UCLouvain

École polytechnique de Louvain

Numerical simulation to compute pile driving into the ocean floor

Author: **Maud Schauss**
Supervisors: **Vincent Legat, Jonathan Lambrechts**
Readers: **Jean-Charles Delvenne, Nathan Coppin**
Academic year 2019–2020
Master [120] in Mathematical Engineering

Abstract

The installation of piles has become a real challenge in civil engineering, especially when it comes to the construction of marine structures. Ever-larger piles must be installed in extreme geotechnical conditions. During the installation process, the soil properties near the pile change significantly, affecting the density, the stresses, and the porosity of the soil.

To gain a better understanding of the behavior of the soil and the influence of the fluid, the pile driving process has been simulated numerically. To do so, different modeling concepts have been explored. The free surface of the fluid is modeled using an interface-tracking method. A simplified Laplacian smoothing method has been developed to adapt the interior nodes of the domain. The conservation equations are solved using the arbitrary Lagrangian-Eulerian description, allowing to take into account the deformation of the domain. The driving process is simulated and analyzed for different parameter choices. The model shows agreement with the results found in the literature.

Travail de fin d'étude fait en collaboration partielle avec Michel Henry

Nos deux travaux de fin d'études se basent sur des développements informatiques faits dans le cadre du même projet MigFlow sous la direction de Vincent Legat et Jonathan Lambrechts, même si les deux applications visées sont distinctes. Nos travaux de recherche ont été effectués en parallèle, mais de manière concertée dans leur équipe de recherche et certains développements ont été effectués de manière collaborative.

A l'issue de cette année, écrire deux manuscrits distincts de mémoire nous a semblé être à tous deux la meilleure manière d'acquérir une compétence essentielle pour notre formation. Ce choix a été fait de manière concertée avec nos promoteurs. Toutefois, certains aspects de notre démarche étaient inévitablement tellement proches que les deux chapitres consacrés à la méthode numérique pour traiter les surfaces libres sont identiques, même si la langue des deux manuscrits est différente.

Michel Henry
Maud Schauss

Juin 2020

Référence du mémoire de Michel Henry : [Henry 2020]

Acknowledgements

First of all, I would like to offer my special thanks to my master thesis supervisors, Vincent Legat and Jonathan Lambrechts, for their availability and advice. I would also like to thank Matthieu Constant and Nathan Coppin for their support and help throughout this year.

My sincere thanks goes as well to Jean-Charles Delvenne for accepting to read this thesis.

I would also like to acknowledge Maïté, Michelle and Antoine who took the time to read this thesis. Thank you for your precious feedback.

Last but not least, I would like to thank my family for their support and their constant source of inspiration.

Contents

1 Introduction **1**

2 State of the art **3**

3 Free surface **7**
3.1 Kinematic condition . . . 7
3.2 Dynamic condition . . . 11
3.3 Volume conservation . . . 13
3.4 Stick-slip flow . . . 15

4 Mesh regularization **21**
4.1 Elliptic grid generation . . . 22
4.2 Simplified elliptic grid generation . . . 22
4.3 Mesh regularization for the pile driving simulation . . . 27

5 Arbitrary Lagrangian-Eulerian Method **33**
5.1 Fundamental ALE equation . . . 34
5.2 Conservation equations . . . 35

6 Pile driving **37**
6.1 Experimental set-up . . . 37
6.2 Resolution method . . . 38
6.3 Pile velocity . . . 42
6.4 Tip shape . . . 43
6.5 Influence of the fluid on the penetration process . . . 46

7 Conclusion **50**

Bibliography **52**

Appendices **56**

Chapter 1

Introduction

For thousands of years, buildings and structures constructed on soft or saturated soil have been founded on piles driven deep into the ground. Those constructions can be observed throughout history all over the world, with well-known examples including the city of Venice, floating villages in South-East Asia, or Vindeby, the first offshore wind farm. With the drastic increase in ocean infrastructure development over the past 30 years, pile driving has become a real challenge in civil engineering. Methods must be found to install ever-larger piles in extreme geotechnical conditions. The construction of large marine structures such as wind turbines, bridges, and offshore oil and gas structures calls for a safe and cost-optimal installation of piles into the ocean floor, while avoiding pile damage during the installation.

To reach its final installation depth, the pile has to be driven through dense soil. This leads to significant changes of the soil properties near the pile, affecting the density, the stresses, and the porosity of the soil. The changes in the soil and the affected area around the pile during installation constitute remaining uncertainties in geotechnical engineering [Phuong 2019], increasing the challenges regarding the feasibility of the driving process. Therefore, correct numerical simulation of pile driving is extremely important to predict the behavior of the pile, the soil and the surrounding fluid flow during driving. Only in the case of meticulous numerical simulations can the secure installation of the pile be guaranteed. However, the complexity of fluid-grain mixes makes the simulation of such problems challenging. On one hand, the flow contains a continuous phase, which is the fluid, and on the other hand, it involves a discrete phase, which is the solid phase, i.e. the solid soil particles.

This master thesis focuses on the simulation of pile driving using the MigFlow[1] software. MigFlow is a software dedicated to the numerical simulation of immersed granular flows. It therefore allows to model multi-phase flows involving grains in a fluid. Immersed granular flows have many industrial applications, such as pharmaceutical, cosmetic and agro-food production, but they are also present in environmental phenomena including avalanches, lava flows and sediment transport. Due to the steady increase of new applications of immersed granular flows, researchers become more and more interested in understanding the physics of such flows in order to be able to simulate them numerically. With this goal, the MigFlow Project was brought to life.

[1] https://www.migflow.be/

The MigFlow software uses a multi-scale approach to model the fluid-grains mixture flow [Constant et al. 2018]. The solid phase is solved at the grain scale using a Discrete Element Method (DEM). The fluid phase is computed at larger scale using a Finite Elements Method (FEM) to solve the Navier-Stokes equations. The two phases are combined by the concept of porosity, which corresponds to the volume fraction occupied by the fluid.
The model is able to simulate complex problems with good accuracy, such as the falling of a swarm of particles into a fluid. Nevertheless, it still lacks the simulation of free surface flows, which is significant in many applications, for example in the simulation of waves in the ocean.

This master thesis aims to present an efficient implementation of the simulation of pile driving into the ocean floor, with the prospect of adding different parts to the MigFlow model. Various modelling concepts have been explored in order to develop the final simulation. These concepts are explained throughout this thesis.
First, a state of the art is presented, which explains the scientific background of free surface flows and the simulation of pile driving. In Chapter 3, the equations necessary to model free-surface flows are provided, and an explanation as to how they have been implemented in the Migflow software is given. Subsequently, the implementation is validated using the die-swell phenomenon. Due to the motion of the pile and the motion of the free surface, the computational mesh gets heavily deformed. To cope with these deformations, a method to regularize the mesh is proposed in Chapter 4. This method is necessary to maintain a valid mesh during the simulation. Chapter 5 addresses the *arbitrary Lagrangian-Eulerian* description used to solve the conservation equations of the immersed granular flow. This description allows to take into account the deformation of the mesh. Finally, Chapter 6 presents the simulation of pile driving by using the concepts laid out in Chapters 3 to 5. Using this simulation, different installation methods are simulated to examine the effects of these methods on the soil and the fluid around the driven pile.

Chapter 2

State of the art

Mathematical study of wave motion began at the end of the 18th century, with the work of Lagrange, who studied the behavior of the motion of fluids and introduced a kinematic condition that predicts the shape of the free surface [Lagrange 1781]. This condition is known today as *the free surface condition*. Since then, steady progress has been made in this field by many researchers. Two important theories have evolved in 19th century: the linear theory for small amplitude waves [Airy 1841] and the non-linear theory for waves in shallow depths [Gerstner 1809]. Using a perturbation series approach, Stokes obtained approximate solutions for non-linear wave motion [Stokes 1847]. However, the use of analytical techniques to these problems has been limited by the difficulties of non-linearities. The fact that the flow domain is not known *a priori* and that it is part of the solutions, makes the analytical solving of these equations very difficult. Therefore, theoretical studies for time-dependent and non-linear free-surface motions rely more on numerical simulations [Tsai and Yue 1996].

The development of numerical models truly began in the 1960s, with various approaches, approximations and implementations. The advent of numerical methods was accompanied by renewed interest in previously unsolvable problems, such the non-linear free surface equation [Ramaswamy 1990]. Since then, many numerical methods for treating moving interfaces have been developed. Muzaferija and Peric classify these methods into two broad categories: interface-capturing and interface-tracking methods [Muzaferija and Peric 1997].

In interface-capturing methods, the computation is performed in a fixed solution domain that extends over regions occupied by both fluids. These methods belong to the class of Eulerian methods, which are characterized by a stationary mesh.
One of the first interface-capturing method is the marker and cell (MAC) method that has been introduced by Harlow and Welch [Harlow and Welch 1965]. In the MAC method, particles are used as markers that locate the material in the mesh. This way, the markers move with the location of the free surface.
Another interface-capturing method is the Volume-of-fluid (VOF) method introduced by Hirt and Nichols [Hirt and Nichols 1981]. The interface is captured by calculating the concentration or the volume fraction of the fluids in each element of the computational mesh.
The major advantage of interface-capturing methods is that the numerical grid does not follow the deformation of the free surface. Therefore, it is very robust for problems containing complicated

boundary geometries and the fluid can undergo arbitrarily great distortions without loss of accuracy. The disadvantage of interface-capturing methods is the impossibility of defining the exact surface. Thus, in order to attain accurate results, grid refinements have to be done [Tezduyar 2006].

In interface-tracking methods, the numerical grid is adapted according to the shape and position of the free surface. The free surface is treated as a boundary of the computational domain, where the kinematic and dynamic boundary conditions are applied [Demirdzic et al. 1999]. This type of method makes it possible to define the shape of the free surface very precisely. Their major disadvantage is their incapability of following large distortions [Donea et al. 2004].

Different methods have been developed to overcome this problem and to allow large deformations of the domain, such as meshless methods (e.g the Smoothed Particle Hydrodynamics method [Monaghan 1992]), mesh-based particle methods (e.g. the Material Point Method [Bardenhagen and Kober 2004]) or Arbitrary Lagrangian-Eulerian (ALE) methods.
The ALE description combines the Lagrangian and the Eulerian description of motion. It consists in describing the fluid in a computational domain that moves arbitrarily and independently from the motion of its material points, thus maintaining a valid mesh. This domain is neither Lagrangian nor Eulerian but is called the reference domain.
First developments of the ALE method can be found in the works of Noh [Noh 1964] and Hirt *et al.* [Hirt, Amsden, and Cook 1974]. They established the ALE description in finite difference formats. The method has later been introduced in finite element format by Hughes *et al.* [Hughes, Liu, and Zimmermann 1981] in the context of incompressible, viscous flows. Since then, the ALE methods have found many application areas.
One important application area of ALE descriptions is the simulation of geotechnical engineering problems, especially when it comes to the simulation of penetration problems, as pile driving. If the soil is modeled as a continuum, the domain undergoes large deformations near the penetrating pile.

Several publications approach pile driving problems through analytical [Salgado, Mitchell, and Jamiolkowski 1997], experimental [Deeks, DJ White, and MD Bolton 2005] and numerical [Jiang, Yu, and Harris 2006] perspectives. When the problem is simulated numerically, granular environments can be represented either by using a continuum based representation or by a particle based representation.

In the past years, various authors have investigated pile penetration processes using continuum-based models, such as the Finite Element Method (FEM). The first finite-element model for the analysis of pile driving was presented by Mabsout and Tassoulas [Mabsout and Tassoulas 1994]. They analysed the problem by using a finite two-dimensional model. However, the applications were limited by the large deformation of the soil. To cope with these deformations, first applications of the ALE method in geomechanics were made by Di *et al.* [Di, Yang, and Sato 2007]. Henke and Grabe extended this model to allow three-dimensional analyses of different pile installation processes. The influence of driving a pile near existing piles was also studied by Henke and Grabe [Henke and Grabe 2009].
Although continuum-based models reduce the computational cost of the simulations, it fails in giving insight on the characteristics of the individual grains and its interactions with other grains and the fluid.

Another possibility to represent granular material is to use the Discrete Element Method (DEM). The method enables a more realistic modelling of soil medium by taking into account each grain. DEM modeling allows particle shapes and sizes to be varied and can cope with large deformations and contact changes. The DEM was intially introduced by Cundall and Stark [Cundall and Strack 1979]. Since then, the use of discrete elements in geotechnical problems is very popular. Huang and Ma were the first to apply DEM in the context of pile driving [Huang and Ma 1994]. Campos *et al.* examined the stress variations and particle movements close to the pile using DEM analyses [Campos et al. 2005]. Esposito *et al.* emphasized the influence of the particle shape and the particle rotation on the pile installation process [Esposito et al. 2018].
The ability to reproduce the macroscopic behavior of the grains presents a major advantage of the DEM.

Chapter 3

Free surface

The waves in the sea, the running river or the cup of coffee that we drink, all constitute free surface flows where the free surface is the interface between the liquid and the air. More precisely, free-surface flows involve two or more immiscible fluids, separated by a sharp interface that evolves in time [Nadim 2008].
In this work, only liquid-gas interfaces are considered. The influence of the gas is neglected since it has a weak impact on the motion of the liquid. Therefore, the liquid moves independently, or freely, with respect to the gas.

Predicting the shape of the free surface is very important in many scientific and technical applications. However, the computation of such flows is difficult because the shape and the position of the interface between gas and liquid changes continuously [Demirdzic et al. 1999].

This chapter aims to present the equations that are used to determine the position and shape of the free surface. Afterward, the details of the numerical implementation are given. The free surface position is updated using an interface-tracking method. Thus, the numerical grid is adapted according to the shape of the free surface. Finally, an example is presented that shows the validity of the numerical approach used in this work.

3.1 Kinematic condition

The kinematic condition describes the motion of the free surface over time. It states that there is no mass flux across the free surface.

Steady state

To find the free surface equation in the steady case, a three-dimensional flow is considered. The free surface can be interpreted as a streamline. By definition, a streamline is a line, which is everywhere parallel to the local velocity vector [Drela 2018]. Hence, there is no flow across it. The velocity vector is defined by :

$$\mathbf{v} = (u, v, w)$$

A vector of infinitesimal length along the free surface can be written as :

$$\mathbf{ds} = (dx, dy, dz)$$

This vector needs to be parallel to the local velocity vector :

$$\mathbf{ds} \times \mathbf{v} = \mathbf{0}$$
$$(w\,dy - v\,dz, u\,dz - w\,dx, v\,dx - u\,dy) = \mathbf{0}$$

Setting each component of the previous equation to zero gives three differential equations that define the shape of the streamline.
In the two-dimensional case, $dz = 0$ and $w = 0$. Thus, the position of the free surface in steady-state can be computed using the following differential equation :

$$v\,dx - u\,dy = 0$$
$$\frac{dy}{dx} = \frac{v}{u}$$

The stationary kinematic condition claims that the free surface is tangent to the velocity vector, as shown in Figure 3.1.

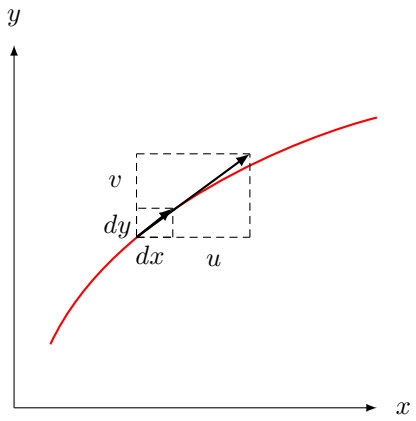


Figure 3.1: Kinematic condition in steady-state, with the free surface represented by the red curve

Transient state

In the transient state, the shape of the free surface is known at the initial time and changes continually. To find the displacement of the free surface, a two-dimensional flow is considered. The upper part of the domain is supposed to be a free boundary [Castro-Orgaz and Hager 2019]. The free surface is located in $y = h(x, t)$ where (x, y) are the Cartesian coordinates and t is the time, see Figure 3.2.

