MSc Thesis

Neural Network-Based Mechanical Analogues in Soil-Structure Interaction

Mariana Fuente Zazueta

Delft University of Technology

TU Delft

MSc Thesis

Neural Network-Based Mechanical Analogues in Soil-Structure Interaction

by

Mariana Fuente Zazueta

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on July 17th, 2024

Student number: 5729386

Project duration: November 11, 2023 – July 17, 2024

Committee:

Dr. ir. A. Tsouvalas,	Dynamics of Solids and Structures, TU Delft
Dr. A. Tsetas,	Dynamics of Solids and Structures, TU Delft
Dr. I.B.C.M. Rocha,	Applied Mechanics, TU Delft

TU Delft

Acknowledgements

These two years have taught me much more than exclusively Structural Engineering. It was definitely a life-changing experience. Thanks to TU Delft and the whole academic team of the Track of Structural Engineering, I am thankful for the opportunity and the shared knowledge.

First of all, I would like to thank the Chair of my Graduation Committee, Dr. ir. Apostolos Tsouvalas, for everything he taught in his courses and for helping me find a Thesis topic that I enjoyed. Also, thanks to Dr. Iuri Rocha for his patience and time in explaining the complexity of Machine Learning. Last but not least, I would like to thank my daily supervisor, Dr. Athanasios Tsetas for every meeting and explanation, for guiding me through this process step by step and for rooting for me along the way.

This journey would have definitely not been the same without the people. These once strangers became friends and the closest thing to family in this lovely rainy country. Thanks to Sneha for proving to me that India and Mexico are basically the same country and bringing a sense of home since day one. Thanks to Lukas for being the best Dynamics team-mate and always telling us stories. Thanks to Emma, for being my 3rd-floor team, with our endless walks up and down the hall when we needed a break and all the laughter along the way. But also thanks to all my friends for the fun, beers in PSOR, lunches on the 6th floor, dinners on Sundays and trips around the world. Thank you, Lara, Andreas, Gabito and all the rest.

Lastly, I want to thank my family. Gracias papá y mamá por esta oportunidad, por empujarme a lanzarme al otro lado mundo a seguir creciendo, por creer en mí y por simpre ser mi mayor porra y apoyo. Gracias a mi hermana por nunca dudar de mí y animarme en los días malos. Gracias también a Coquito por siempre estar al pendiente. Gracias a Joso, por estar ahí todos los días, los buenos celebrando y los malos abrazando a la distancia. Sin ustedes esto no hubiera sido posible.

Mariana Fuente Zazueta
Delft, July 2024

Summary

It is well-established that the effect of the soil significantly influences the overall behaviour of structures, particularly in the presence of vibrations. The majority of the developed approaches can be categorised into two groups: the direct approach and the substructure method. The first one is based on detailed modelling of the soil, foundation and structure using FEM and is not commonly used because it is computationally expensive and thus not feasible for standard engineering practice. The second method uses the dynamic stiffness matrix of the substructure to include it in the analysis of the general structure. Due to the impracticality of the first method, various methodologies have been investigated to better address SSI through the dynamic stiffness matrix, also known as impedance functions. They provide an approximate representation of the SSI that facilitates quicker analysis and sufficiently accurate results.

These approximation techniques typically involve intricate systems composed of lumped masses, dashpots, and springs. The characteristics of these elements vary based on soil and foundation types. For instance, a circular plate laying on homogeneous soil can be modelled as a mass, spring, dashpot single-degree-of-freedom system, where the coefficients of the elements represent the SSI of the disc. Similarly, a pile embedded in soil can be modelled with springs on the side that represent the effect of the friction between the structure and the soil. Some of the existing models for these two cases will be studied in this work.

Four mechanical analogues will be computed and compared for the first SSI problem: Lysmer's, Kausel's, and two of Wolf's models. These models define the mass, stiffness, and damping coefficients differently, but yield similar responses. Additionally, the results will be compared to a finite element (FE) model with the same soil and foundation characteristics to assess their applicability. For the second SSI problem, Novak's solution for piles under vertical vibrations will be studied and compared to the response of an FE model. The relevance of these SSI problems lies in the simplicity of reproducing the analogues to use as a reference. They are suitable benchmark problems to evaluate an alternative computational approach that is proposed in this work.

This work's objective is to determine the feasibility of analysing the response of these systems using a hybrid computational approach. This method involves generating responses using an FE model with a range of soil and foundation parameters as a first step. Secondly, these results are input into a Neural Network (NN). The NN is trained to reproduce the response of any system by knowing only the geometric characteristics of the foundation, and the shear modulus, Poisson's ratio, and density of the soil. The network's output is the frequency response function (FRF) of said system, which is subsequently used in an optimisation process. The output of this second process is the values for the system's mass, stiffness, and damping characteristics. Ultimately, these values can be used to construct the dynamic stiffness matrix necessary for solving the soil-structure interaction of a particular structure.

The central question addressed in this research is: What constitutes an efficient data-driven process for translating large soil-foundation datasets into simplified mechanical analogues that provide a reduced and accurate representation of the system? After assessing its applicability, it can be concluded that a hybrid two-step computational approach is an efficient and generic method for solving SSI problems.

