal phase poisson ratios. **Journal of the Mechanics and Physics of Solids**, v. 43, n. 9, p. 1349–1362, 1995. ISSN 0022-5096. DOI: https://doi.org/10.1016/0022-5096(95)00050-S. Disponível em: <https://www.sciencedirect.com/science/article/pii/002250969500050S>.

HASHIN, Zvi; SHTRIKMAN, Shmuel. A variational approach to the theory of the elastic behaviour of multiphase materials. **Journal of the Mechanics and Physics of Solids**, Elsevier, v. 11, n. 2, p. 127–140, 1963.

HILL, Rodney. A self-consistent mechanics of composite materials. **Journal of the Mechanics and Physics of Solids**, Elsevier, v. 13, n. 4, p. 213–222, 1965.

HUGHES, TJR; LIU, WK. lmplicit˜ Explicit finite elements in transient analysis: stability theory. **Journal of applied Mechanics**, v. 45, 1978.

JACOB FISH, Ted Belytschko. **A first course in finite elements**. [S.l.]: John Wiley , Sons Ltd, 2007. ISBN 9780470035801,9780470510841,0470035803. Disponível em: <http: //gen.lib.rus.ec/book/index.php?md5=578c3bbfa7c71440a2e80f3cf592b0a5>.

KIRAN, U.; GAUTAM, S.S.; SHARMA, D. GPU-based matrix-free finite element solver exploiting symmetry of elemental matrices. **Computing**, Pleiades Publishing, v. 102, p. 1941–1965, 2020. DOI: 10.1007/s00607-020-00827-4.

LEIDERMAN, Ricardo; BRAGA, Arthur MB; BARBONE, Paul E. Scattering of ultrasonic waves by defective adhesion interfaces in submerged laminated plates. **The Journal of the Acoustical Society of America**, Acoustical Society of America, v. 118, n. 4, p. 2154–2166, 2005.

LOPES, Pedro C. F. et al. A vectorized assembly-free FEM solver for image-based numerical homogenization. **Journal of the Brazilian Society of Mechanical Sciences and Engineering**, v. 44, n. 8, p. 343, jul. 2022. ISSN 1806-3691. DOI: 10.1007/s40430-022-03640-6. Disponível em: <https://doi.org/10.1007/s40430-022-03640-6>.

LOPES, Pedro C.F. et al. Simulation toolkit for digital material characterization of large image-based microstructures. **Computational Materials Science**, v. 219, p. 112021, 2023. ISSN 0927-0256. DOI: https://doi.org/10.1016/j.commatsci.2023.112021. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0927025623000150>.

LOPES, Pedro Cortez Fetter et al. A GPU implementation of the PCG method for large-scale image-based finite element analysis in heterogeneous periodic media. **Computer Methods in Applied Mechanics and Engineering**, v. 399, p. 115276, 2022. ISSN 0045-7825. DOI: https://doi.org/10.1016/j.cma.2022.115276. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0045782522003978>.

_____._____. **Computer Methods in Applied Mechanics and Engineering**, v. 399, p. 115276, 2022. ISSN 0045-7825. DOI: https://doi.org/10.1016/j.cma.2022.115276. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0045782522003978>.

MARTÍNEZ-FRUTOS, Jesús; HERRERO-PÉREZ, David. Efficient matrix-free GPU implementation of Fixed Grid Finite Element Analysis. **Finite Elements in Analysis and Design**, v. 104, p. 61–71, 2015. ISSN 0168-874X. DOI: https://doi.org/10.1016/j.finel.2015.06.005. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0168874X1500092X>.

MAZZOCHI, Rudimar; SUAREZ, Oscar Alfredo Garcia de; ROSSI, Rodrigo. Numerical approach of free and forced elastic vibrations using high-regularity Hermitian partition of unities. **Journal of the Brazilian Society of Mechanical Sciences and Engineering**, Springer, v. 42, n. 6, p. 1–19, 2020.