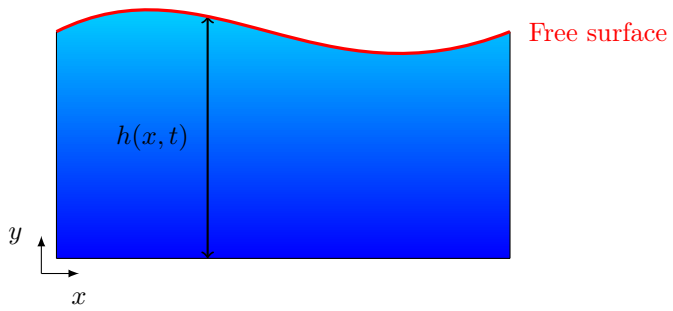


Figure 3.2: Two dimensional representation of a flow with the free surface represented by the red curve

The free surface is described by the following equation :

$$F(x,y,t) = y - h(x,t) = 0 \qquad \forall (x,y) \in \text{Free surface} \tag{3.1}$$

The mass is supposed to be conserved inside the domain. Therefore, there is no mass flux across the free surface, i.e. no particle can cross the free surface. This condition requires the free surface to move as a material surface [Tsai and Yue 1996]. The material derivative of F is given by :

$$\begin{aligned} \frac{\mathrm{D}F}{\mathrm{D}t} &= \frac{\partial F}{\partial t} + \mathbf{v} \cdot \nabla F = 0 \\ &= \frac{\partial F}{\partial t} + u\,\frac{\partial F}{\partial x} + v\,\frac{\partial F}{\partial y} \end{aligned} \tag{3.2}$$

The derivatives of F are calculated using equation (3.1) :

$$\frac{\partial F}{\partial t} = -\frac{\partial h}{\partial t} \qquad \frac{\partial F}{\partial x} = -\frac{\partial h}{\partial x} \qquad \frac{\partial F}{\partial y} = 1 \tag{3.3}$$

Inserting Equations (3.3) into Equation (3.2) becomes :

$$\frac{\mathrm{D}F}{\mathrm{D}t} = -\frac{\partial h}{\partial t} - u\,\frac{\partial h}{\partial x} + v = 0$$

Finally, the motion of the free surface is given by the following differential equation :

$$\boxed{\frac{\partial h}{\partial t} + u\,\frac{\partial h}{\partial x} - v = 0} \tag{3.4}$$

This equation is known as the *kinematic free surface condition*. The free surface condition in three dimensions with $\mathbf{v} = (u, v, w)$ can be found in a similar way as equation (3.4), resulting in :

$$\frac{\partial h}{\partial t} + u\,\frac{\partial h}{\partial x} + v\,\frac{\partial h}{\partial y} - w = 0$$

Problem statement

The kinematic condition (3.4) is used to find the new location of the free surface. Neumann conditions are applied on the boundary points of the free surface. The conditions consist in imposing a contact angle of 90° between the lateral walls and the free surface, as seen on Figure 3.3.

In order to find the height $h(x,t)$ of the free surface, the following equation needs to be solved :

$$\begin{aligned} &\frac{\partial h}{\partial t} + u\frac{\partial h}{\partial x} - v = 0 \qquad \text{on the free surface} \\ &\frac{\partial h}{\partial x} = 0 \qquad\qquad\qquad \text{at its boundary points} \end{aligned} \tag{3.5}$$

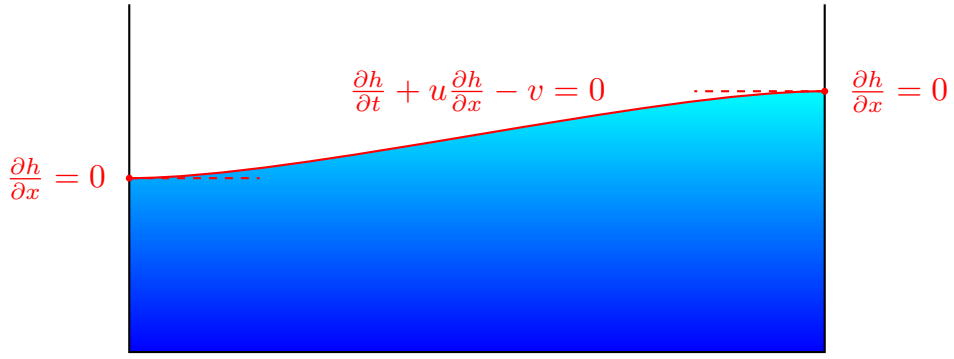


Figure 3.3: Schematic representation of the problem

Weak formulation

The weak formulation of the problem is obtained by multiplying the kinematic condition by test functions $\hat{h} \in L_2(\partial\Omega_{fs})$, where $\partial\Omega_{fs}$ denotes the free surface and L_2 is the space of square-integrable functions on $\partial\Omega_{fs}$. This expression is then integrated on the free surface. The following notation is used :

$$< f\, g >= \int_{\partial\Omega_{fs}} f\, g \, \mathrm{d}S$$

The weak formulation of the problem is given by :

Find $h \in L_2(\Gamma)$ such that :

$$< \frac{\partial h}{\partial t}\, \hat{h} > + < u\frac{\partial h}{\partial x}\, \hat{h} > - < v\, \hat{h} >= 0, \qquad \forall \hat{h} \in L_2(\Gamma)$$

To solve the problem numerically, the previous equations are discretized, resulting in :

$$\boxed{\begin{array}{l}\text{Find } h^h \in \mathcal{H}^h \text{ such that} \\ \\ \quad < \dfrac{\partial h^h}{\partial t}, \hat{h}^h > + < u\dfrac{\partial h^h}{\partial x}, \hat{h}^h > - < v, \hat{h}^h >= 0, \qquad \forall \hat{h}^h \in H^h\end{array}} \tag{3.6}$$

where $h^h(x,t)$ is the polynomial approximation of $h(x,t)$ and $\mathcal{H}^h$ is the discrete linear subspace of L_2.

Equation (3.6) is a linear system of n equations, where n is the number of free surface nodes. Solving this system allows to find the time derivative of $h(x,t)$. The position of the free surface $h(x,t)$ is obtained by solving this time derivative using an explicit Euler method.

3.2 Dynamic condition

The second condition that holds on the free surface is the dynamic condition. The condition states that the forces acting on fluids in contact at the free surface are in equilibrium, i.e. the momentum is conserved :

$$\sigma \cdot \mathbf{n} = (\gamma\,\kappa + p_{ext}) \cdot \mathbf{n} \tag{3.7}$$

where σ is the stress tensor, $\mathbf{n}$ is the vector normal to the domain in the outbound direction and p_{ext} represents the external pressure. The surface tension correspond to a pressure that is proportional to a coefficient γ and to the curvature of the free surface κ, see Figure 3.4.

The normal vector can be written as :

$$\mathbf{n} = \left(\left(\frac{\partial h}{\partial x}\right)^2 + 1\right)^{-1/2} \left(-\frac{\partial h}{\partial x}, 1\right) \tag{3.8}$$

The surface curvature is defined as :

$$\kappa = \mathrm{tr}(\nabla\,\mathbf{n})$$

where ∇ is the gradient along the free surface.

By using equation (3.8), the curvature κ becomes :

$$\begin{aligned}\kappa &= \frac{\partial}{\partial x}\left[\left(\left(\frac{\partial h}{\partial x}\right)^2 + 1\right)^{-1/2} \frac{\partial h}{\partial x}\right] \\ &= \left[\frac{\partial^2 h}{\partial x^2}\left(\left(\frac{\partial h}{\partial x}\right)^2 + 1\right)^{-1/2} - \quad \frac{\partial h}{\partial x}\left(\left(\frac{\partial h}{\partial x}\right)^2 + 1\right)^{-3/2} \frac{\partial h}{\partial x}\frac{\partial^2 h}{\partial x^2}\right] \\ &= \frac{\partial^2 h}{\partial x^2}\left(\left(\frac{\partial h}{\partial x}\right)^2 + 1\right)^{-3/2}\end{aligned} \tag{3.9}$$

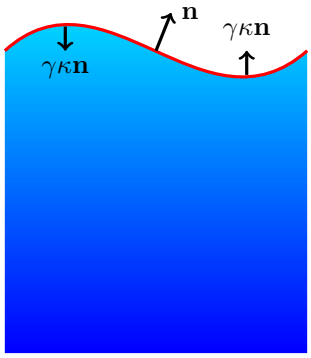


Figure 3.4: Direction of the surface tension along the free surface

By neglecting the viscous effects at the free surface, the stress tensor can be written as :

$$\sigma = -p\mathbf{I}$$

where $\mathbf{I}$ is the unit vector. Finally, the dynamic condition (3.7) becomes :

$$\boxed{p - p_{ext} = -\gamma \frac{\partial^2 h}{\partial x^2} \left(\left(\frac{\partial h}{\partial x} \right)^2 + 1 \right)^{-3/2}} \tag{3.10}$$

The surface tension coefficient is equal to $7 \cdot 10^{-2}$ Nm^{-1} for water-air interfaces at room temperature [Adamson and Gast 1997].

In many problems including free surfaces, the surface tension is neglected due to the fact that the dimensions of the solution domain are chosen to be sufficiently large. For such cases, the dynamic condition for an incompressible Newtonian fluid reduces to the statement that the pressures on either side of the interface are equal [Muzaferija and Peric 1997].

$\mathbf{L_2}$ projection

In order to solve the dynamic condition, the curvature (3.9) of the surface needs to be determined. This requires the computation of the second order derivative of $h(x,t)$. To calculate this term, a projection into the space L_2 is performed. This consists in multiplying the second derivative by a basis function τ_i and integrating this expression on the free surface. The integral is calculated by parts. The notation $\ll \cdot \gg$ is used to write the integral on the boundary Γ of the free surface :

$$\ll f\, g \gg = \int_\Gamma f\, g\, dl$$

The projection is given by :

$$\begin{aligned} < \tau_i \frac{\partial^2 h}{\partial x^2} > &= - < \frac{\partial \tau_i}{\partial x} \frac{\partial h}{\partial x} > + \ll \mathbf{n} \cdot \tau_i \frac{\partial h}{\partial x} \gg \\ &= - < \frac{\partial \tau_i}{\partial x} \frac{\partial h}{\partial x} > + 0 \end{aligned} \tag{3.11}$$

The integral on the boundary is equal to zero because the Neumann condition on these boundaries states that $\frac{\partial h}{\partial x} = 0$, see Figure 3.3.

Equation (3.11) is discretized using :

$$h(x,t) \approx \sum_{j=0}^{n-1} \tau_j(x) H_j(t)$$

Finally, the second order derivative of $h(x,t)$ is found by solving the following system of n equations, where n is the number of free surface nodes :

$$\boxed{\sum_{j=0}^{n-1} < \tau_i \tau_j > \left(\frac{\partial^2 h}{\partial x^2}\right)_j = -\sum_{j=0}^{n-1} < \frac{d\tau_i}{dx}\frac{d\tau_j}{dx} > H_j} \tag{3.12}$$

3.3 Volume conservation

When solving the kinematic condition, the velocity field must satisfy the incompressibility condition in order to ensure the volume conservation. In the case of poor satisfaction of the incompressibility condition, the volume varies with each iteration and the numerical solution can diverge. Therefore, it is necessary to correct the velocity field. This problem is also illustrated and resolved by Rose *et al.* [Rose, Buffett, and Heister 2017]. This correction is only applied for problems that conserve the volume, i.e. when the free surface is the only open boundary of the domain.

Rose *et al.* formulates the kinematic condition using the free surface normal vector $\mathbf{n} = \left(-\frac{\partial h}{\partial x}, 1\right)$:

$$\frac{\partial h}{\partial t} = \mathbf{v} \cdot \mathbf{n} \tag{3.13}$$

The domain is denoted by Ω and its boundary by $\partial\Omega$. By integrating equation (3.13) on the the free surface $\partial\Omega_{fs} \subset \partial\Omega$ and by using the divergence theorem, the equation becomes :

$$\begin{aligned} \int_{\partial\Omega_{fs}} \frac{\partial h}{\partial t}\frac{1}{||\mathbf{n}||} dl &= \int_{\partial\Omega_{fs}} \mathbf{v} \cdot \hat{\mathbf{n}}\, dl \\ \int_0^L \frac{\partial h}{\partial t}\, dx &= \int_{\partial\Omega} \mathbf{v} \cdot \hat{\mathbf{n}}\, dl - \int_{\partial\Omega \setminus \partial\Omega_{fs}} \mathbf{v} \cdot \hat{\mathbf{n}}\, dl \\ &= \int_\Omega \nabla \cdot \mathbf{v}\, dS - \int_{\partial\Omega \setminus \partial\Omega_{fs}} \mathbf{v} \cdot \hat{\mathbf{n}}\, dl \end{aligned} \tag{3.14}$$

The vector $\hat{\mathbf{n}}$ is the unit normal vector. Since the fluid is incompressible, the expression becomes :

$$\int_0^L \frac{\partial h}{\partial t}\, dx = -\int_{\partial\Omega \setminus \partial\Omega_{fs}} \mathbf{v} \cdot \hat{\mathbf{n}}\, dl \tag{3.15}$$

Along the walls of the domain, the fluid is supposed to stick. Therefore, it is required that the speed of the fluid is equal to the speed of the solid wall. If the walls are stationary, Equation (3.15) reduces to :

$$\int_0^L \frac{\partial h}{\partial t}\, dx = 0$$

The previous equation states that the overall variation of the height of the free surface is zero. Using Equation (3.14), the condition that needs to be satisfied in order to conserve the volume is given by:

$$\int_{\partial\Omega_{fs}} \mathbf{v} \cdot \hat{\mathbf{n}}\, dl = 0 \tag{3.16}$$

The correction made on the velocity field can be written as :

$$\mathbf{v} = \mathbf{v}^* + a\mathbf{e_y}$$

where $\mathbf{v} = (u, v)$ and $\mathbf{v}^* = (u^*, v^*)$, are the fluid velocities after and before the correction, respectively and $\mathbf{e_y}$ is the vertical unit vector. To find the expression of the correction parameter a, the corrected velocity $\mathbf{v}$ is inserted in Equation (3.16) :

$$\int_{\partial\Omega_{fs}} \mathbf{v}^* \cdot \hat{\mathbf{n}}\ \mathrm{dl} + a \underbrace{\int_{\partial\Omega_f} \frac{1}{\|\mathbf{n}\|}\ \mathrm{dl}}_{=L} = 0$$

$$\Rightarrow a = -\frac{1}{L} \int_{\partial\Omega_f} \mathbf{v}^* \cdot \hat{\mathbf{n}}\ \mathrm{dl}$$

To ensure the volume conservation, the velocities must be corrected as :

$$\boxed{v = v^* - \frac{1}{L} \int_{\partial\Omega_{fs}} \mathbf{v}^* \cdot \hat{\mathbf{n}}\ \mathrm{dl}.} \tag{3.17}$$

3.4 Stick-slip flow

One well-known application of free surface flows is the stick-slip flow, which is also known as the die-swell phenomenon.

Description

The geometry, the boundary conditions and the parameters of the problem are given in Table (3.1). A Poiseuille flow is imposed at the inlet of the tube $\partial\Omega_{\text{In}}$. The stick-slip flow is characterized by abrupt changes of boundary conditions. The flow changes from zero velocity (stick velocity) at the tube wall $\partial\Omega_{\text{Die}}$ to a finite velocity (slip velocity) on the free surface $\partial\Omega_{\text{fs}}$. The surface tension is neglected. Therefore, the condition $p = p_{\text{ext}}$ is applied on the free surface.

The fluid emerging from the tube expands (or contracts) to a certain swelling ratio. This swelling ratio is defined as :

$$\chi = H_f/H$$

where H is the radius of the cylindrical tube out of which the liquid is extruded and H_f is the final radius of the jet when the velocity field has become uniform.

The swelling of the jet depends on the rheological properties of the fluid and the boundary conditions [André and Clermont 1987]. Experimental and numerical swelling values for Newtonian fluids can be found in the works of Goren and Wronski [Goren and Wronski 1966] and Omodei [Omodei 1980]. Goren and Wronski analyzed experimentally the dependence between the Reynolds number and the swelling rate. They showed that for Reynolds numbers less than about 16, the jet expands and for Reynolds numbers greater than 16, it contracts. Omodei solved the problem numerically and he analyzed the influence of the surface tension on the swelling ratio.