Contents

Nomenclature

Abbreviations

Abbreviation	Definition
2D	Two-dimensional
3D	Three-dimensional
adam	Adaptive Moment Estimation
AI	Artificial Intelligence
ANN	Artificial Neural Networks
DoF	Degree-of-Freedom
DSM	Dynamic Stiffness Matrix
EoM	Equation of motion
FE	Finite Element
FEM	Finite Element Method
FNN	Feedforward Neural Networks
FRF	Frequency Response Function
ML	Machine Learning
MLP	Multi-Layer Perceptron
NN	Neural Networks
ReLU	Rectified Linear Unit
sgd	Stochastic gradient descent
SSI	Soil-structure interaction
tanh	Hyperbolic tangent activation function

Symbols

Dynamics of structures

Symbol	Definition	Unit
a_0	Dimensionless frequency	[-]
A	Cross-sectional area	[m^2]
C, c	Damping	[Ns/m]
c_s	Shear wave velocity	[m/s]
f	Frequency	[Hz] or [1/s]
$\tilde{F}$	Flexibility matrix	[m/N]
E	Young's modulus	[Pa]
Em	Embedment depth	[m]
G	Shear modulus	[Pa]
$\Im$	Imaginary part	[-]
K	Stiffness of element or system	[N/m]
$\tilde{K}_{dyn}$	Dynamic stiffness matrix	[N/m]
L	Length	[m]
M, m	Mass	[kg]
P_0, $\tilde{p}$	Force	[N]
$\Re$	Real part	[-]
S, Z	Impedance	[N/m]
u	Horizontal displacement	[m]
r	Radial cylindrical coordinate	[m]
r_0	Radius of element	[m]
t	Time	[s]
w	Vertical displacement	[m]
ϕ	Rotational displacement	[rad]
γ	Dimensionless damping coefficient in Wolf's model	[-]
θ	Azimuthal cylindrical coordinate	[rad]
λ	Wavelength	[m]
μ	Dimensionless mass coefficient in Wolf's model	[-]
ν	Poisson's ratio	[-]
ξ	Damping coefficient	[-]
ρ	Density	[kg/m^3]
σ	Stress	[Pa]
ω	Angular frequency	[rad/s]

Artificial Intelligence and Machine Learning

Symbol	Definition
b	Bias
h	Activation function
L	Loss
t	Target
w	Weights

Symbol	Definition
x	Input
$\hat{y}$	Predicted output
λ	Regularization term

1

Introduction

"The concept of soil-structure interaction refers to static and dynamic phenomena mediated by a compliant soil and a stiffer superstructure" [19]; this is how Kausel briefly defines soil-structure interaction. However, this area encircles the extensive study of the response of soil and structures subjected to static and dynamic forces, like heavy machinery and earthquakes. It does so by making various assumptions and analysing the effects of such loads on structures and the underlying soil. On that account, due to its complexity, scope, and significance, it is expected that many scientists from different disciplines have investigated the topic.

The first ones to examine soil-structure interaction were Lamé and Clapeyron at the beginning of the 19th century [19]. They studied the possibility of modelling an infinite domain, such as the soil, as a *halfspace*. The results were not conclusive and it was not until 1848 that Sir William Thomson found a fundamental solution[1][19] [54]. More complicated scenarios, assumptions and applications arose and were inspected. In the late 1800's, Boussinesq derived the solution to the stresses in a soil halfspace due to a vertical load [19]. After him, it was Lamb who found a fundamental solution for a homogeneous halfspace subjected to a dynamic load [22], and to this day this is still referred to as *Lamb's problem* [19]. Like them, countless other scientists studied, proved and compared solutions for soil-structure interaction problems during the 19th and 20th centuries.

Beyond scientific approaches and solutions that explore soil-structure interaction, it has been observed that the surrounding soil significantly affects the substructure's contribution to the overall behaviour of structures. Housner [14] and many subsequent researchers have shown that the amplitude of a structure's response to incoming waves, when soil-structure interaction (SSI) is considered, differs from that of a structure on a fixed base. In other words, the overall dynamic stiffness of the structure is modified by the interaction between its substructure and the soil. This interaction typically results in an elongation of the structure's natural period and an increase in effective damping, thereby reducing the base shear, a beneficial outcome from a civil engineering perspective [1]. This particular static and dynamic effect arises due to two main reasons. Firstly, the difference in stiffness between the substructure and the soil. Secondly, the coupled behaviour between the structure and the soil, as the displacements of both are dependent on and affect each other recursively. However, recent studies have concluded that neglecting SSI is not always advantageous to the overall behaviour of the structure. Therefore,

[1]"A fundamental solution is an analytical expression for the response anywhere in a solid elicited by a static or dynamic point source at some arbitrary location" [19]. It is also known as *Green's functions* [18]. In other words, the fundamental solutions are particular solutions of differential operators for a point force in space and time in the form of a Dirac-delta function applied in an unbounded domain [6].

understanding this topic is crucial to better estimating the response of structures under various loading conditions.

One main effect of soil-structure interaction in the behaviour of the superstructure is that an important amplification of the structure's response can occur with incoming waves if the natural period coincides with the highly energetic response of the soil [50]. Furthermore, it was observed in Mexico City's and Kobe's earthquakes, among others, that the interaction between the foundation piles and the soil cannot be assumed to be always beneficial to the behaviour of the structure as it was believed in the past. Gazetas [9] investigated that the horizontal displacement of the pile, especially at the cap, can be significantly large due to the fixity of the pile to it. Along with the previous work, other studies have proved that inelastic structures in particular cannot ignore the effect of the interaction between soil and sub-structure [1], especially in the presence of earthquakes.