MICHEL, J.C.; MOULINEC, H.; SUQUET, P. Effective properties of composite materials with periodic microstructure: a computational approach. **Comput. Methods Appl. Mech. Engrg.**, Elsevier, v. 172, p. 109–143, 1999. DOI: 10.1016/S0045-7825(98)00227-8.

MILED, K; SAB, Karam; LE ROY, Robert. Effective elastic properties of porous materials: Homogenization schemes vs experimental data. **Mechanics Research Communications**, Elsevier, v. 38, n. 2, p. 131–135, 2011.

MORI, Tanaka; TANAKA, Kohichi. Average stress in matrix and average elastic energy of materials with misfitting inclusions. **Acta metallurgica**, Elsevier, v. 21, n. 5, p. 571–574, 1973.

NASSAR, Hussein; HE, Q-C; AUFFRAY, Nicolas. On asymptotic elastodynamic homogenization approaches for periodic media. **Journal of the Mechanics and Physics of Solids**, Elsevier, v. 88, p. 274–290, 2016.

NEMAT-NASSER, Sia; SRIVASTAVA, Ankit. Bounds on effective dynamic properties of elastic composites. **Journal of the Mechanics and Physics of Solids**, Elsevier, v. 61, n. 1, p. 254–264, 2013.

_____. Overall dynamic constitutive relations of layered elastic composites. **Journal of the Mechanics and Physics of Solids**, Elsevier, v. 59, n. 10, p. 1953–1965, 2011.

PHAM, Kim; KOUZNETSOVA, Varvara G; GEERS, Marc GD. Transient computational homogenization for heterogeneous materials under dynamic excitation. **Journal of the Mechanics and Physics of Solids**, Elsevier, v. 61, n. 11, p. 2125–2146, 2013.

REUSS, András. Berechnung der fließgrenze von mischkristallen auf grund der plastizitätsbedingung für einkristalle. **ZAMM-Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik**, Wiley Online Library, v. 9, n. 1, p. 49–58, 1929.

S. S. QUEK, G.R. Liu. **The Finite Element Method: A Practical Course**. [S.l.]: Butterworth-Heinemann, 2003. ISBN 9780750658669,9781417505593,0750658665. Disponível em: <http://gen.lib.rus.ec/book/index.php?md5=85760afdc4189ab75d846ee5fd53d6aa>.

SAENGER, EH; KRÜGER, OS; SHAPIRO, SA. Numerical considerations of fluid effects on wave propagation: Influence of the tortuosity. **Geophysical Research Letters**, Wiley Online Library, v. 31, n. 21, 2004.

SAENGER, Erik H. Numerical methods to determine effective elastic properties. **International Journal of Engineering Science**, Elsevier, v. 46, n. 6, p. 598–605, 2008.

SAENGER, Erik H; CIZ, Radim et al. Finite-difference modeling of wave propagation on microscale: A snapshot of the work in progress. **Geophysics**, Society of Exploration Geophysicists, v. 72, n. 5, sm293–sm300, 2007.

SAENGER, Erik H; ENZMANN, Frieder et al. Digital rock physics: Effect of fluid viscosity on effective elastic properties. **Journal of Applied Geophysics**, Elsevier, v. 74, n. 4, p. 236–241, 2011.

SAENGER, Erik H; GOLD, Norbert; SHAPIRO, Serge A. Modeling the propagation of elastic waves using a modified finite-difference grid. **Wave motion**, Elsevier, v. 31, n. 1, p. 77–92, 2000.

SAENGER, Erik H; KRÜGER, Oliver S; SHAPIRO, Serge A. Effective elastic properties of randomly fractured soils: 3D numerical experiments. **Geophysical Prospecting**, European Association of Geoscientists & Engineers, v. 52, n. 3, p. 183–195, 2004.

SCHINDELIN, Johannes et al. Fiji: an open-source platform for biological-image analysis. **Nature Methods**, v. 9, n. 7, p. 676–682, jul. 2012. ISSN 1548-7105. DOI: 10.1038/nmeth.2019. Disponível em: <https://doi.org/10.1038/nmeth.2019>.