Finite difference method

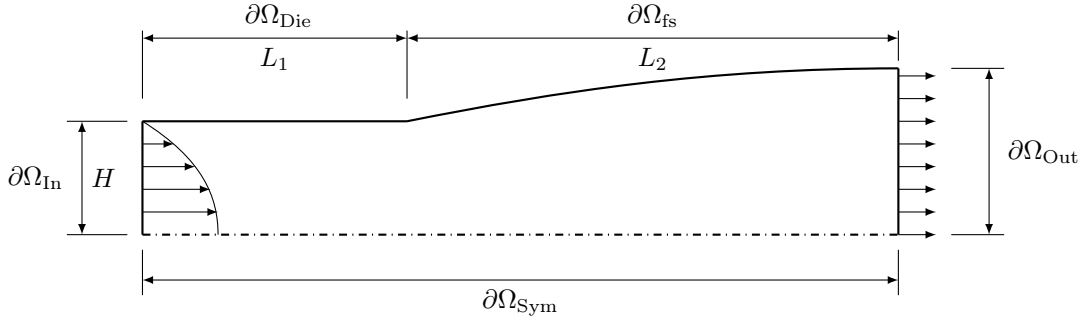
In order to solve the kinematic condition (3.4) on the free surface, a finite difference method is used. The time derivative is discretized using an explicit Euler method. The spatial derivative is solved by applying an upwind scheme. The reason why such a scheme is chosen is that the flow propagates from left to right. Therefore, a method that propagates the information in this direction is necessary. To write the discretization of the equations, the following notation is used :

$$h(x_i, t_n) = h_i^n$$

where $x_i = x_0 + i\Delta x$ and $t_n = t_0 + n\Delta t$ with Δx the spatial step and Δt the time step. By discretizing equation (3.5), the following finite difference scheme is obtained :

$$\begin{cases} \dfrac{h_i^{n+1} - h_i^n}{\Delta t} + u_{i-1}^n \dfrac{h_i^n - h_{i-1}^n}{\Delta x} - v_{i-1}^n = 0 & \forall i \in \{1, ..., n-2\} \\ \dfrac{h_i^{n+1} - h_i^n}{\Delta t} - v_{i-1}^n = 0 & \forall i \in \{0, n-1\} \end{cases} \tag{3.18}$$

Geometry



Boundary conditions

$\partial\Omega_{\text{In}}$:	$u = 2(1-y^2)$, $v = 0$	$\partial\Omega_{\text{Out}}$:	$v = 0$, $p = 0$
$\partial\Omega_{\text{Die}}$:	$u = 0$, $v = 0$	$\partial\Omega_{\text{Sym}}$:	$v = 0$
$\partial\Omega_{\text{fs}}$:	$p = p_{\text{ext}}$		

Physical parameters	**Geometric parameters**	**Numerical parameters**
$\rho = 12 - 18$ kg·m^{-3}	$H = 1$ m	$\Delta t = 0.005$ s
$\mu = 1$ Pa·s	$L_1 = 3.5$ m	$t_{End} = 100$ s
$Re = 12 - 36$	$L_2 = 20$ m	# elements = 10684

Table 3.1: Geometric and physical characteristics of the die-swell problem

The order of accuracy of this scheme is $\mathcal{O}(\Delta t, \Delta x)$. The Courant–Friedrichs–Lewy (CFL) condition must be respected to assure convergence :

$$\frac{u_{max}\,\Delta t}{\Delta x} \leq 1 \tag{3.19}$$

Results

The numerical results are in agreement with the results found in the literature, as it can be seen in Table (3.2). The final free surface profiles are shown on Figure 3.5. On the same figure, the free surface profile obtained by Omodei for $Re = 24$ is represented by the green curve.

Re	χ	χ_N	χ_E
24	0.955	0.954	0.955
29	0.941	0.940	0.941
36	0.924	0.926	0.928

Table 3.2: Swelling ratio χ compared with the numerical values χ_N found by Omodei and the experimental values χ_E found by Goren and Wronski

The snapshots of the simulation at the first iteration and at the last iteration on Figure 3.6 show the deformation of the free surface, as well as the mesh that has been used.
The die-swell problem converges to stationary free surface. Therefore, on the free surface, the velocity vector is everywhere tangent to the surface, as seen on Figure 3.7.

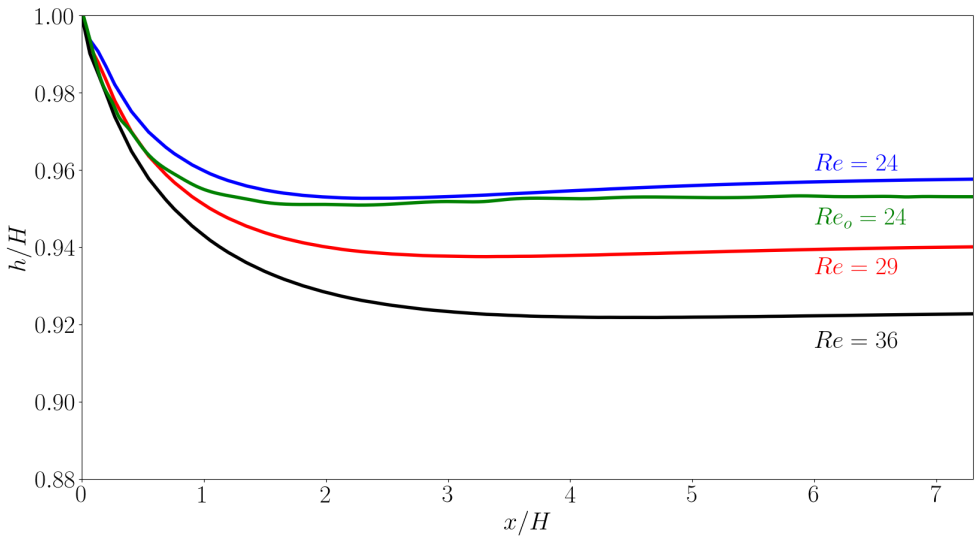


Figure 3.5: Free surface profiles for $Re = \{24, 29, 36\}$ and results obtained by Omodei for $Re = 24$ in green

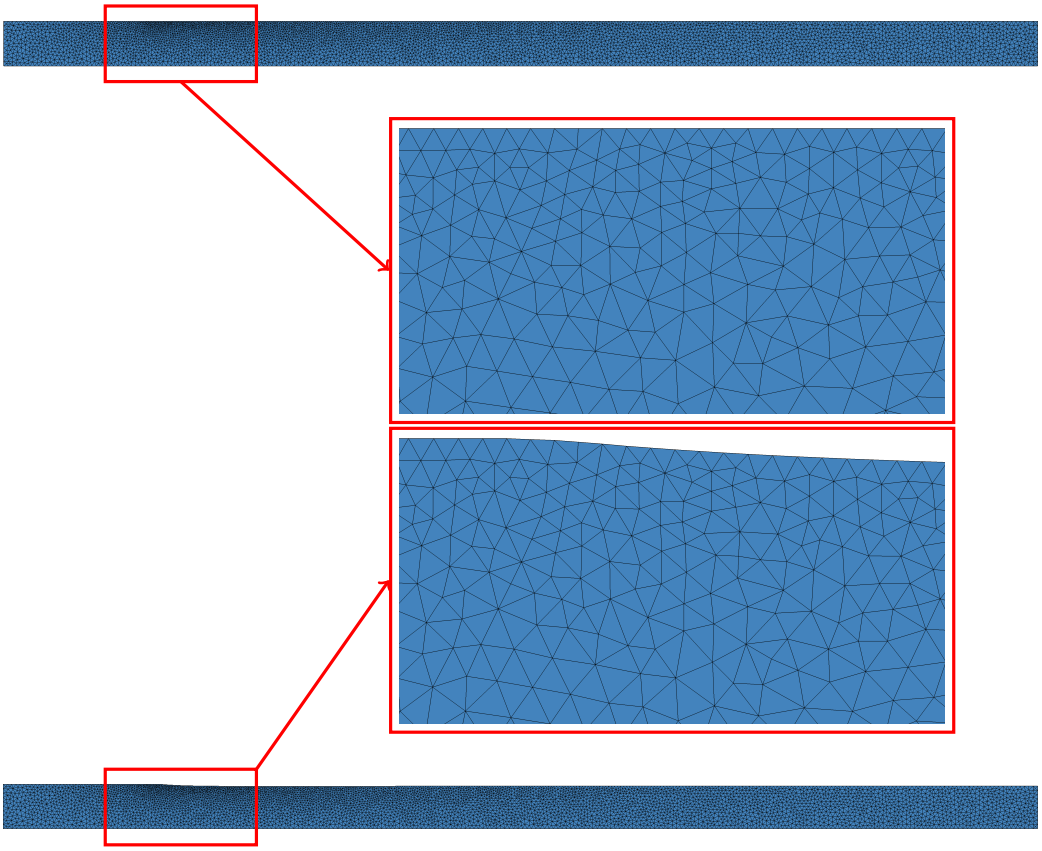


Figure 3.6: Snapshots of the simulation at $t = 0$ et $t = 100$s for $Re = 24$

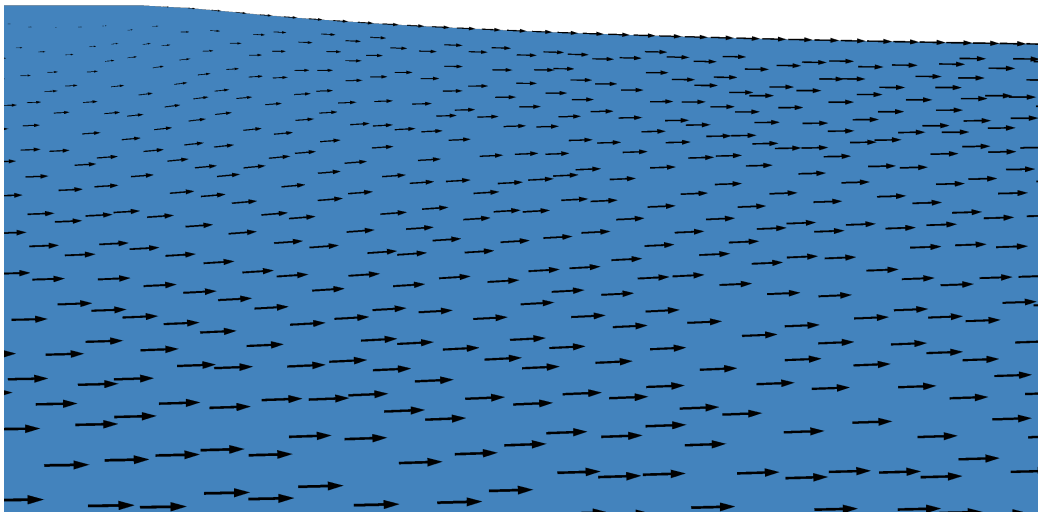


Figure 3.7: Velocity vectors tangent to the free surface for $Re = 29$

Chapter 4

Mesh regularization

As seen in the previous chapter, the free surface is treated as a material line. Thus, its displacement is simulated by moving the nodes located on it. This leads to a deformation of the domain. In order to keep a valid mesh, the nodes inside the domain need to be adapted by taking into account the motion of the free surface. This way, the good quality of the mesh can be preserved during the computation. In fact, very thin or skewed elements may lead to poor approximation results and instabilities of the solution [Shewchuk 2002]. Therefore, developing a procedure to improve the quality of mesh elements is of great importance.
The method used to regularize the mesh is in most of the cases problem dependent. There exists various mesh modification algorithms that can be used to guarantee that the mesh does not get tangled or too distorted during the computation. Park and Shontz classify these techniques in three groups: swapping, adaptivity and smoothing techniques [Park and Shontz 2011].

Mesh swapping techniques change the connectivity of the mesh by swapping the faces or edges to form new elements. For each possible swapped configuration, the quality of the mesh is measured. The mesh that provides the best quality improvement is kept and is used as the new mesh [Feuillet et al. 2018].

Adaptive mesh methods consist in adapting the mesh in order to refine the elements of the mesh in certain areas. The nodes are moved towards sensitive or turbulent regions requiring more resolution [Askes and Rodriguez-Ferran 2001]. This method preserves the connectivity of the mesh. In order to determine the areas requiring refinement, it is necessary to estimate the error committed by the numerical method. The difficulty of this method is to find an appropriate estimator for any type of problem [Boxho 2019].

Smoothing methods aim to improve the quality of the mesh by keeping it as regular as possible. This is done by moving individual nodes while preserving the connectivity of the mesh. Vartziotis and Bohnet describe the two main approaches of mesh smoothing: optimization-based and geometric-based approaches [Vartziotis and Bohnet 2017].
Optimization-based methods try to maximize an objective function which correspond to a mesh quality measure. The choice of this measure significantly affects the performance of the optimization method. Even though this method is very effective in improving the mesh quality, it requires additional computational effort [Durand, Rosero, and Oliveira 2019].

The geometric-based approaches move the nodes according to a geometric rule in order to improve the overall mesh quality. Due to its simplicity and its low numerical complexity, elliptic grid generation methods are the most popular geometric-based methods. This method has been chosen to regularize the mesh in this work.

4.1 Elliptic grid generation

For free surface flows, elliptic grid generation methods (also called *Laplacian smoothing method*) are often used to adapt the position of the interior nodes of the mesh. Each time the position of the free surface is updated, the interior nodes of the numerical mesh need to be adapted in order to respond to the movement of the vertices of the free surface.
Elliptic grid generation methods consist in solving a Laplace equation for each component of the nodal positions. This method was initially proposed by Winslow [Winslow 1963].

The coordinates of the new mesh nodes $\mathbf{x} = (x, y)$ depend on the coordinates of the initial mesh $\boldsymbol{\xi} = (\xi, \eta)$ and the coordinates of the new boundary $\mathbf{x}^* = (x^*, y^*)$. of the domain.
In order to find the position of the new mesh nodes $\mathbf{x}$, the method proposes to solve the following system [Legat 1992] :

$$\begin{cases} \Delta_{\boldsymbol{\xi}} \mathbf{x} = 0 & \text{on } \Omega \\ \mathbf{x} = \mathbf{x}^* & \text{on } \partial\Omega \end{cases} \tag{4.1}$$

where Ω denotes the domain and $\partial\Omega$ its boundary.

The coordinates of the initial mesh do *a priori* not satisfy this system of equations. Therefore, the application of the elliptic method modifies the internal configuration of the mesh, even if the boundaries have not changed. This modification is not desirable if the initial mesh is supposed to have an optimal configuration.
This undesired deformation of the interior nodes can be avoided by writing the system (4.1) in terms of the displacement of the nodes $\delta\mathbf{x}$:

$$\delta\mathbf{x} = \mathbf{x} - \boldsymbol{\xi}$$

System (4.1) becomes :

$$\begin{cases} \Delta_{\boldsymbol{\xi}}\, \delta\mathbf{x} = 0 & \text{on } \Omega \\ \mathbf{x} = \mathbf{x}^* & \text{on } \partial\Omega \end{cases} \tag{4.2}$$

Equation (4.2) can be solved numerically using finite element or finite difference techniques. The complexity of the method is $\mathcal{O}(n^3)$ where n is the number of nodes of the computational mesh.

4.2 Simplified elliptic grid generation

Using the fact that the free surface only moves vertically, makes it possible to simplify the method (4.2). The simplified smoothing method consists in readjusting vertically the position of the interior nodes to preserve the same relative position between the free surface and the opposite boundary [Muzaferija 1999]. Equation (4.2) can thus be simplified by imposing the following constraint on the interior nodes of the mesh :

$$\frac{\partial^2(\delta y)}{\partial \eta^2} = 0 \qquad \forall y \in \Omega \tag{4.3}$$

The equation can be simplified as :

$$\frac{\partial^2 y}{\partial \eta^2} = 0 \qquad \forall y \in \Omega \tag{4.4}$$

The solution of the differential equation (4.4) is given by :

$$y(\xi, \eta, t) = A(\xi, t)\,\eta + B(\xi, t) \qquad \forall\ (\xi, \eta)\ \in \Omega \tag{4.5}$$

Two boundary conditions are required to find the parameters $A(\xi, t)$ and $B(\xi, t)$. The Dirichlet boundary conditions on the free surface and on the bottom of the domain are given by :

$$\begin{aligned} y(\xi, \eta_{fs}, t) &= h(\xi, t) \qquad\qquad \forall\ (\xi, \eta_{fs})\ \in \partial\Omega_{fs} \\ y(\xi, 0, t) &= 0 \end{aligned}$$

where $h(\xi, t)$ is the height of the free surface node located in η at time t. Using these conditions, Equation (4.5) becomes :

$$\boxed{y(\xi, \eta, t) = \frac{h(\xi, t)}{\eta_{fs}}\,\eta} \tag{4.6}$$

Equation (4.6) states that, for each node, the ratio $\frac{\eta}{\eta^*}$ remains constant during the whole calculation. The main advantage of this method is its simplicity and its efficiency. In fact, the smoothing method can be done in $\mathcal{O}(n)$ operations, where n is the number of nodes (if the position of the free boundary can be found in $\mathcal{O}(1)$) .