This suggests that, from both geotechnical and structural perspectives, the study of soil-structure interaction remains a challenging and ongoing area of research. For example, modern developments have introduced more sensitive structures where the consequences of failure are more significant, like nuclear plants [27] and offshore wind turbines [25] [20]. These structures greatly benefit society, but their failure can be fatal, so ensuring proper performance is crucial. Additionally, urban growth necessitates the construction of more residential buildings, leading to a denser presence of foundations in the soil. The interaction between pile-soil-pile has been studied in the later years and is also known as the pile-group effect [25] [17]. This condition is also found on offshore wind-turbine parks and has been proved to have an additional effect on the behaviour of the structure and therefore should be further investigated [25]. Accordingly, in contemporary times practically every structural project should rigorously analyse and design the substructure considering its interaction with the subsoil. However, there are still several challenges to overcome in this field of study.

Firstly, the soil domain is challenging to model and understand due to the proportions and complexity of 3D motion in heterogeneous media. Particularly in the presence of waves, the soil exhibits radiation, meaning that the waves attenuate as they travel through it due to energy spreading in a semi-infinite space. Soil-structure interaction models must account for this phenomenon to ensure that waves are not artificially reflected in the structure, which would otherwise alter the computed dynamic response. This characteristic of the soil, along with many others, makes the task of modelling it a significant challenge.

Far-reaching domains help to ensure the radiation of the waves in numerical models. But they also require time-consuming simulations that cannot always be implemented. In engineering practice time constraints are decisive, and thus the need for meticulous analysis does not mean that lengthy and time-consuming solutions are preferred. Additionally, conventional computers are still insufficient to solve extensive Finite Element (FE) models with large soil domains. It is still very computationally expensive to model a three-dimensional domain to examine the actions and effects between structure and soil. Resultantly, faster solutions are demanded and extensive simplifications are made in practice. These simplifications imply a vast number of assumptions that are rarely representative of the physical problem considered. For example, soil is usually considered linear elastic, whereas this is not necessarily true. It is mostly modelled as homogeneous and isotropic, while the soil's characteristics are highly variable in space. Consequently, the results may differ greatly from reality and are simplistic representations of the interaction between soil and structure. Ultimately, the accuracy of these simplifications – even though computationally advantageous – is questionable.

Throughout the years, scientists, mathematicians and engineers have tried to develop better methods to solve SSI problems. As detailed by Kausel [19], various approaches have been

studied to obtain better results. These approaches go from more specialised solutions for specific cases like the simplified model for large-diameter caissons proposed by Varun et al [56], to more advanced models of existing ones that grasp the system's complexity better like the model for piles proposed by Pang et al. that takes the softening or hardening of the soil (sand in this case) into account [39], or new methods that incorporate an additional characteristic of the system like the effect of the length-to-diameter ratio in the lateral response of piles as studied by Wang et al. [58]. Nonetheless, this wide variety of models and solutions lacks a mechanical analogue that applies to every kind of foundation.

Having more specialised and specific solutions for particular problems solves the issue of making important simplifications and assumptions that are physically inaccurate. Scientists usually implement more complexity in the models and bring the solution closer to reality. However, the immense variety of projects and their solicitations make this approach not widely applicable and more complex models are still not preferred in engineering practice. In contrast, discovering a generic framework for addressing all types of soil-structure interaction problems could yield more precise outcomes for the industry. This would greatly enhance the ability to construct and develop projects using superior tools.

This thesis aims to develop a generic computational approach for obtaining the numerical solution of SSI problems that can be applied in different foundation scenarios. The objective is to derive mechanical analogues that closely approximate the true dynamic behaviour of the soil-foundation system and can be incorporated in the system's mass, stiffness, and damping matrices to then be used to solve the full SSI problem. Achieving this would enhance the accuracy of current models in practice without increasing their computational size and complexity. To accomplish this, Machine Learning will be employed to derive the mass, damping, and stiffness matrices of simple SSI systems, such as a disc over homogeneous soil halfspace and a pile in a soil stratum. Existing analogues provide sufficiently accurate solutions for these foundations, serving as references and benchmarks for the present computational approach. Using these models as a reference will be the first step to assess whether the approach can be applied to more complex SSI problems, where lies its true potential due to its generic character and the absence of other approximate solutions.

1.1. Soil-Structure Interaction

Soil-structure interaction refers to the effect of the soil on a structure's behaviour, which is studied especially when subjected to external dynamic excitation. The description of the concept of soil-structure interaction is a challenge as the topic is wide and complex. In the definition given by Kausel in the previous section, the terms *static* and *dynamic* are already extensive and enclose infinite kinds of static and time-dependent loads [19]. Similarly, the *compliant soil* might have unlimited variations in characteristics and a broad variety of foundation arrangements in it. However, the real sophistication relies upon the approach rather than the description.

Structural engineering focuses on the analysis of the general behaviour of a structure and the design of the individual elements that compose it. This part of any structural system is sometimes referred to as *superstructure* and it lies on the *substructure* or *foundation*. The latter has finite dimensions and is embedded in the soil, which is semi-infinite, and its function is to support the structure and transfer the loads to the soil. The general problem is illustrated in Figure 1.1. In this image, the bounded superstructure represents any construction or building lying on an unbounded soil medium. The soil-structure interface indicates the layer at which the two domains interact, where the incident waves arrive and the reflected waves are radiated into the soil's unbounded domain. The free-field motion indicates a point where the vibrations are usually measured and to which seismic engineering refers for earthquake analysis. It stands for the response of the soil without the presence of the structure or the effect of an excavation. The unbounded soil is also referred to as *halfspace*, which is a mathematical model used to represent the infinity soil in two dimensions [60] [7].