SCHNEIDER, Caroline A.; RASBAND, Wayne S.; ELICEIRI, Kevin W. NIH Image to ImageJ: 25 years of image analysis. **Nature Methods**, v. 9, n. 7, p. 671–675, jul. 2012. ISSN 1548-7105. DOI: 10.1038/nmeth.2089. Disponível em: <https://doi.org/10.1038/nmeth.2089>.

SRIVASTAVA, Ankit. Elastic metamaterials and dynamic homogenization: a review. **International Journal of Smart and Nano Materials**, Taylor & Francis, v. 6, n. 1, p. 41–60, 2015.

UYANIK, Osman et al. Prediction of mechanical and physical properties of some sedimentary rocks from ultrasonic velocities. **Bulletin of Engineering Geology and the Environment**, Springer, v. 78, p. 6003–6016, 2019.

VAN PAMEL, Anton et al. Finite-element modelling of elastic wave propagation and scattering within heterogeneous media. **Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences**, The Royal Society Publishing, v. 473, n. 2197, p. 20160738, 2017.

VIANNA, Rafael S et al. Computing effective permeability of porous media with fem and micro-ct: an educational approach. **Fluids**, MDPI, v. 5, n. 1, p. 16, 2020.

VOIGT, Woldemar. Ueber die Beziehung zwischen den beiden Elasticitätsconstanten isotroper Körper. **Annalen der physik**, Wiley Online Library, v. 274, n. 12, p. 573–587, 1889.

WILLIS, John R. Effective constitutive relations for waves in composites and metamaterials. **Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences**, The Royal Society Publishing, v. 467, n. 2131, p. 1865–1879, 2011.

_____. Exact effective relations for dynamics of a laminated body. **Mechanics of Materials**, Elsevier, v. 41, n. 4, p. 385–393, 2009.

_____. The construction of effective relations for waves in a composite. **Comptes Rendus Mécanique**, Elsevier, v. 340, n. 4-5, p. 181–192, 2012.

WU, Shen R. Lumped mass matrix in explicit finite element method for transient dynamics of elasticity. **Computer Methods in Applied Mechanics and Engineering**, Elsevier, v. 195, n. 44-47, p. 5983–5994, 2006.

ZHU, Wei; SHAN, Rui. Digital core based transmitted ultrasonic wave simulation and velocity accuracy analysis. **Applied Geophysics**, Springer, v. 13, n. 2, p. 375–381, 2016.

ZHU, Wei; SHAN, Rui et al. An efficiently dynamic stress strain simulation method on digital rock. **Journal of Applied Geophysics**, Elsevier, v. 147, p. 10–15, 2017.

ZIMMER, JE; COST, James R. Determination of the elastic constants of a unidirectional fiber composite using ultrasonic velocity measurements. **The Journal of the Acoustical Society of America**, Acoustical Society of America, v. 47, 3B, p. 795–803, 1970.

APPENDIX A – Image processing and segmentation

In this appendix, we are going to explore the segmentation step employed to obtain the model used at test 6.

First, microtomography was used to obtain images from a plug from Berea Sripe (BS) sandstone with 2 μm of resolution, generating multiple images that represent slices from the actual plug. The original image is dark and lack contrast between the different phases. So we need to use image processing to enhance the contrast, the brightness and to filter the noise of the image, increasing it's the overall quality in order to segment the pore phase from the sandstone. Hence, obtaining an approximation of the plug micro-structure.