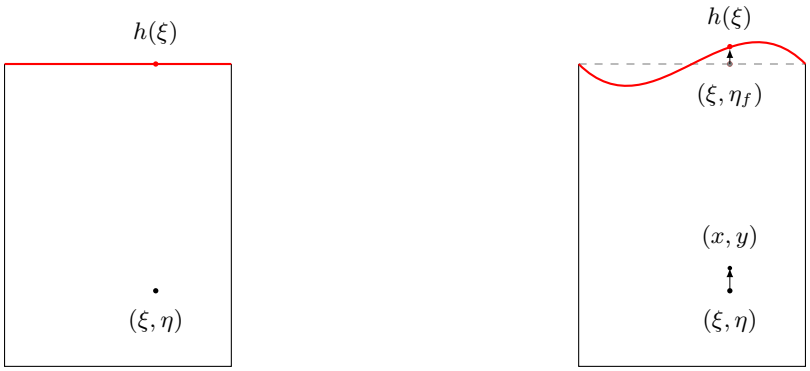


Figure 4.1: Illustration of regularization. The free surface, represented in red, is deformed using the kinematic condition. The inner nodes are moved vertically to satisfy Equation (4.6). Their new position is denoted by (x, y).

Application : Wave propagation

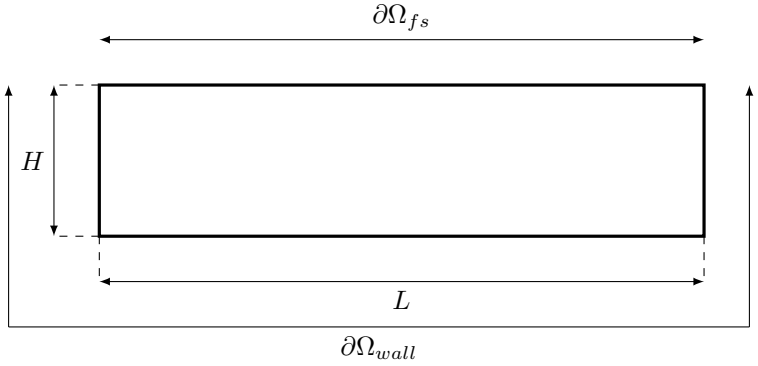
The simulation for the propagation of a wave is shown on Figure 4.2. The domain containing the fluid is subjected to a gravity acceleration acting downwards. The geometry as well as the

parameters used to simulate the problem are described in Table (4.1). An initial deformation is applied to the free surface, given by :

$$h(x, t = 0) = a\frac{\sin \tilde{x}}{\tilde{x}}, \qquad \tilde{x} = \frac{x - b}{c}$$

where $a = H/4$, $b = L/2$ and $c = L/30$. Due to the gravity, the waves propagate between the two lateral walls. At each time step, the free surface is deformed by moving the nodes located on it. This deformation is computed using the free surface condition seen in the previous chapter. Then, the interior nodes are displaced using relation (4.6). It can be seen on Figure 4.2 that the mesh nodes are adapted according to the free surface position and that the mesh remains regular during the simulation.

Geometry



Boundary conditions

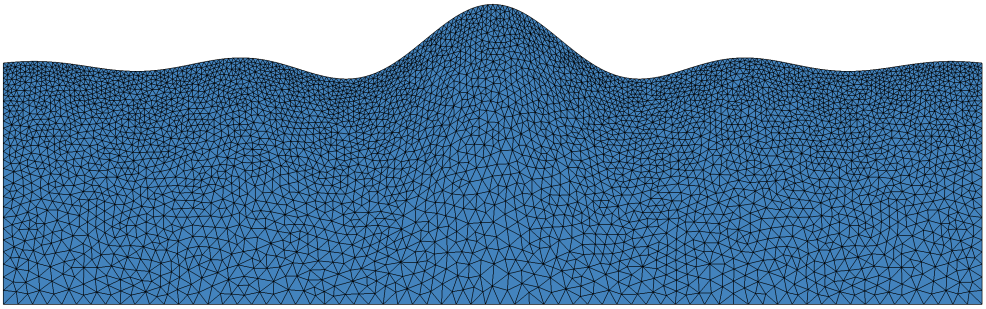
$\partial\Omega_{wall}$: $u = 0$, $v = 0$
$\partial\Omega_{fs}$: $p = p_{ext}$

Physical parameters	**Geometric parameters**	**Numerical parameters**
$\rho = 10^3$ kg m^{-3}	$L = 1$ m	$\Delta t = 10^{-3}$ s
$\mu = 10^{-3}$ Pa s	$H = 0.25$ m	$t_{End} = 1$ s
$\gamma = 0.07$ N m^{-1}		# elements = 9216
$g = 9.81$ m s^{-2}		

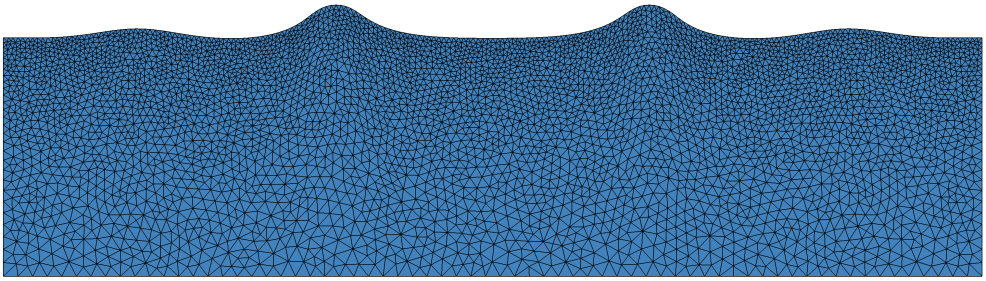
Table 4.1: Geometric and physical characteristics of the problem

Limitations of the method

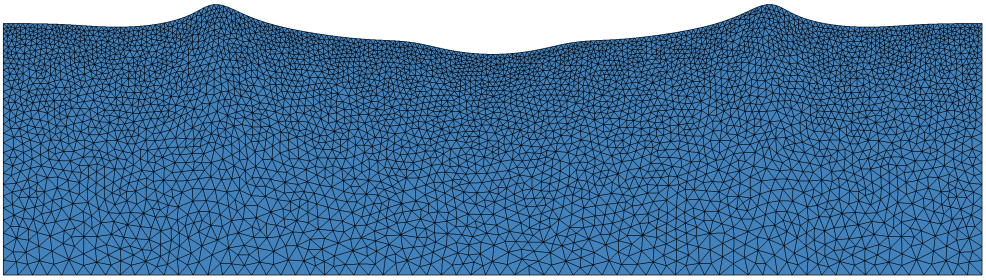
Although the simplified method is very effective in many situations, it can lead to an invalid mesh if the free surface is subject to large amplitude oscillations that are not sufficiently smooth. In such situations, the mesh elements become highly deformed and tangled, which can degrade the numerical solution. An example of an invalid mesh can be seen on Figure 4.3, where the free surface



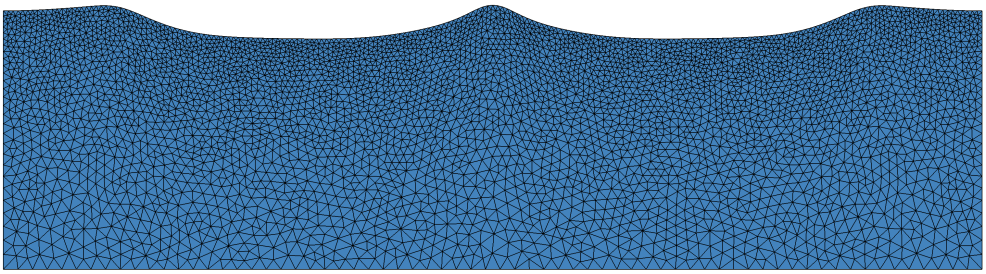
(a) t=0.00s



(b) t=0.25s



(c) t=0.50s



(d) t=0.75s

Figure 4.2: Free surface and mesh regularization at various time levels

is subject to a sharp oscillation. The elements below the free surface are strongly deformed and the red element and the green element become entangled.
In order to avoid this undesirable effect, more robust regularisation methods can be used, such as the Laplacian smoothing method of Equation (4.2).
In this work, these adverse effects have been avoided by choosing the problem parameters such that the oscillations remain sufficiently smooth.

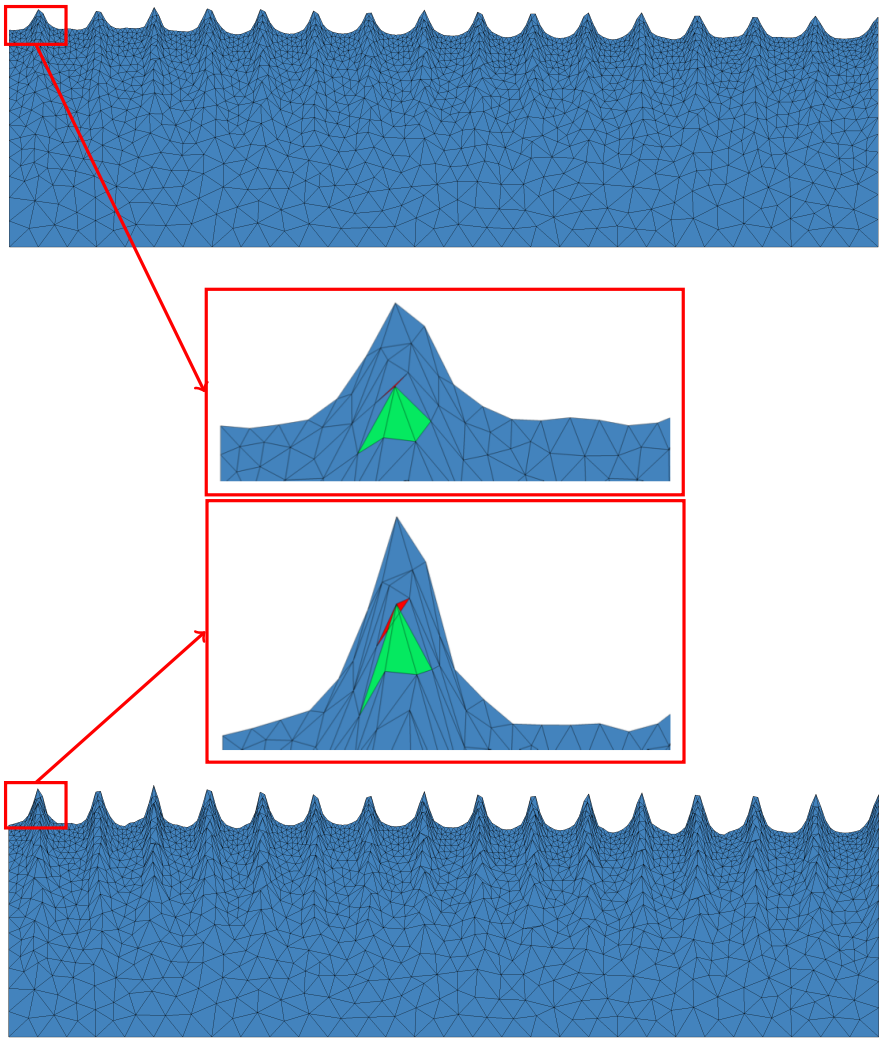
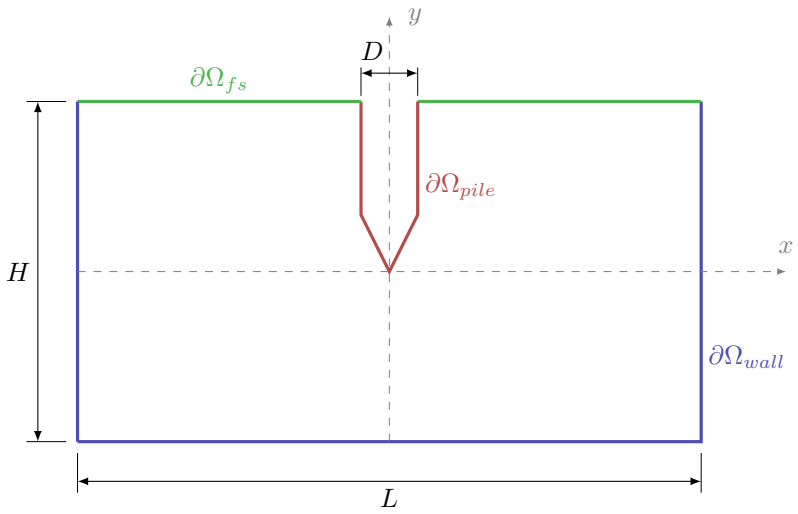


Figure 4.3: Mesh before and after the elements get tangled

4.3 Mesh regularization for the pile driving simulation

The final goal of this work is to simulate pile driving into a granular saturated soil. The pile is considered as a rigid Lagrangian body with a conical tip. The geometry of the domain and the problem parameters can be seen in Table 4.2. At this state, the domain is only composed of the fluid with the goal to show the mesh deformation. The complete simulation of pile driving into the solid particles is discussed in the following chapter.
The pile penetrates vertically into the domain. For this application, the mesh moves not only with the free surface, but also with the motion of the pile. Therefore, a problem dependent method is necessary to regularize the mesh.

Geometry



Boundary conditions

$\partial\Omega_{wall}$: $u = 0,\ v = 0$
$\partial\Omega_{pile}$: $u = 0,\ v = -0.2\sin(0.2\pi t)$
$\partial\Omega_{fs}$: $p = p_{ext}$

Physical parameters	**Geometric parameters**	**Numerical parameters**
$\rho = 10^3$ kg·m^{-3}	$L = 5$ m	$\Delta t = 10^{-3}$ s
$\mu = 10^{-3}$ Pa·s	$H = 2.4$ m	$t_{End} = 20$ s
$\gamma = 0.07$ N·m^{-1}	$D = 0.25$ m	# elements = 2349
$g = 9.81$ m·s^{-2}		

Table 4.2: Geometric and physical characteristics of the problem

The mesh nodes distant from the pile are adapted by taking into account only the motion of the free surface, using Equation 4.6. Whereas the mesh nodes near the pile need to take into account the motion of the pile and the motion of the free surface. The detailed description and implementation

can be found in Appendix A. When the pile penetrates deep into the domain, large deformations of the mesh elements near the pile can be observed. The deformation of the mesh can be seen on Figure 4.4.

Quality analysis of the mesh

The shape and the distribution of the mesh elements have an important impact on the numerical stability as well as on the solution accuracy. The quality of a mesh is determined more effectively by analyzing quality metrics rather than by visual inspection. Mesh quality is a very broad term and there exists several metrics to represent the quality of elements of a mesh. According to Knupp, a quality metric is a scalar function that measures some geometric property of the element [Knupp 2001].

A straightforward quality metric is the minimum angle of a triangular mesh element. Elements with small angles are considered to be of worse quality than ones with larger angles. The best shaped triangles are supposed to be equilateral triangles with a minimum angle of 60°. Triangles with their minimum angles smaller than 30° are considered to be of poor quality and should be avoided.

Another quality metric that is often used is the shape quality measured as [Bhatia and Lawrence 1990] :

$$q = \frac{4\sqrt{3}A}{l_1^2 + l_2^2 + l_3^2}$$

where A relates to the area of a triangle and l_1, l_2 and l_3 to its edge lengths. The normalization factor $4\sqrt{3}$ ensures that a perfect triangle (equal edges) has a quality of 1. A quality value of 0 indicates a completely degenerate cell.