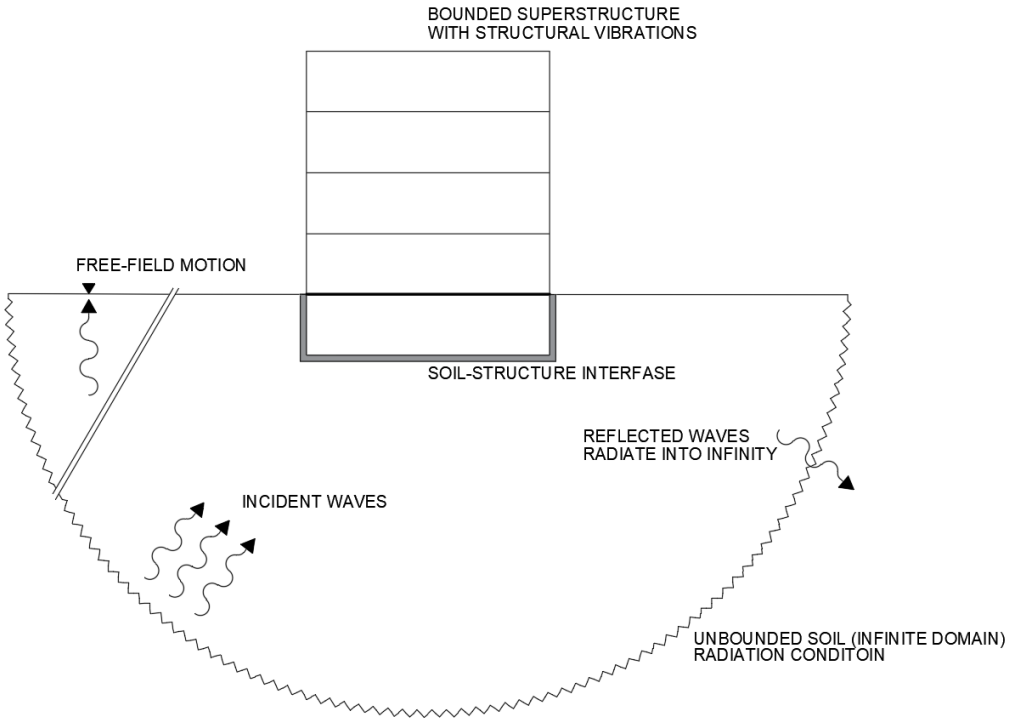


Figure 1.1: General representation of SSI problems [60].

For any SSI problem, the loads are either *external* or *internal*, and it is the relation of both that is analysed. On one hand, the external forces refer to the dynamic loading exerted on the structure, like machinery, wind loads, moving loads like trains or the movement of the

structure due to earthquakes, where the *inertial* forces are transferred to and radiated through the ground. This is also referred to as *inertial interaction*, and it represents an increase in base shear, moment and torsion of the foundation which translate into displacement and rotation in the interaction between soil and foundation [29]. On the other hand, internal loads refer to the ones applied through the soil and transmitted to the structure, like earthquakes, where the structure reflects part of the waves and modifies the soil around the foundation [7]. It is also known as *kinematic interaction*, the wave passage or travelling wave problem. It depends on the geometry of the foundation and the soil around it, although it mainly refers to "the difference in the arrival times of seismic waves at different locations along [...]" [57] the foundation. It can be specially observed in embedded foundations, long-span bridges or buildings. [7].

These two interactions play a role in the conventional solution methods used for SSI. There are two main methods: the *direct approach* and *substructure approach*. However, regardless of the solution method, the input dataset for the models relies largely on the properties of the soil and how it is characterised. The soil can be mainly described by its density, Poisson's ratio and shear modulus. Their relevance in SSI is outlined below.

- The *density of the soil* - ρ [kg/m^3]. This parameter is needed to calculate the velocity of the shear waves in the soil, playing an important role in the problem at hand.
- The *Poisson's ratio* - ν [-]. This parameter indicates the ratio between the transversal strain and the axial strain. In other words, it measures the lateral deformation given by the deformation in the axial direction.
- The *shear modulus of the soil* - G [Pa]. This parameter relates the shear stress to the shear strain of the soil and is directly related to the Young's modulus. It is used in determining the shear wave velocity of the soil.

These parameters are needed to obtain the shear wave velocity (c_s), which plays a crucial role in SSI. It is upon it that the definition of the radiation conditions or the wavelength (λ) is defined, which also depends on the frequency of the motion (f). The respective equations are Equations 1.1 and 1.2.

$$c_s = \sqrt{\frac{G}{\rho}} \tag{1.1}$$

$$\lambda = \frac{c_s}{f} \tag{1.2}$$

where c_s is the wave velocity, G is the shear modulus of the soil, ρ is the density of the soil, λ is the wavelength and f is the frequency.

After outlining the main aspects of soil-structure interaction, the conventional methods used to solve SSI problems can be described. Understanding these approaches elucidates the application and objective of this work, which is to derive the reduced mass, stiffness, and damping matrices from the soil-foundation (or soil only) to facilitate structural analysis. Further details are provided in subsequent sections.

1.1.1. Direct approach

This approach analyses the soil as a continuum and solves it simultaneously with the structure. To do so, FE models are commonly used, making the elements in the foundation and the soil match (see Fig. 1.2). It considers a truncated soil boundary that must have the capacity to simulate wave radiation. Moreover, the input force must be applied at the boundary. The FE model requires a sufficiently fine mesh to capture the information of the shortest wavelength

and an appropriately large domain to capture that of the longest wavelength [55]. The main advantage is that it allows the simulation of systems that present material or geometric non-linearity.