In order to accomplish this task, we used the Fiji:ImageJ which is an open source image processing software(SCHINDELIN et al., 2012; SCHNEIDER; RASBAND; ELICEIRI, 2012). So, we applied normalization process to enhance the contrast between phases, also brightening the image, getting figure:

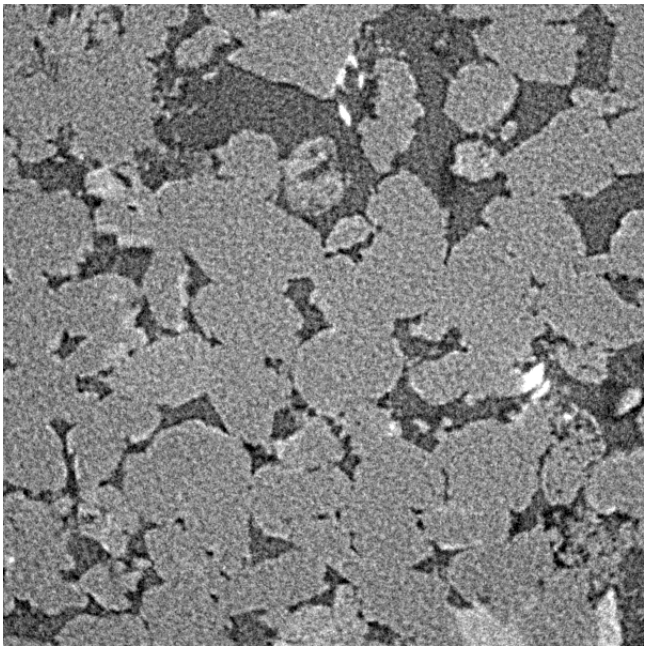


Figure A.1: BS

Normalization also additions noise to the image, then we need a filter to decrease this noise. We used the Non-Local means plugin followed by Unsharp mask to enhance the constrast between phases, leading to image A.2. After this step, we used threshhold to segmentate the pore and the matrix geometries, which were used on our simulations from test 6 at section 4. Lastly, the image was cropped, obtaining an image with 600x600 pixels A.3.

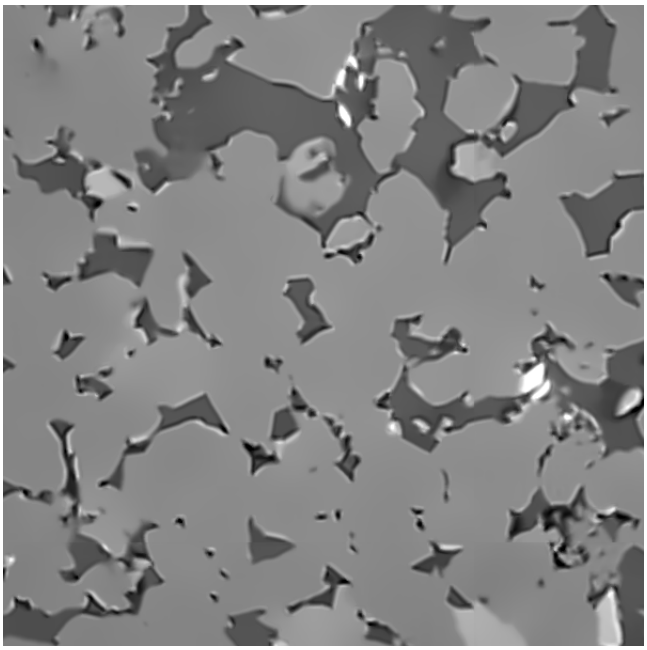


Figure A.2: BS

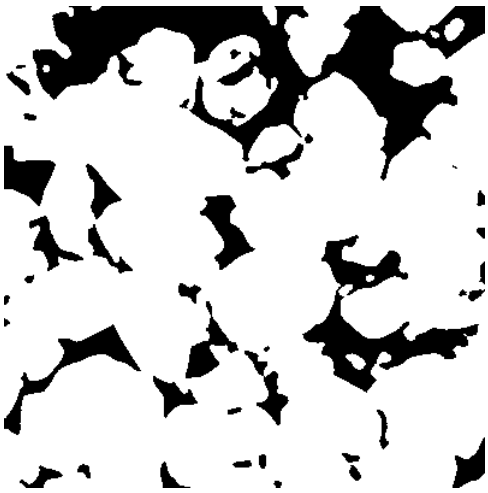


Figure A.3: BS