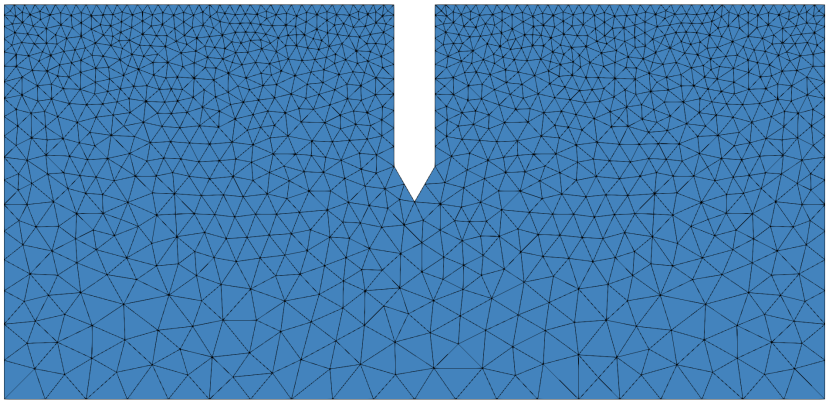
Figures 4.5a and 4.5b show the distribution of the shape quality value q of the elements and the distribution of the smallest angle, at the moment the pile reaches its maximum depth, i.e. the mesh on Figure 4.4. It can be seen that approximately 5% of the elements have shape quality values below 0.5 and that 25% have minimum angles smaller than 30°. This is an indication for poor mesh quality and can lead to convergence difficulties and numerical instabilities.

Furthermore, in order for the multi-scale model used by the MigFlow software to be valid, it is crucial that the size of the particles is smaller than the size of the elements of the FEM. Except for very small particles, this criteria can not be satisfied for the highly compressed elements below the pile tip.

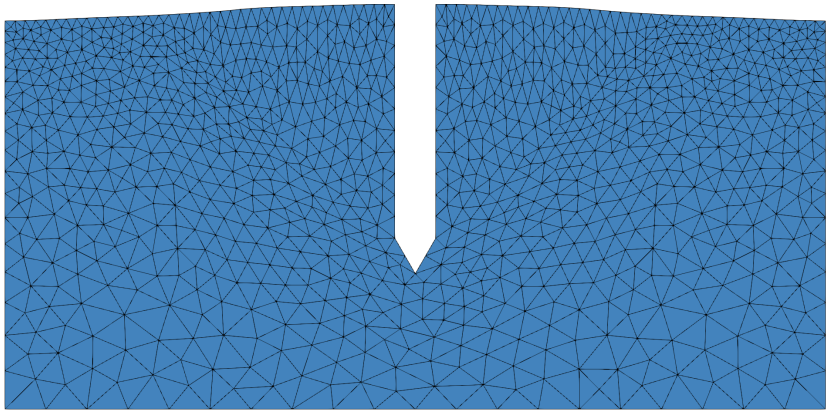
Re-meshing

A strategy is needed to cope with the large deformations of the mesh and to provide a suitable mesh quality throughout the calculation process. One common approach is to generate a new mesh (re-mesh), before it gets too distorted. The re-meshing process is done using the software GMSH[1]. A complete new mesh is loaded and the solution is projected from the old distorted mesh to the new regular mesh. The new mesh is generated independently from the old mesh, so the number of vertices in the domain changes during the numerical simulation.

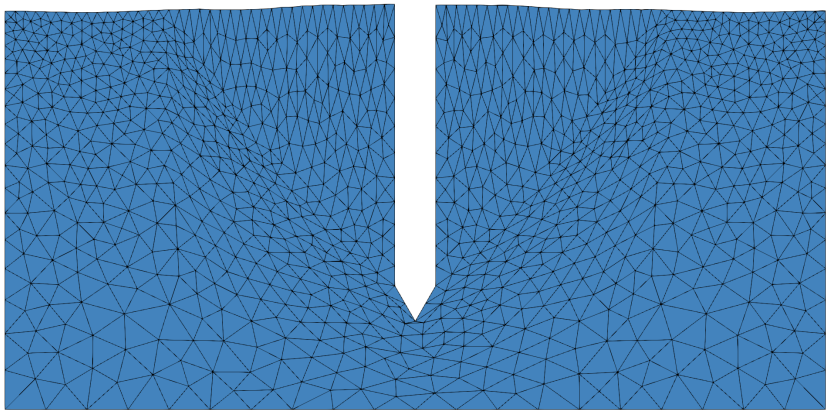
[1] https://gmsh.info/



(a) Initial mesh configuration and initial height of the pile



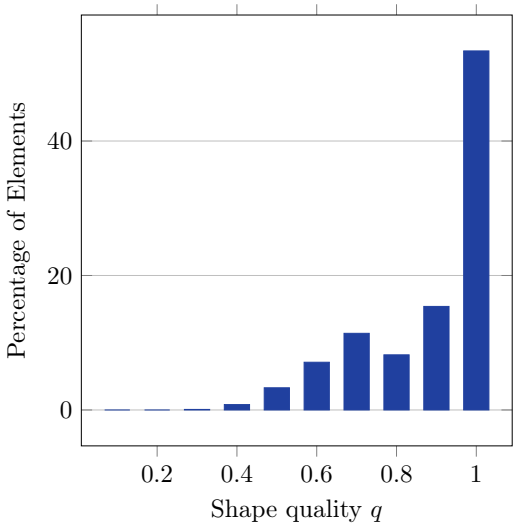
(b) Deformed mesh as the pile moves downwards



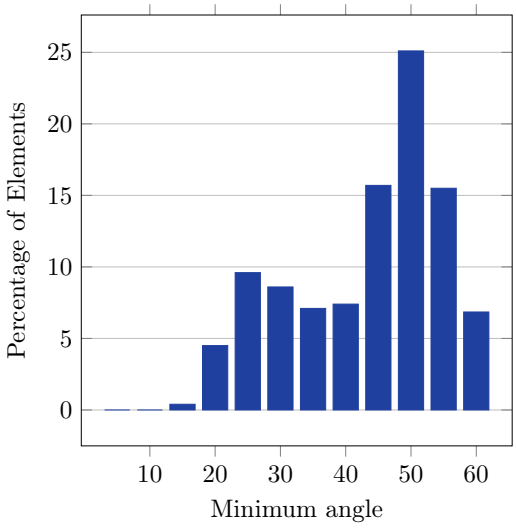
(c) Deformed mesh as the pile reaches its maximum depth

Figure 4.4: Mesh deformation for different pile depths

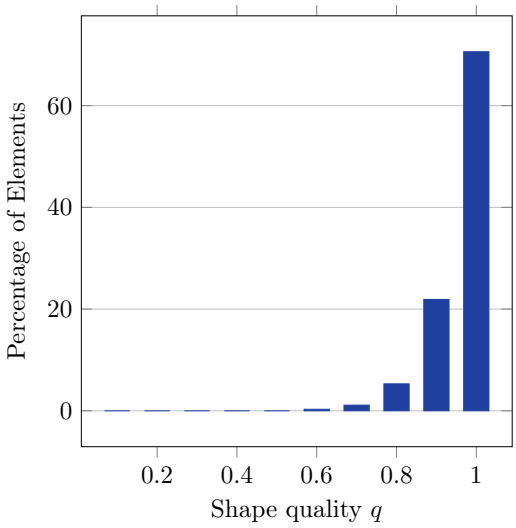
For the pile driving problem, a new mesh is generated if 5% of the elements have minimum angles smaller than 30°. This criteria has been chosen by the method of trial and error. The distribution of the shape quality and the minimum angle are shown on figure 4.5c and figure 4.5d respectively. The distribution is more satisfactory from the shape quality viewpoint. All of the triangles lie now in the gap $q \in [0.7, 1]$, which is very acceptable. The number of elements with shape quality $q \in [0.9, 1]$ has increased significantly. As for the minimum angles, the distribution is now concentrated around 50° and nearly all elements have minimum angles of more than 30°.
Hence, the quality of the mesh increases, while the risk of numerical errors due to mesh distortion decreases.



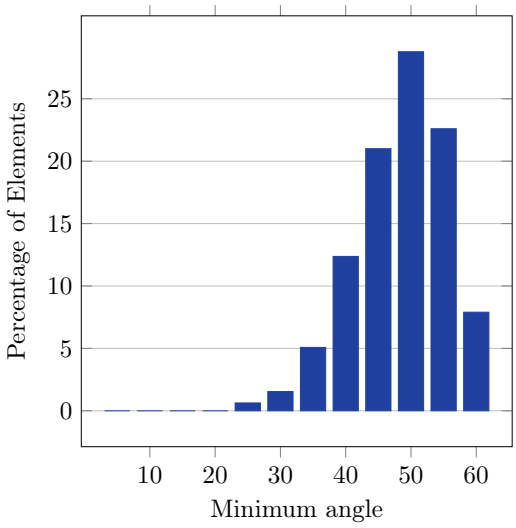
(a) Shape quality distribution without remeshing



(b) Minimum angle distribution without remeshing



(c) Shape quality distribution with remeshing



(d) Minimum angle distribution with remeshing

Figure 4.5: Quality metrics of the mesh when the pile reaches its maximum depth position

Chapter 5

Arbitrary Lagrangian-Eulerian Method

In continuum mechanics, there are two classical ways of looking at the description of the motion in space, which are referred to as Lagrangian and Eulerian formulations.

In the *Lagrangian description*, the observer follows the moving body. More precisely, the nodes of the mesh follow the material particles as they move through space. The Lagrangian description is often used in solid mechanics [Donea et al. 2004]. The major advantage of this method is that element boundaries coincide with the material interfaces. This makes it easy to impose boundary conditions and to track material interfaces. The drawback of this method is that in case of large distortions of the mesh, one needs to frequently remesh the domain to avoid numerical errors.

The *Eulerian method* describes the flow field in a fixed region. The continuum moves with respect to the fixed mesh. This method finds many applications in fluid mechanics. However, if the domain is variable, the Eulerian method is not optimal because the tracking of moving interfaces needs a significant amount of numerical calculus.

Numerical simulations of pile driving problems are characterized by large deformations of the continuum, accompanied by an evolution of material interfaces and free surfaces [Aubram 2015]. Purely Lagrangian or purely Eulerian descriptions of the flow are not suitable to describe these large distortions of the continuum. The Arbitrary Lagrangian-Eulerian method (ALE method) is an effective way in numerical fluid mechanics to couple an Eulerian formulation with a Lagrangian formulation and to combine their benefits.

The ALE method allows the nodes of the computational mesh to move in some arbitrary way. This method fixes the attention neither on material points, nor on a fixed region of space, but on what are called reference points. These points move with the continuum, but independently from the material points. The motion of the reference points is taken into account in the conservation equations. Using this method, large distortions of the continuum can be handled properly.

5.1 Fundamental ALE equation

In order to express the conservation equations using the ALE description, a relation between the material derivative and the referential derivative is needed. This relation is called the *fundamental ALE equation*.

To find this relation, three different configurations are introduced (Figure 5.1) :

- the material domain Ω_X, with the material particles $\mathbf{X}$
- the spatial domain Ω_x, with the spatial points $\mathbf{x}$
- the referential domain Ω_χ, with the referential (or grid) points $\boldsymbol{\chi}$

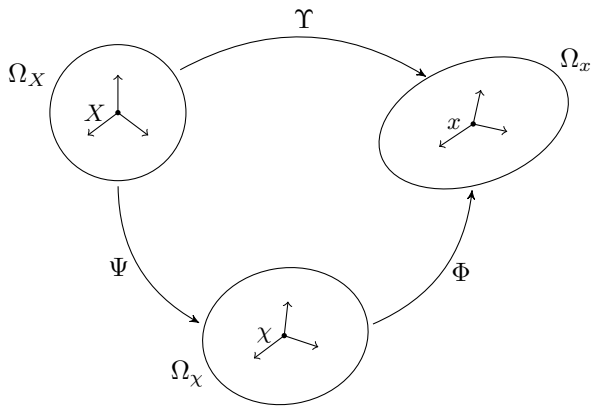


Figure 5.1: Three domains (material, spatial and referential) and the mappings Υ, Φ, Ψ between them

The velocity of the referential points $\mathbf{x}$ is given by :

$$\left.\frac{\partial \mathbf{x}}{\partial t}\right|_{\boldsymbol{\chi}} = \hat{\mathbf{v}}$$

This quantity is also referred to as the mesh velocity, since it measures the time variation of the spatial points while holding the referential points fixed. The material velocity is defined as :

$$\left.\frac{\partial \mathbf{x}}{\partial t}\right|_{\boldsymbol{X}} = \mathbf{v}$$

The particle velocity in the referential domain can be written as :

$$\left.\frac{\partial \boldsymbol{\chi}}{\partial t}\right|_{X} = \mathbf{w}$$

The relation between the velocities can be obtained by differentiating $\Upsilon = \Phi \circ \Psi$ [Donea et al. 2004], resulting in :

$$\mathbf{c} = \mathbf{v} - \hat{\mathbf{v}} = \frac{\partial \mathbf{x}}{\partial \boldsymbol{\chi}} \cdot \mathbf{w} \tag{5.1}$$

where $\mathbf{c}$ is the relative velocity between the fluid and the mesh.

Consider a scalar physical quantity described by $q(\boldsymbol{\chi}, t)$ and $q(\mathbf{X}, t)$ in the referential and in the material domain respectively. The relation between these two domains is given by :

$$q(\mathbf{X}, t) = q(\boldsymbol{\chi}, t) \circ \Psi = q(\Psi(\mathbf{X}, t), t)$$

where Ψ is the mapping between the two domains. The material derivative of $q(\boldsymbol{X}, t)$ can be expressed as :

$$\begin{aligned} \frac{Dq(\mathbf{X}, t)}{Dt} \triangleq \left.\frac{\partial q(\mathbf{X}, t)}{\partial t}\right|_{\mathbf{X}} &= \left.\frac{\partial q(\boldsymbol{\chi}, t)}{\partial t}\right|_{\boldsymbol{\chi}} + \left.\frac{\partial q(\boldsymbol{\chi}, t)}{\partial \boldsymbol{\chi}} \cdot \frac{\partial \boldsymbol{\chi}}{\partial t}\right|_{\mathbf{X}} \\ &= \left.\frac{\partial q(\boldsymbol{\chi}, t)}{\partial t}\right|_{\boldsymbol{\chi}} + \frac{\partial q(\boldsymbol{\chi}, t)}{\partial \boldsymbol{\chi}} \cdot \mathbf{w} \end{aligned} \tag{5.2}$$

By inserting the Expression (5.1) into Equation (5.2), the *fundamental ALE equation* is found and can be written as :

$$\boxed{\frac{Dq(\mathbf{X}, t)}{Dt} = \left.\frac{\partial q(\boldsymbol{\chi}, t)}{\partial t}\right|_{\chi} + \mathbf{c} \cdot \nabla q(\boldsymbol{\chi}, t)} \tag{5.3}$$

where the operator ∇ represents the gradient.

The material time derivative of a physical quantity thus consists of a local time derivative with respect to the referential domain and a convective component due to the relative velocity between material particles and referential points.

5.2 Conservation equations

In order to solve the conservation equations in the referential domain, the material time derivatives must be rewritten using the ALE formulation (5.3). A detailed development of these equations has been done by Henry [Henry 2020].

Spatial domain

The conservation equations for immersed granular flows in the spatial domain are given by [Constant et al. 2018] :

Mass :

$$\frac{\partial \phi}{\partial t} + \nabla \cdot \mathbf{u} = 0$$

Momentum :

$$\rho \left(\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot \frac{\mathbf{u}\mathbf{u}}{\phi} \right) = \nabla \cdot \left(2\mu\phi \mathbf{d}(\mathbf{u}) - p\mathbf{I} \right) + \mathbf{f} + \phi\rho\mathbf{g}$$

where ϕ is the porosity, $\mathbf{u} = \phi\mathbf{v}$ is the mean velocity of the fluid phase, p is the pressure, $\mathbf{f}$ is the force density due to the fluid-grains interaction, $\mathbf{I}$ is the identity tensor and $\mathbf{d}(\mathbf{u})$ is the rate of deformation tensor. For a Newtonian fluid, the deformation tensor is defined by :

$$\mathbf{d}(\mathbf{u}) \triangleq \left(\nabla \frac{\mathbf{u}}{\phi} + \left(\nabla \frac{\mathbf{u}}{\phi} \right)^T \right)$$

Referential domain

The conservation equations for immersed granular flows in the referential domain are given by [Henry 2020] :

Mass :

$$\frac{\partial \phi}{\partial t} - \frac{\hat{\mathbf{u}}}{\phi} \cdot \nabla \phi + \nabla \cdot \mathbf{u} = 0$$

Momentum :

$$\rho \left(\frac{\partial \mathbf{u}}{\partial t} - \frac{\hat{\mathbf{u}}}{\phi} \cdot \nabla \mathbf{u} + \nabla \cdot \frac{\mathbf{u}\mathbf{u}}{\phi} \right) = \nabla \cdot \left(2\mu\phi \mathbf{d}(\mathbf{u}) - p\mathbf{I} \right) + \mathbf{f} + \phi\rho\mathbf{g}$$

where $\hat{\mathbf{u}} = \phi\hat{\mathbf{v}}$ is the mean mesh velocity of the fluid phase. The ALE description thus makes it possible to take into account the displacement of the nodes of the mesh in the conservation equations.