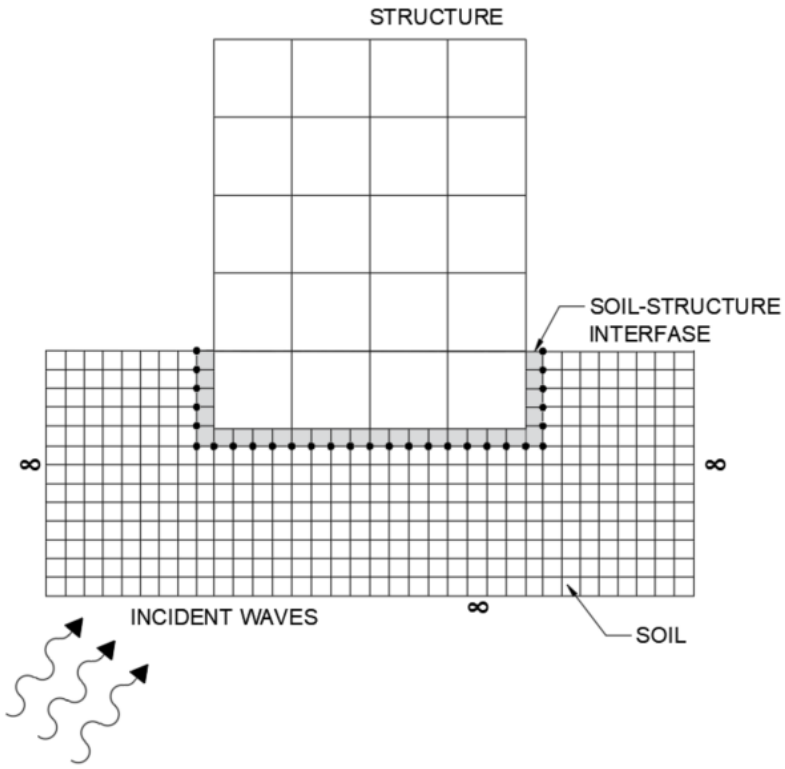


Figure 1.2: Solution method: Direct approach [55].

However, this method is not commonly used in seismic engineering. Since an earthquake motion is recorded with seismographs placed in arbitrary locations, the measured accelerations correspond to that specific location and cannot be extrapolated to other zones. This means that the incident waves for a specific project with a particular location and characteristics cannot be accurately defined. On one hand, the acceleration in the area is not known. On the other hand, the accelerations from the free field do not give all the information about the waves themselves. Additionally, the computational expense is so high that it is rarely used in standard engineering practice and only in special cases.

1.1.2. Substructure approach

The second method, contrarily to the direct approach, divides the system into two domains: the structure and the soil (see Fig. 1.3). To solve it, it uses the principle of superposition, hence it assumes a linear behaviour for both the soil and the structure. Firstly, the free-field motion with the soil's characteristics is evaluated. Then the free-field motion is transferred to foundation input motions and translates the stiffness and damping of the soil into springs and dashpots. Afterwards, the dynamic stiffness matrix ($\tilde{K}_d(\omega)$) is obtained for the correspondent displacement and rotational degrees of freedom. These springs and dashpots are characterised as *impedance functions* given by certain models, like Pais-Kausel's for embedded slabs or Novak's for piles.

Ultimately, the structure's response with the springs and dashpot subjected to the founda-

tion input motion is solved. In other words, the dynamic stiffness matrix ($\tilde{K}_d(\omega)$) substitutes the soil domain when solving the structure [29]. Additionally, the soil structure interface must satisfy the force equilibrium and the displacement compatibility between the two domains [55].

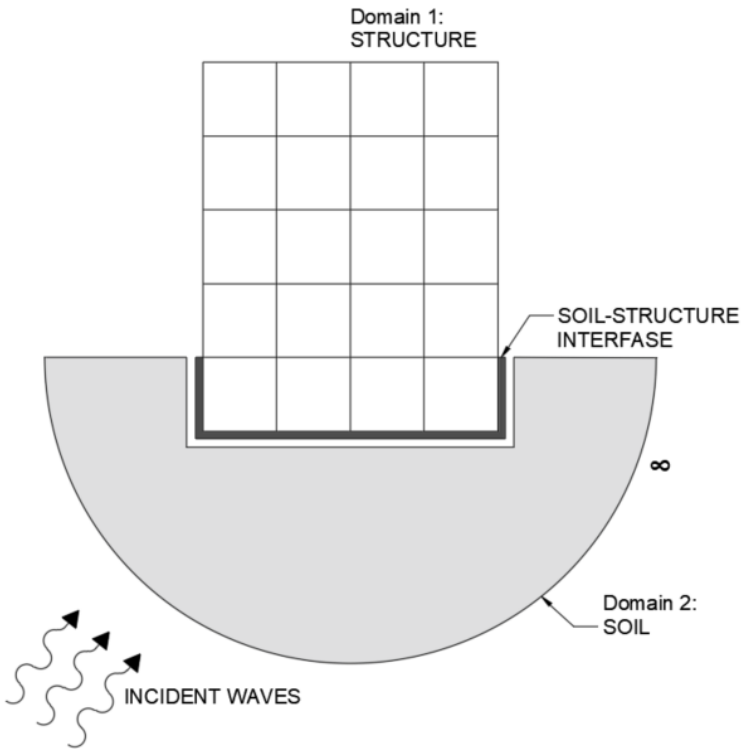


Figure 1.3: Solution method: Substructure approach [55].