Chapter 6

Pile driving

This chapter addresses the final simulation of pile driving into an immersed granular medium. The penetration of the pile into the soil leads to significant changes of the soil properties near the pile. To gain a better understanding of these changes, numerical modelling can be used to study the pile driving problem.

The concepts explained in the previous chapters are used for the simulation of pile driving. The free surface is deformed by solving the kinematic free surface equation. The mesh is regularized using the simplified Laplacian smoothing method. The deformation of the mesh is taken into account by solving the conservation equations in ALE description.

First, the experimental set-up is explained, including the geometry of the domain and the parameters of the fluid and the particles. Then, the resolution method is shown. Finally, the simulation of the driving process is analyzed for different parameter choices.

6.1 Experimental set-up

The domain is a rectangular box composed of fluid and grains, as shown in Figure 6.2. The top wall of the box is removed to simulated the free surface of the fluid. The fluid inside the box corresponds to water. Particles of different sizes are randomly deposited inside the domain to form the granular ground. The particles have a maximum radius of $1.92 \cdot 10^{-2}$ m and minimum radius of $1.12 \cdot 10^{-2}$ m. The particle size needs to be chosen large enough to make the simulations practical (limiting the CPU time). The friction coefficient between the particles is chosen to be $\mu_{p-p} = 0.5$ as proposed by Jiang *et al.* [Jiang, Yu, and Harris 2006]. The walls of the box are kept frictionless in order to improve the homogeneity [Jiang, Dai, et al. 2014].

The pile is simulated as a rigid Lagrangian body. The pile *tip* has an apex angle of 60°, which corresponds to the standard angle for pile driving. The pile *sleeve* is simulated by vertical walls. The penetration depth d of the pile is defined with respect to the origin of the domain, as shown in Figure 6.1.

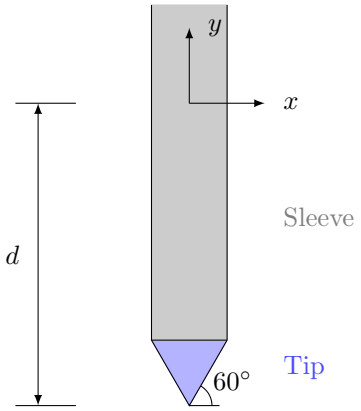


Figure 6.1: Definition of the penetration depth d

The dimensions of the pile and domain need to be carefully selected in order to minimize the boundary effects and obtain rational results. Bolton *et al.* suggested that the width of the domain L should be at least 40 times larger than the pile radius R [Malcolm Bolton et al. 1999]. Based on this finding, the radius of the pile is chosen to be $R = 0.125$m.

The initial mesh is shown on Figure 6.3. The number of vertices and elements slightly change during the simulation because of the re-meshing process explained in Chapter 4. The time step is chosen to be $\Delta t = 10^{-4}$ s. With this choice of parameters, the CFL condition (3.19) is satisfied. The parameters of the problem are presented in Table 6.1.

6.2 Resolution method

Using the concepts seen in the previous chapters, the pile driving process can be simulated. The simulation has been done in `Python`. First, the conservation equations are solved using the ALE description. Then, the position of the pile is updated. Afterward, the new location of the free surface is calculated by solving the kinematic free surface condition, using the Finite Element Method. The velocity field is corrected to ensure the conservation of volume. The mesh is then regularized by taking into account the position of the free surface and the position of the pile. Finally, if the mesh quality condition is not satisfied, a new mesh is computed in order to guarantee the stability of the solution. The solution is then projected on the new mesh. The resolution method is presented in Algorithm 2 [Henry 2020].

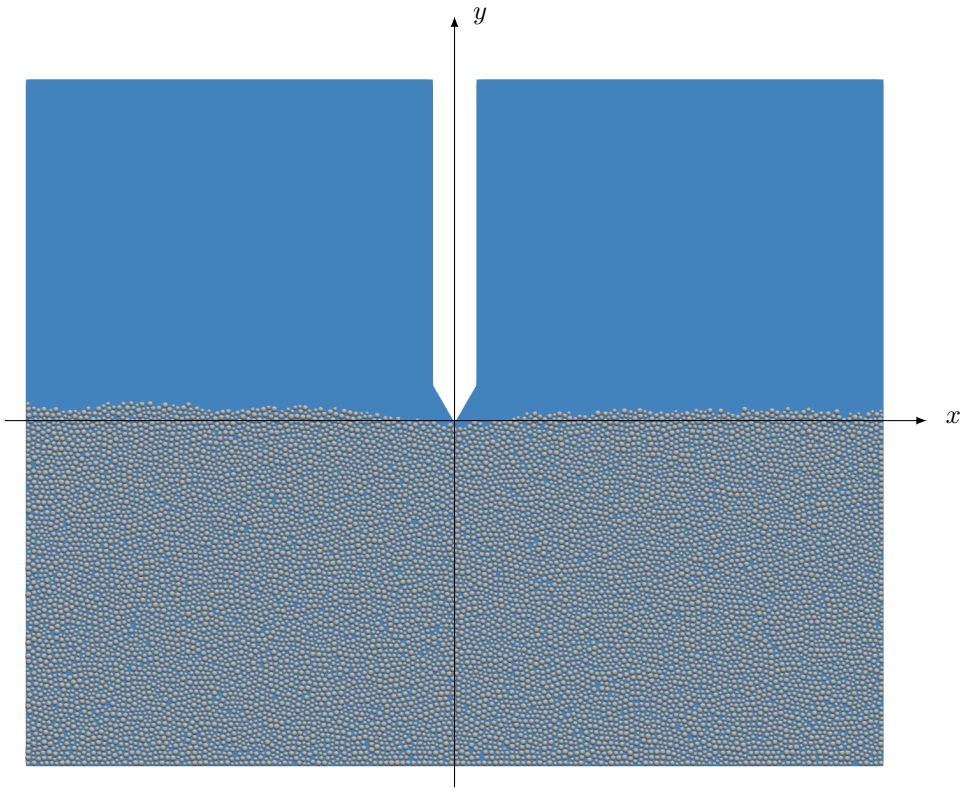


Figure 6.2: Initial configuration, pile depth $d = 0$

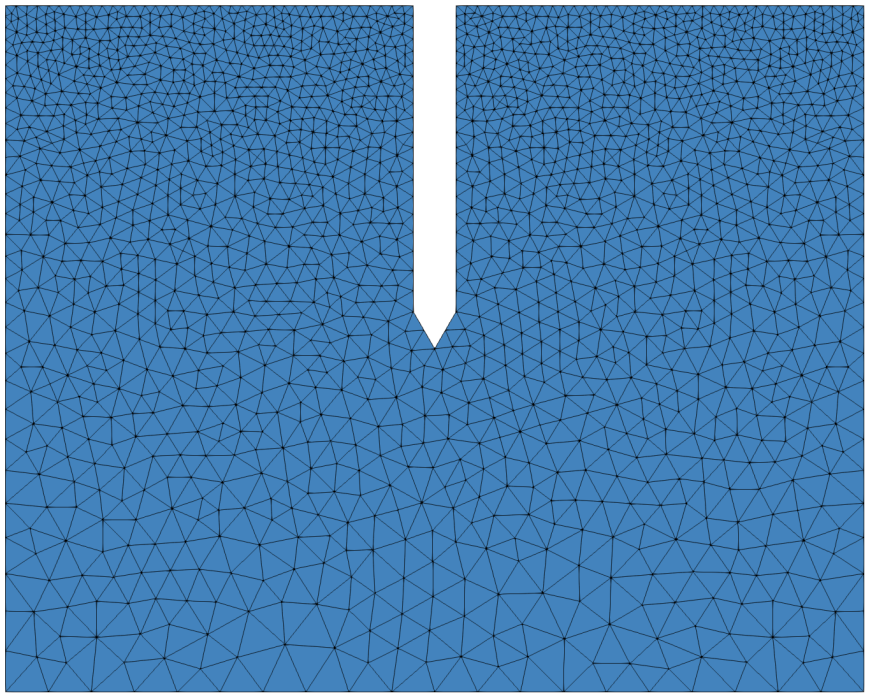


Figure 6.3: Initial mesh

Parameters	*Value*
Particles	
Number of particles	11596
Particle density ρ_p	1500 kg/m^3
Particle radius r_p	$1.12 \cdot 10^{-2}$ – $1.92 \cdot 10^{-2}$ m
Particle-particle coefficient of friction μ_{p-p}	0.5
Particle-wall coefficient of friction μ_{p-w}	0
Particle-pile coefficient of friction μ_{p-pile}	0.5
Fluid	
Fluid density ρ	1000 kg/m^3
Fluid kinematic viscosity ν	10^{-6} m^2/s
Domain	
Domain height H	4 m
Domain width L	5 m
Pile radius R	0.125 m
Pile apex angle α	60°
Simulation	
Time step Δt	10^{-4} s
Number of vertices	1948
Number of nodes	3903

Table 6.1: Parameters of the pile driving simulation

Algorithm 1: Resolution method

$t = 0$;
while $t < t_{end}$ **do**
  Time integration in ALE form $\rightarrow \mathbf{u}, p$;
  Update pile position $\rightarrow d$;
  Correction of the velocities $\rightarrow \tilde{\mathbf{u}}$;
  Update free surface position $\rightarrow h$;
  Regularization of the mesh $\rightarrow \mathbf{x}$;
  Calculation of the mesh velocity $\rightarrow \hat{\mathbf{u}}$;
  Calculation of the porosity $\rightarrow \phi$;
  if *Mesh quality criteria not satisfied* **then**
    Compute new mesh;
    Project solution on the new mesh;
  end
  $t = t + dt$;
end

6.3 Pile velocity

The penetration of the pile was performed with three different velocities to investigate the influence of penetration velocity on the pile driving process, corresponding to 0.06, 0.2 and 0.4 m/s.

The pressure on rigid boundaries can be calculated as [Jiang, Yu, and Harris 2006] :

$$p_i = \frac{\sum f_i}{h \cos(\alpha_i)}$$

where f_i is the contact force acting on the boundary, h is the length of the boundary and α_i is angle between the force f_i and the boundary. The tip resistance q_c is defined as the vertical pressure applied on the tip. Using the previous equation, the tip resistance can be calculated. For a pile with an apex angle of 60°, the tip resistance is given by :

$$q_c = \frac{\sum f_y}{R} \tag{6.1}$$

where f_y is the vertical contact force acting on the pile tip. The tip resistance is of great interest from the engineering point of view. It allows to select the adequate material of the pile tip.

Results are shown in Figure 6.4. As seen, all curves show high fluctuations due to the variable and limited number of particles in contact with the pile tip. It can be seen that the fluctuations are larger with increasing velocity. There seems to be a trend of q_c increasing with depth. This is likely due to the proximity of the bottom boundary of the domain, as explained by [Huang and Ma 1994]. The boundary gives a higher rigidity to the particles, and thus results in higher pile tip resistance values. Furthermore, the results show the tendency of tip resistance increase with the penetration velocity. This behavior can be explained by the increase in dynamic forces with greater pile velocities. The obtained results are in agreement with the numerical evidences of Esposito *et al.* [Esposito et al. 2018] and Phong [Phuong 2019].

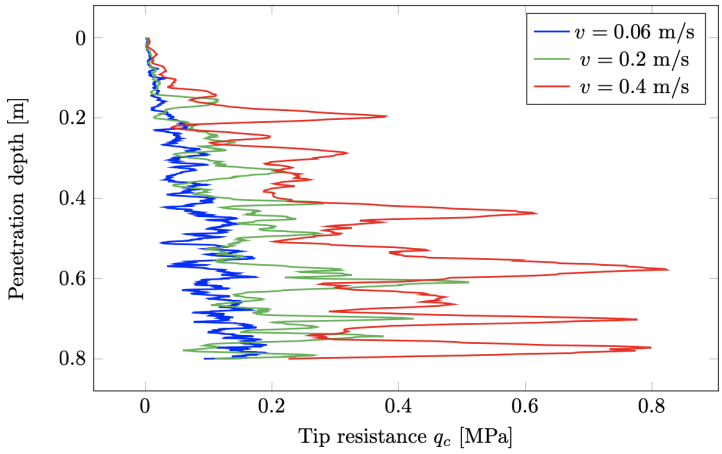


Figure 6.4: Tip resistance for different penetration velocities

6.4 Tip shape

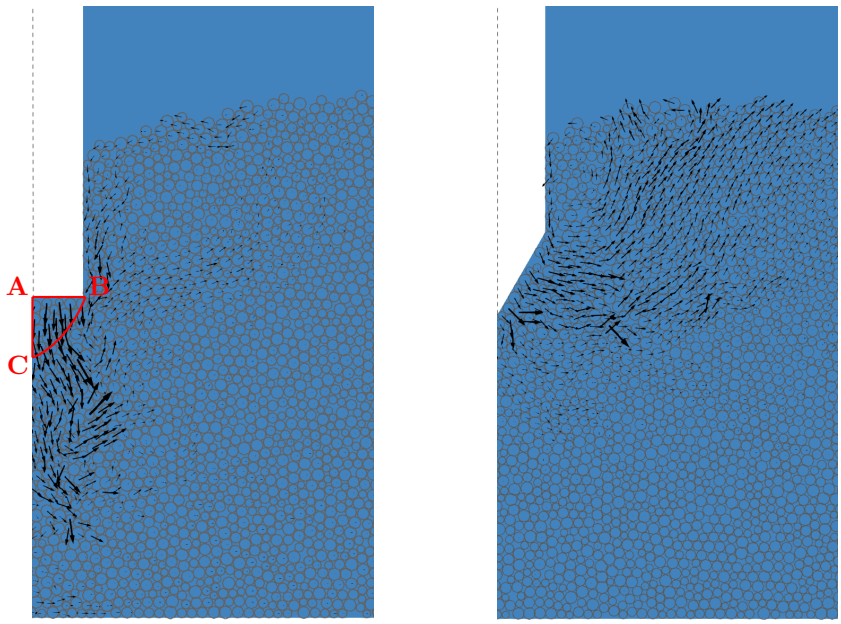
In this section, the influence of pile tip shape on the soil is examined. The triangular pile tip used in the previous sections is compared with a pile that has a flat tip. The same parameters as for the previous simulations have been used. The pile is moved downwards with a velocity of $v = 0.06$ m/s.

Displacement of soil around the pile tip

Figure 6.5 illustrates the velocity vectors of the particles, after the pile reached a depth of $d = 0.2$m.

Under the flat pile tip, the particles in zone ABC have no horizontal displacement. They are mainly moving down with the same displacement as the pile. This phenomenon has also been observed by White *et al.* [David White and Malcolm Bolton 2001] and is called the *nose cone*. The nose cone is a highly compressed region of soil below the pile tip. This zone is stationary relative to the pile tip. The penetration into the ground is possible by side movement of the elements below the tip and the upward movement of the particles just above the tip.

In contrast, there is no such *nose cone* formed under the triangular pile tip. The pile penetrates into the soil by pushing the particles laterally. A large number of particles are set in motion by the penetration of the pile, compared to the case of the flat pile tip.



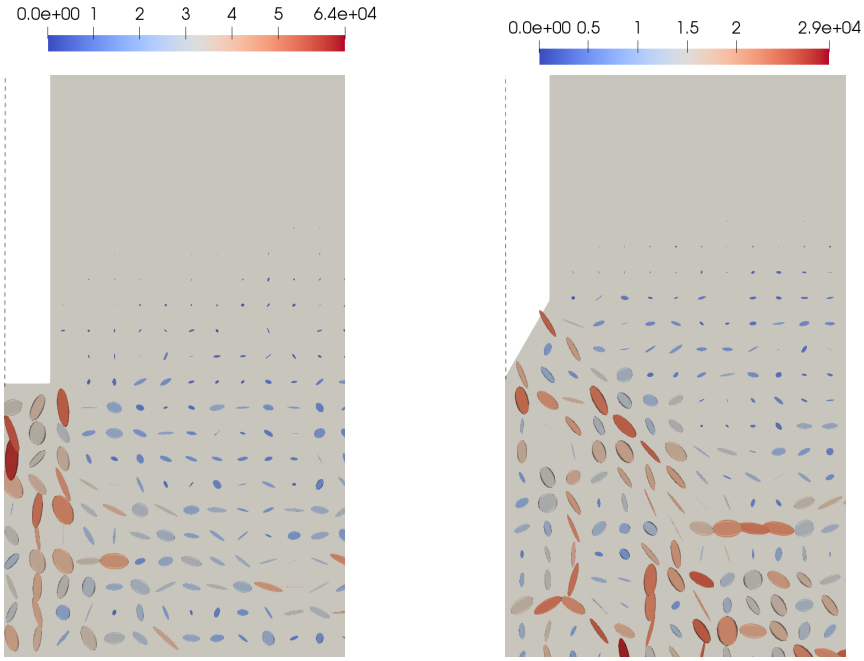
(a) Flat pile tip (b) Triangular pile tip

Figure 6.5: Velocity vectors of the particles for different pile shapes

Stresses in the soil around the pile tip

The stress state in the soil is represented by using *stress ellipses*. The length of the semi-axes of the ellipse is equal to the magnitudes of the principal stresses. Their direction corresponds to the directions of the principal axes [Marshak 1987].

The comparison of the stresses for different pile tip shapes is shown on Figure 6.6. The colour of the ellipse indicates the magnitude of the maximum principal stress. The flat pile tip generates very high vertical stresses underneath it. The stresses are up to two times larger than for the triangular pile tip. The high stresses under the flat pile tip are a direct consequence of the highly densified *nose cone*.



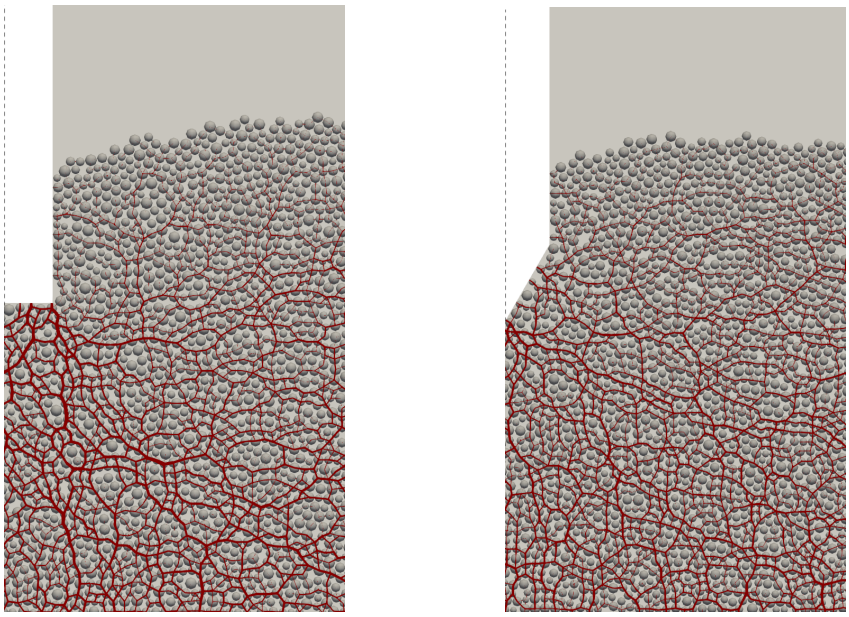
(a) Flat pile tip (b) Triangular pile tip

Figure 6.6: Stress tensor for different pile shapes

Force network

The distribution of the normal contact forces, or force network, is shown in Figure 6.7. The thickness of the lines is proportional to the magnitude of the force.

The visualization of the contact forces confirms the previous results. For the flat pile tip, the figure shows that the contact forces are concentrated in the vertical direction below the tip. The contact forces between the particles are larger for the flat tip. For the triangular tip, the forces are laterally more widespread.



(a) Flat pile tip (b) Triangular pile tip

Figure 6.7: Force network for different pile shapes

6.5 Influence of the fluid on the penetration process

When submerged in a fluid, the particle motion is influenced by the presence of the fluid. Different simulations have been done to analyse the influence of the fluid on the driving process. The fully saturated soil (for $\rho = 1000$ kg/m^3 and $\rho = 1300$ kg/m^3) is compared to dry soil.
The velocity vectors of the fluid for $\rho = 1000$ kg/m^3 are shown on Figure 6.9.

The tip resistance is calculated using Equation (6.1) and represented on Figure 6.8. It can be seen that the tip resistance is higher for the dry case than for the saturated cases. This is due to the fact that the particles move easier when they are submerged in fluid of density close to the particle density ($\rho_p = 1500$ kg/m^3). The apparent weight of the particles is reduced by the buoyancy force acting against the gravity. The buoyancy force is proportional to the density of the fluid. Thus, the pile tip resistance decreases with increasing fluid density.

Figure 6.10 shows the contact forces between the particles for different fluids. In the dry cases, strong force chains are observed, offering more resistance to the pile during penetration. To penetrate into the soil, the pile needs to break the contact chains, which demands the application of a high load on the pile. In the saturated cases, the magnitude of the contact forces decreases with the increase of the density of the fluid.
Figure 6.11 shows the stress tensor for the different situations. The stresses in the dry soil are up to twice as high as the stresses in the saturated soils. These observations confirm the previous results stating that the pile resistance is higher in the dry case.

These result show that a larger loading force on the pile is needed to achieved the same penetration depth in the case of the dry sand, than in the case of saturated sand. The liquid acts as a lubricant between sand particles and the pile. Therefore, in many civil engineering applications, water is added to the soil to facilitate the penetration process.

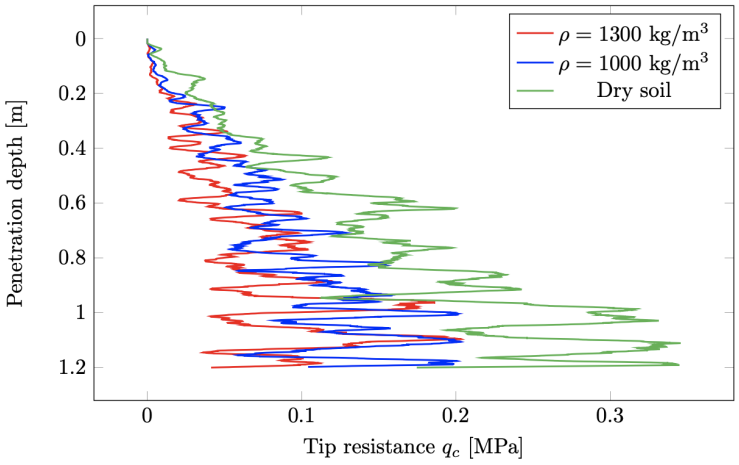
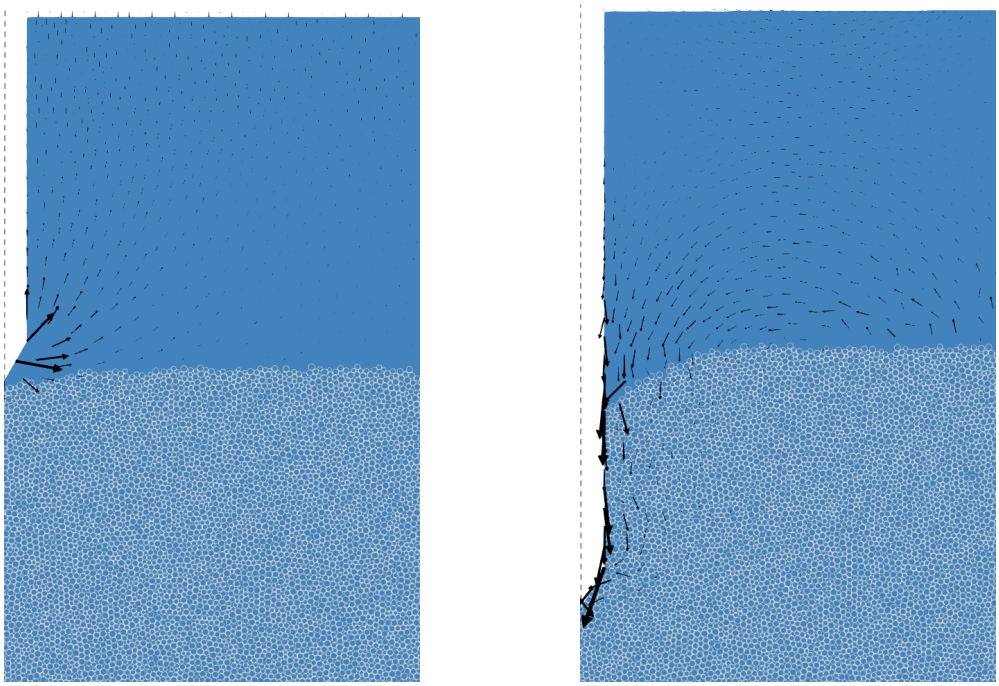
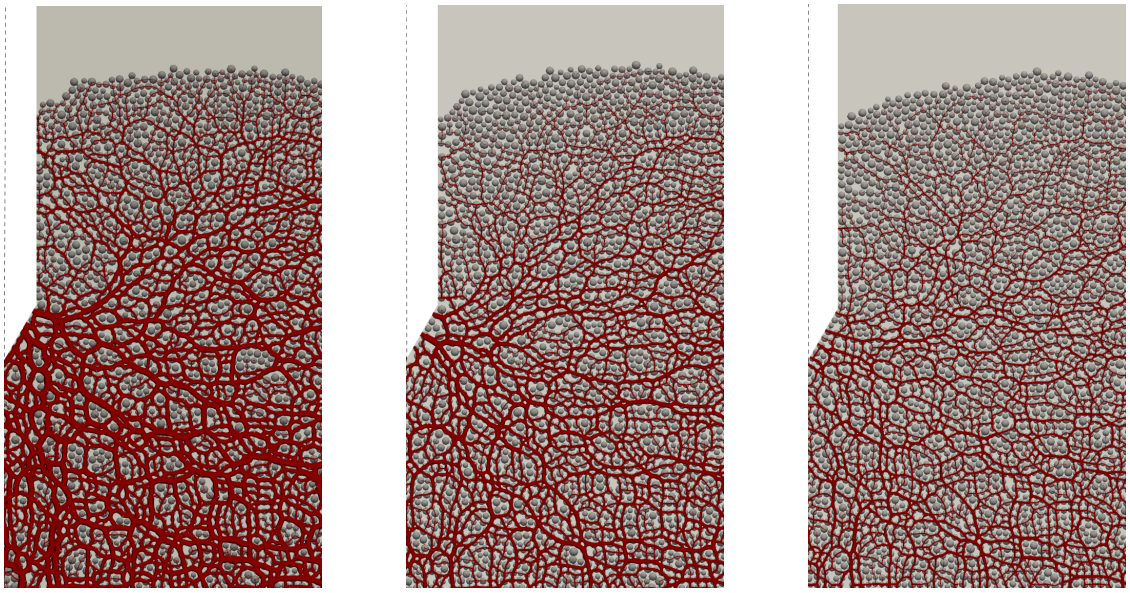


Figure 6.8: Pile tip resistance for different fluids



(a) Penetration depth $d = 0$ (b) Penetration depth $d = 1.2$ m

Figure 6.9: Velocity vectors of the fluid for different penetration depths

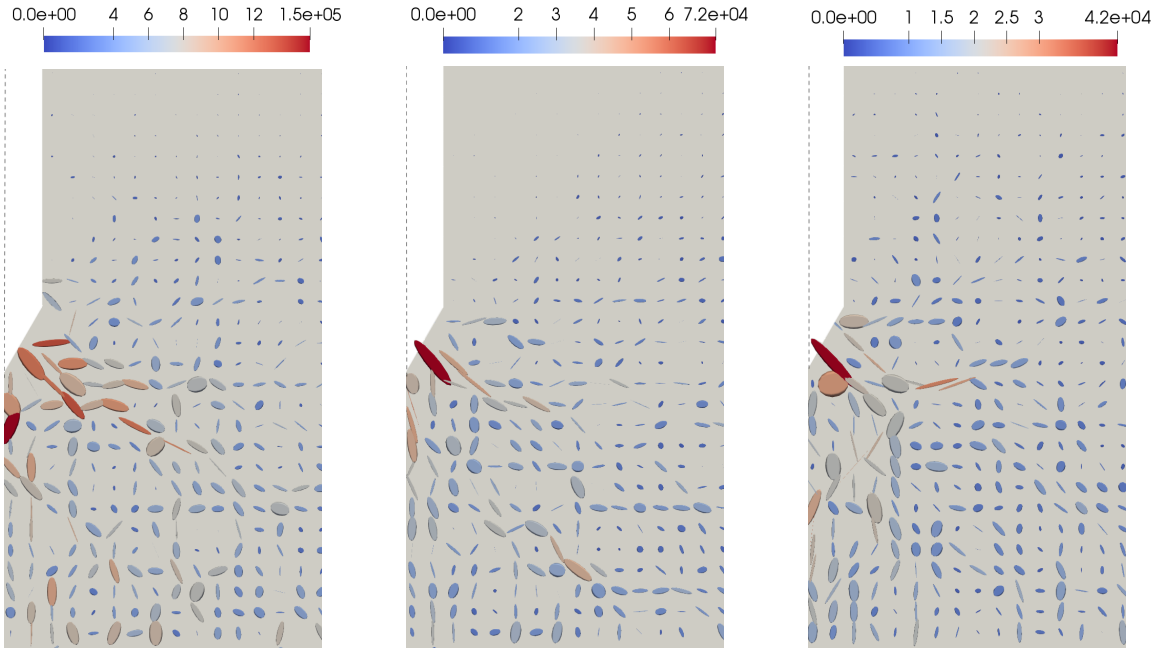


(a) Dry soil

(b) Fully saturated soil
$\rho = 1000\text{kg/m}^3$

(c) Fully saturated soil
$\rho = 1300\text{kg/m}^3$

Figure 6.10: Force network for different fluids



(a) Dry soil

(b) Fully saturated soil $\rho = 1000\text{kg/m}^3$

(c) Fully saturated soil $\rho = 1300\text{kg/m}^3$

Figure 6.11: Stress tensor for different fluids

Chapter 7

Conclusion

This master thesis addresses the challenge of simulating pile driving, making use of an already existing model, which is able to solve immersed granular flows. In order to come to a conclusion, various modeling concepts have been explored and implemented, with the prospect of adding different parts to the MigFlow software.

First, this master thesis develops the simulation of free surface flows. The kinematic and the dynamic condition, both necessary to calculate the position and the shape of the free surface, are laid out. The free surface is modeled by using an interface-tracking method. The free surface nodes of the mesh have been deformed according to the free surface position. In addition, for domains that need to conserve the volume, a correction of the velocities is proposed to ensure the conservation. The free surface equations have then been implemented numerically by using the finite element method and the finite difference method. In order to verify the equations and their correct implementation, the die-swell problem has been simulated. The simulation yielded results which are in accordance with numerical and experimental values found in the literature.

Chapter 4 presents a method to regularize the mesh. Due to the motion of the free surface nodes, the computational mesh gets deformed. To adapt the interior nodes, a mesh regularization method is needed. A simplified Laplacian smoothing method has been used and implemented numerically. The method allows to keep a regular mesh for the considered applications. Nevertheless, the method lacks robustness and requires improvements when complex free surface flows or geometries are considered. The mesh regularization method in the case of pile driving needed to be slightly adapted. Since the pile is modeled as a Lagrangian body penetrating into the domain, its motion needs to be taken into account during the mesh regularization process. As large deformations of the mesh elements were still unavoidable, endangering the stability and the accuracy of the solution, it was necessary to re-mesh the domain. A re-meshing condition has been chosen that allowed to decrease the risk of numerical errors due to the distorted elements.