This approach requires two equations of motion to be solved. The first one solves the response of a massless structure and represents the kinematic interaction given at the arrival of waves from the soil. The second one uses the results obtained from the first equation and accounts for the inertial interaction. It solves the response of the structure (with mass) subjected to inertial forces. In this case, the springs and dashpots calculated with the impedance functions are optionally attached to the structure's analysis. Both equations of motion are presented in Equations 1.4 and 1.3 respectively, as defined by [7].

$$\boldsymbol{M}\ddot{\boldsymbol{u}}_1 + \boldsymbol{C}\dot{\boldsymbol{y}}_1 + \boldsymbol{K}\boldsymbol{y}_1 = 0 \tag{1.3}$$

$$\boldsymbol{M}\ddot{\boldsymbol{y}}_2 + \boldsymbol{C}\dot{\boldsymbol{y}}_2 + \boldsymbol{K}\boldsymbol{y}_2 = -\boldsymbol{M}_2\ddot{\boldsymbol{u}}_1 \tag{1.4}$$

In these equations, $\boldsymbol{M}$ is the total mass, which results from the sum of $\boldsymbol{M}_1$, the mass of the system without the structure, and $\boldsymbol{M}_2$, the mass of the structure only. Additionally, $\boldsymbol{y}$ is the relative displacement between the structure and the ground: $\boldsymbol{y} = \boldsymbol{u} - \boldsymbol{u}_g$, where $\boldsymbol{u}$ is the absolute displacement – the displacement of the entire system ($\boldsymbol{u} = \boldsymbol{u}_1 + \boldsymbol{u}_2$) – and $\boldsymbol{u}_g$ is the displacement of the ground. Furthermore, $\boldsymbol{u}_1 = \boldsymbol{y}_1 + \boldsymbol{u}_g$, $\boldsymbol{u} = \boldsymbol{u}_1 + \boldsymbol{y}_2$ and $\boldsymbol{y} = \boldsymbol{y}_1 + \boldsymbol{y}_2$

In this approach, the dynamic stiffness matrix depends on the foundation's geometry and flexibility, the level of embedment (depth into the soil), the soil characteristics, and the excitation force. It considers the six degrees of freedom – three translational (in x, y and z) and three rotational (around x, y, and z) – that the foundation can be subjected to. It applies to

both shallow and deep foundations, with various solutions for different foundation types. An example of a shallow foundation is a disc over a homogeneous halfspace, which is thoroughly explained in Section 2.2. Conversely, an example of a deep foundation is a pile embedded in the soil, for which a solution method is detailed in Section 2.3.

A slightly modified version of this method is called the *Three-step approach*. This method is used for foundations that can be considered infinitely rigid and massless [7]. It allows to completely remove the effect of the structure's mass from the system and solve it in three steps, hence the name. However, the similarity to the substructure method is vast but considers an extra step. For this reason, it is not commonly used.

1.2. Conventional models for Soil-Structure Interaction

Most cases of SSI are not solved using the direct approach but rather the substructure approach that solves the response of soil on its own to obtain the impedance functions and the dynamic stiffness matrix. This is what most conventional analogues are based on. The approach is not necessarily correct or exact, but it is evident from recent studies that it is attractive in practice to this day, as it is still being largely studied and applied ([56], [58], [39], [5]). Many different scientists and engineers have searched for the solution of various types of foundations by substituting the reaction of the soil with springs and dashpots. These can be frequency-dependent or independent, yet the former are not applicable to time-domain analysis. Further, they can be positioned in diverse arrangements but are based on the same principles.

After the first solutions to simple SSI problems, more complicated and accurate solutions have been investigated by different scientists. One example is the truncated cone model for the disc over soil proposed by Velestos and Meek [60] and then extended by Wolf [59]. The first approximation gave reasonable results for a lower range of frequencies, while Wolf managed to find a solution for a wider range. These models are based on 1D rod theory and use the basic characteristics of the soil to build the cone with its apex height for each degree of freedom. Furthermore, the cone is comprised of infinitesimal soil layers with a stiffness equivalent to that of a rod subjected to a unit load or displacement [60].

From this model, it was observed that the stiffness of the soil could be considered frequency-independent and thus could be modelled as a conventional spring. Moreover, the damping property of the soil could be modelled as a frequency-independent dashpot and then the whole system could be modelled as a single-degree-of-freedom system. Models like these, which are explained in more detail in Section 2.2, were proposed by Lysmer [24], Pais and Kausel [38], and Wolf [59] among many others like Chen et al [4]. Furthermore, Wolf proposed an adaptation of this model with an additional degree of freedom to fit the response better, called the Monkey-tail model.

Eventually, more complex versions of these models were developed, such as nested lumped-parameter models proposed by Wu and Lee where, instead of having one spring and one dashpot, several of these are arranged in parallel and in series to better grasp the complexity of the system. Moreover, this nested principle allowed for other foundation arrangements to be modelled using springs and dashpots (see Fig. 1.4) [62]. Furthermore, a mathematical approach to systematically solve these nested models was proposed by the same authors to reduce the mathematical complexity of its solution [61].

However, not all foundations in practice resemble the disc over soil and the necessity for specialised solutions arose. When it comes to deep foundations, piles are an important example to be studied. Novak proposed several analyses and solutions that consider the flexibility of a slender pile and the different displacements along its length [37], [31] [30], [35] (refer to Section 2.3 for Novak's model). On top of this, further investigations have been made to solve more

specific problems, like the simplified model for large-diameter caisson foundations developed by Varun et al [56]. They proposed a solution for this type of intermediate foundation, that is embedded but not as deeply as piles, where the Winkler model is used for the lateral response. Their analysis proved that they can fairly accurately represent the interaction between the soil and the caisson with translational and rotational springs on the sides and two additional springs of each type at the bottom (see Fig. 1.5).