Chapter 5 addresses the *arbitrary Lagrangian-Eulerian* description. In the ALE description, a reference computational domain has been used. The finite element mesh is therefore neither attached to the material nor fixed in space. The motion of the mesh can be chosen arbitrarily and independently from the motion of the material. As a result, the ALE formulation can handle large

mesh deformations, as required for pile driving, and free surface flows, while maintaining the mesh fineness. Subsequently the *fundamental ALE equation* has been introduced. Using this equation, the conservation equations for immersed granular flow have been extended to the ALE description.

The last chapter presents the final simulation of pile driving into the immersed granular soil using the concepts laid out in the previous chapters. Different situations have been tested using this simulation. First, different penetration velocities have been tested. The results show that the pile tip resistance increases with the increasing velocity. Thereafter, the influence of the tip shape of the pile has been analyzed by comparing a plat tip shape to a triangular one. Under the flat tip shape, a nose cone is formed, consisting of a highly compressed zone of particles that moved only vertically with the motion of the pile. Finally, the influence of the fluid has been analysed, by comparing the results obtained for the installation into dry soil with the ones obtained for saturated soil. The comparison shows that the fluid has an important influence on the tip resistance. The free surface seems to have no impact on the penetration process. The obtained results are in agreement with results found in the literature. Consequently, we can come to the conclusion that the simulation presented in this thesis is able to realistically reproduce the behavior of the soil and the surrounding fluid in pile driving processes.

In order to obtain results which are even closer to a realistic pile driving process, various recommendations will be briefly addressed in this paragraph. First, in order to simulate the penetration process in a more realistic manner, the domain should be larger. This way, negligence of boundary effects could be ensured better, and the pile could be penetrated deeper into the ground. Second, to simulate the pile driving into sand, the particles need to be smaller. For the purpose of this thesis, and in order to make the simulations practical and to limit the CPU time, the number of particles needed to be limited. In addition, if the free surface equation and the method of mesh regularisation were adapted, the simulation could be done in three dimensions (3D), which would allow the result to be even closer to reality. Finally, to make use of the free surface during pile driving, a vibratory force could be applied on the pile.

Bibliography

Adamson, A.W. and A.P. Gast (1997). *Physical chemistry of surfaces*. Wiley.

Airy, George Biddell (1841). "Tides and waves". In: *Mixed Sciences, Encyclopaedia Metropolitana*. Vol. 3.

André, Patrick and Jean-Robert Clermont (1987). "Numerical simulation of the die swell problem of a Newtonian fluid by using the concept of stream function and a local analysis of the singularity at the corner". In: *Journal of Non-Newtonian Fluid Mechanics* 23, pp. 335–354.

Askes, Harm and Antonio Rodriguez-Ferran (2001). "A combined rh-adaptive scheme based on domain subdivision. Formulation and linear examples". In: *International Journal for numerical methods in engineering* 51.3, pp. 253–273.

Aubram, Daniel (2015). "Development and experimental validation of an arbitrary Lagrangian-Eulerian (ALE) method for soil mechanics". In: *Geotechnik* 38.3, pp. 193–204.

Bardenhagen, Scott G and Edward M Kober (2004). "The generalized interpolation material point method". In: *Computer Modeling in Engineering and Sciences* 5.6, pp. 477–496.

Bhatia, R.P. and K.L. Lawrence (1990). "Two-dimensional finite element mesh generation based on stripwise automatic triangulation". In: *Computers and Structures* 36.2, pp. 309–319.

Bolton, Malcolm et al. (1999). "Centrifuge cone penetration tests in sand". In: *Géotechnique* 49, pp. 543–552.

Boxho, Margaux (2019). "Estimateur d'erreur efficace a posteriori pour la résolution d'écoulements granulaires immergés". MA thesis. Université catholique de Louvain.

Campos, JLE et al. (2005). "Numerical experiments with discrete elements to simulate pile penetration in granular soils". In: *Proceedings of the XXVI Iberian Latin-American Congress on Computational Methods in Engineering*.

Castro-Orgaz, Oscar and Willi Hager (2019). *Shallow Water Hydraulics*. Springer International Publishing. Chap. 1.

Constant, Matthieu et al. (2018). "Implementation of an unresolved stabilised FEM–DEM model to solve immersed granular flows". In: *Computational Particle Mechanics* 6.

Cundall, Peter A and Otto DL Strack (1979). "A discrete numerical model for granular assemblies". In: *Geotechnique* 29.1, pp. 47–65.

Deeks, AD, DJ White, and MD Bolton (2005). "A comparison of jacked, driven and bored piles in sand". In: *Proceedings of the International Conference on Soil Mechanics and Geotechnical Engineering*. Vol. 16. 4, p. 2103.

Demirdzic, Ismet et al. (1999). "Computation of Flows with Free Surfaces". In: *Scientific Computing in Chemical Engineering II*. Springer, Berlin Heidelberg, pp. 360–367.

Di, Y., J. Yang, and T. Sato (2007). "An operator-split ALE model for large deformation analysis of geomaterials". In: *International Journal for Numerical and Analytical Methods in Geomechanics* 31.12, pp. 1375–1399.

Donea, Jean et al. (2004). "Arbitrary Lagrangian–Eulerian Methods". In: *Encyclopedia of Computational Mechanics*. American Cancer Society. Chap. 14.

Drela, Mark (2018). *Lecture notes in Fluid mechanics : Streamlines, Pathlines, Streaklines*. Massachusetts Institute of Technology.

Durand, Raul, Bryan Pantoja Rosero, and Vicente Oliveira (2019). "A general mesh smoothing method for finite elements". In: *Finite Elements in Analysis and Design* 158, pp. 17–30.

Esposito, Ricardo Gurevitz et al. (2018). "Multi-scale sensitivity analysis of pile installation using DEM". In: *Computational Particle Mechanics* 5.3, pp. 375–386.

Feuillet, Rémi et al. (2018). "Connectivity-change moving mesh methods for high-order meshes: Toward closed advancing-layer high-order boundary layer mesh generation". In: *2018 Fluid Dynamics Conference*.

Gerstner, Franz (1809). "Theorie der Wellen". In: *Annalen der Physik* 32.8, pp. 412–445.

Goren, Simon L. and Stanislaw Wronski (1966). "The shape of low-speed capillary jets of Newtonian liquids". In: *Journal of Fluid Mechanics* 25.1, pp. 185–198.

Harlow, Francis and Eddie Welch (1965). "Numerical Calculation of Time-Dependent Viscous Incompressible Flow of Fluid with Free Surface". In: *The Physics of Fluids* 8.12, pp. 2182–2189.

Henke, S and J Grabe (2009). "Numerical modeling of pile installation". In: *International Conference on Soil Mechanics and Geotechnical Engineering*, pp. 1321–1324.

Henry, Michel (2020). "Interaction of Faraday waves with a granular bed". MA thesis. Université catholique de Louvain.

Hirt, Cyril W., Anthony A. Amsden, and J.L Cook (1974). "An arbitrary Lagrangian-Eulerian computing method for all flow speeds". In: *Journal of Computational Physics* 14.3, pp. 227–253.

Hirt, Cyril W. and Benjamin D. Nichols (1981). "Volume of fluid (VOF) method for the dynamics of free boundaries". In: *Journal of Computational Physics* 39.1, pp. 201–225.

Huang, An-Bin and Max Y Ma (1994). "An analytical study of cone penetration tests in granular material". In: *Canadian Geotechnical Journal* 31, pp. 91–103.

Hughes, Thomas J.R., Wing Kam Liu, and Thomas K. Zimmermann (1981). "Lagrangian-Eulerian finite element formulation for incompressible viscous flows". In: *Computer Methods in Applied Mechanics and Engineering* 29.3, pp. 329–349.

Jiang, Mingjing, Yongsheng Dai, et al. (2014). "Investigating mechanism of inclined CPT in granular ground using DEM". In: *Granular Matter* 16.

Jiang, Mingjing, H.-S. Yu, and D. Harris (2006). "Discrete element modelling of deep penetration in granular soils". In: *International Journal for Numerical and Analytical Methods in Geomechanics* 30.4, pp. 335–361.

Knupp, Patrick (2001). "Algebraic mesh quality metrics". In: *SIAM Journal on Scientific Computing* 23.

Lagrange, Joseph Louis (1781). "Mémoire sur la théorie du mouvement des fluides". MA thesis. Nouveaux mémoires de l'Académie royale des sciences et belles-lettres de Berlin.

Legat, Vincent (1992). "Méthodes d'éléments finis pour le calcul d'écoulements tridimensionnels à surfaces libres". PhD thesis. Université catholique de Louvain.

Mabsout, Mounir E and John L Tassoulas (1994). "A finite element model for the simulation of pile driving". In: *International Journal for numerical methods in Engineering* 37, pp. 257–278.

Marshak, Stephen (1987). "Stress ellipsoid". In: *Structural Geology and Tectonics*. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 750–751.

Monaghan, Joe (1992). "Smoothed particle hydrodynamics". In: *Annual review of astronomy and astrophysics* 30.1, pp. 543–574.

Muzaferija, Samir (1999). "Computation of free-surface flows using interface-tracking and interface-capturing methods". In: *Nonlinear Water Wave Interaction*. Chap. 2.

Muzaferija, Samir and Milovan Peric (1997). "Computation of free-surface flows using the finite-volume method and moving grids". In: *Numerical Heat Transfer, Part B: Fundamentals* 32.4, pp. 369–384.

Nadim, Ali (2008). "Interface Capturing Schemes for Free-Surface Flows". In: *Encyclopedia of Microfluidics and Nanofluidics*. Ed. by Dongqing Li. Springer US, pp. 841–849.

Noh, W.F. (1964). "CEL : A time-dependent, two-space dimensional, coupled Eulerian-Lagrangian code". In: *Methods in computational physics*. Vol. 3. Academic Press: New York.

Omodei, Bernhard (1980). "On the die-swell of an axisymmetric Newtonian jet". In: *Computers and Fluids* 8, pp. 275–289.

Park, Jeonghyung and Suzanne Shontz (2011). "An Alternating Mesh Quality Metric Scheme for Efficient Mesh Quality Improvement". In: *Procedia CS* 4, pp. 292–301.

Phuong, Nguyen Thi Viet (2019). "Numerical Modelling Of Pile Installation". PhD thesis. Delft university of technology.

Ramaswamy, Balasubramaniam (1990). "Numerical simulation of unsteady viscous free surface flow". In: *Journal of Computational Physics* 90.2, pp. 396–430.

Rose, Ian, Bruce Buffett, and Timo Heister (2017). "Stability and accuracy of free surface time integration in viscous flows". In: *Physics of the Earth and Planetary Interiors* 262, pp. 90–100.

Salgado, R, JK Mitchell, and M Jamiolkowski (1997). "Cavity expansion and penetration resistance in sand". In: *Journal of Geotechnical and Geoenvironmental Engineering* 123.4, pp. 344–354.

Shewchuk, Jonathan Richard (2002). *What Is a Good Linear Finite Element? - Interpolation, Conditioning, Anisotropy, and Quality Measures*. Tech. rep. In Proc. of the 11th International Meshing Roundtable.

Stokes, George Gabriel (1847). "On the theory of oscillatory waves". In: *Transactions of the Cambridge Philosophical Society* 8, p. 442.

Tezduyar, Tayfun E. (2006). "Interface-tracking and interface-capturing techniques for finite element computation of moving boundaries and interfaces". In: *Computer Methods in Applied Mechanics and Engineering* 195.23, pp. 2983–3000.

Tsai, Wu-ting and Dick K. Yue (1996). "Computation of Nonlinear Free-Surface Flows". In: *Annual Review of Fluid Mechanics* 28.1, pp. 249–278.

Vartziotis, Dimitris and Doris Bohnet (2017). "A geometric mesh smoothing algorithm related to damped oscillations". In: *Computer Methods in Applied Mechanics and Engineering* 326, pp. 102–121.

White, David and Malcolm Bolton (2001). "Displacement and strain paths during plane-strain model pile installation in sand". In: *Geotechnique* 54.

Winslow, A. (1963). *Equipotential zoning of two-dimensional meshes*. Tech. rep. California Univ., Livermore (USA). Lawrence Livermore Lab.

Appendices

Appendix A

Mesh regularization for pile driving

First, the domain is divided into different zones, as it can be seen on Figure A.1. Then, the parameters (a, b) are assigned to each zone and calculated for all interior nodes, see Figure A.2. Using these parameters, $\alpha = \max(a, b)$ is calculated for each node. The regularization method is shown in Algorithm 2.
If $\alpha = 1$, then the node only moves with respect to the motion of the free surface, FS_i. However, if $\alpha = 0$, then its adaptation depends only on the motion of the pile, Pile_i.

Algorithm 2: Interior nodes regularization

for *each interior node* i **do**
- $a = 0$;
- $b = 0$;
- **if** *node* $i \in$ *Zone 1* **then**
  - $a = 1$;
- **else**
  - $a = |(x_i - x_1)/(x_2 - x_1)|$;
- **end**
- **if** *node* $i \in$ *Zone 2* **then**
  - $b = |(y_i - y_1)/(y_2 - y_1)|$;
- **end**
- $\alpha = \max(a, b)$;
- $y_i^{new} = y_i^{old} + \alpha\ \text{FS}_i + (1 - \alpha)\ \text{Pile}_i$;

end

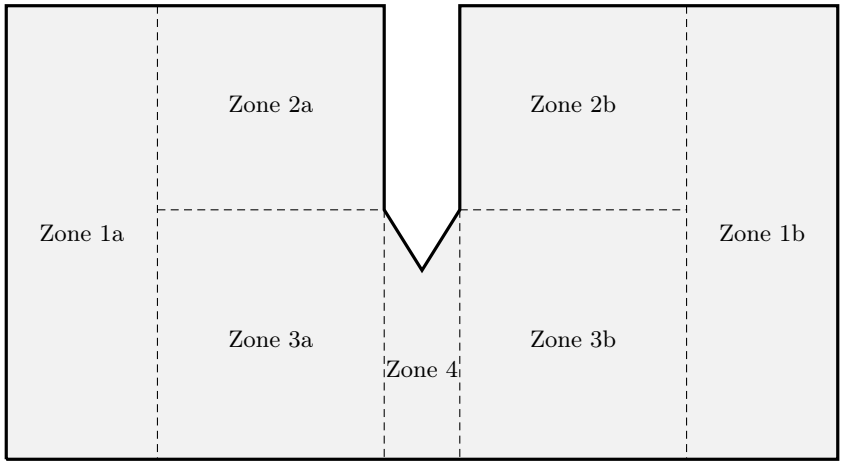


Figure A.1: Division of the domain

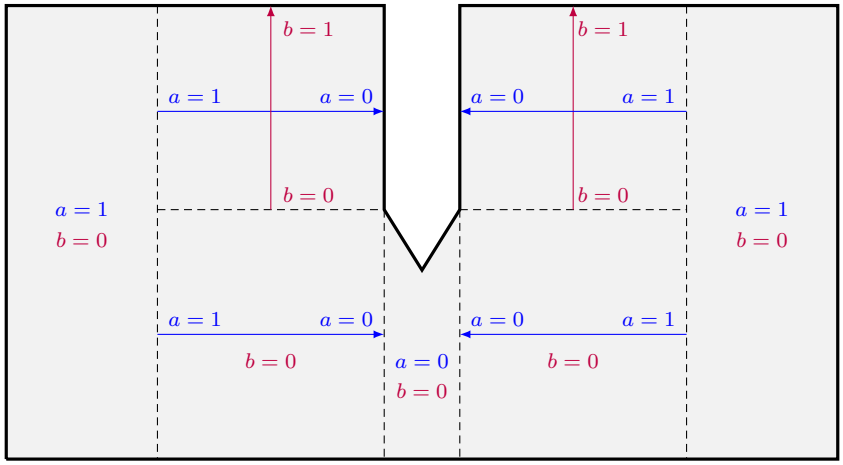


Figure A.2: Definition of (a, b) for each zone

UNIVERSITÉ CATHOLIQUE DE LOUVAIN
École polytechnique de Louvain
Rue Archimède, 1 bte L6.11.01, 1348 Louvain-la-Neuve, Belgique | **www.uclouvain.be/epl**