In the end, even though they are all validated with complex FE models, there are some discrepancies between the formulations and they present tailored solutions to the specific problem configurations. Moreover, there are several models for different foundations (i.e. piles, caissons, slabs), and even different types of soil, as the latter's characteristics also play an important role. For instance, Pang et al. developed a model to predict SSI for piles which considers a two-surface plasticity model by accounting for the softening or hardening of sand [39]. Conto et al. developed an approach that investigates piles subjected to ground-borne vibrations allowing rotational motion and reaction torques using the singular boundary method [5]. Wang et al. proposed a method that uses a 2-spring approach to predict the lateral behaviour of a pile in soft clay and considers the effect of the length-to-diameter ratio (see Fig. 1.7) [58].

Many different methods and approximations have been developed to represent the interaction between soil and structure. Some are more complicated than others, and most are specific to particular cases. Some even analyse the problems in the time domain instead of the frequency domain in order to capture the effect of transient loads [32] [33]. The following sections explain and explore two benchmark cases: "Disc over homogeneous halfspace" and "Pile in soil". These are clear examples of SSI problems for which fairly accurate solutions have been developed. The objective is to use them as reference problems to examine if a generic approach is feasible.

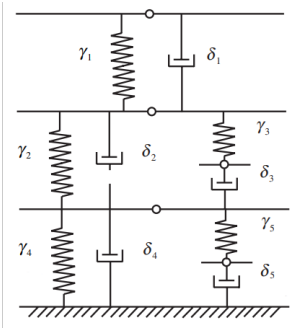


Figure 1.4: Systematic nested lumped-parameter models [62].

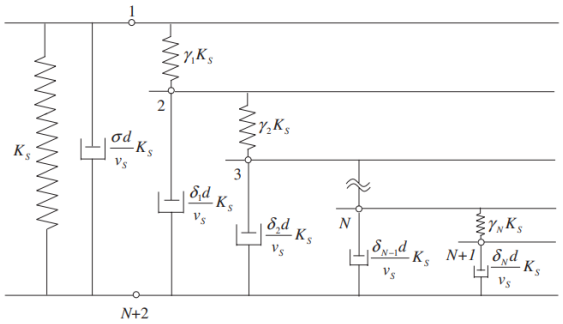


Figure 1.6: Nested lumped-parameter models [61].

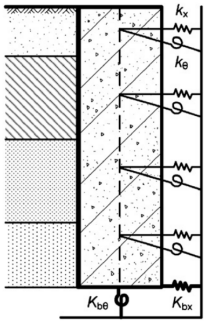


Figure 1.5: Simplified model for lateral response of caissons [56].

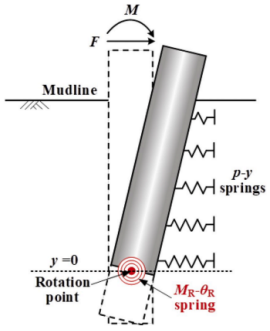


Figure 1.7: Two-spring approach for piles in clay [58].

1.3. Objectives and scope

The research gap addressed in this investigation is the lack of homogeneity between different approaches and the extensive simplifications assumed for the conventional models. The goal is to simplify the process by developing a model applicable to various foundation types. This approach utilises key characteristics of the soil and foundation, such as density, shear modulus and Poisson's ratio of the former and dimensions of the latter. Additionally, the model aims to establish a process that outputs a set of reduced mass, stiffness, and damping matrices to be incorporated in the SSI analysis of a structure.

This research's objective is to develop a new method that describes the mass (M), damping (C) and stiffness (K) matrices of a system using a generic computational approach. These matrices will then be used in the solution of the structure to account for the effects of SSI on the overall behaviour. The focus is on developing a general approach for circular plates over a homogeneous halfspace and piles embedded in soil subjected to vertical motion. These SSI cases are the first to be obtained with the approach and are benchmark problems. Eventually, the method could be applied to other SSI problems, especially the ones for which mechanical analogues of sufficient accuracy do not exist.

The goal of this research is to answer the following research questions and sub-questions:

1. *Research question: What are the essential SSI parameters and necessary tools required to establish a consistent modelling framework for generating a suitable dataset for a computational approach for SSI cases?*
    - *What are the main properties of the soil that must be considered when modelling the soil-structure interaction?*
    - *What are the main properties of the substructure that must be considered when modelling the soil-structure interaction?*

2. *Research question: What constitutes an efficient data-driven process for translating large soil-foundation datasets into simplified mechanical analogues that provide a reduced and accurate representation of the system in the frequency domain?*
    - *How can a data-driven approach be incorporated to derive an SSI reduced model?*
    - *How should a generic mechanical analogue be formulated to be applicable in different SSI problems?*
    - *What common characteristics from an SSI system should be considered to obtain a computational approach applicable for different types of foundations?*

Answering these questions will potentially yield a new method that solves SSI cases without extensive simplifications. The scope is limited to obtaining the surrogate models for two foundation scenarios. The first is "Disc over homogeneous halfspace" under vertical, horizontal and rocking motions. In this case, the solution is uncoupled. The second scenario is "Pile in soil" under vertical vibrations. This surrogate model is based on a new computational approach to be presented in the following chapters. The variables of the proposed approach are illustrated in Figure 1.8 as a flowchart depicting the process and elements involved in each step.

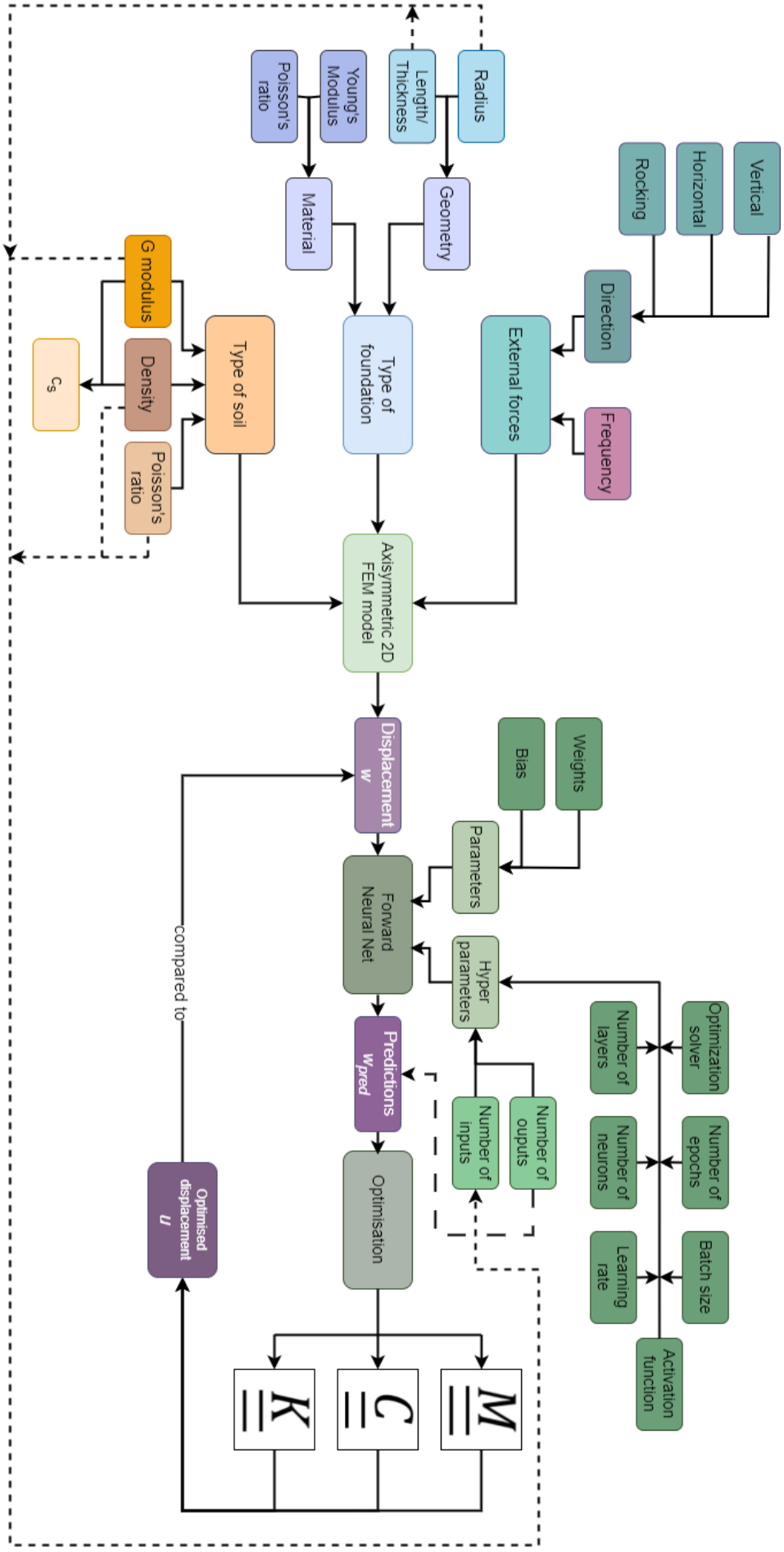


Figure 1.8: Flowchart presenting independent and dependent variables.

1.4. Thesis outline

The content of this document provides the most important information about the previous research, the application of several models and approaches, the steps taken to arrive at the outcome and the results. It is divided into four chapters.

Chapter 1 briefly summarises some relevant points of the early investigations in soil-structure interaction. This helps to understand the problem at hand and its importance. It also explains the methods used in soil-structure interaction when tackling various scenarios. Additionally, it introduces the existing analogues in this field. Finally, it states the objective and scope of the work and schematically shows the used variables.

Subsequently, Chapter 2 examines the existing models utilised in practice for soil-structure interaction (SSI). It starts by introducing the dynamic stiffness matrix and the impedance functions. It then dives into a more detailed explanation of some available analogues for "Disc over homogeneous halfspace", like Lysmer's, Pais-Kausel's, and two of Wolf's models. It then explains Novak's model for piles under vertical vibrations.

After understanding the use of mechanical analogues in Chapter 2, Chapter 3 dives into Machine Learning, focusing on specific aspects and details related to the scope of this work. It evaluates the possibility of using this approach for SSI problems and how that can be achieved. The variables affecting Feedforward Neural Networks, like the number of layers or neurons, and the theory behind the tuning of these models are outlined. Further computational approaches and the application to dynamic problems are presented as building-up steps towards a functioning Neural Network for SSI problems.

Finally, the chosen Neural Network for SSI is shown in Chapter 4. It begins by explaining the approach for a general SSI scenario, the influencing factors and other relevant considerations. Then, it describes how the input dataset was obtained. In the end, it illustrates the intermediate results and the ultimate working model, which yields the final goal of this work.

The document finishes with the Conclusions in Chapter 5, where the process of this work and the results are wrapped up and revised one last time. Some recommendations for future works are given and areas for potential improvements and further research are suggested.

2

Mechanical analogues for Soil-Structure Interaction

2.1. Definition of Dynamic Stiffness Matrix

The dynamic stiffness matrix ($\tilde{\boldsymbol{K}}_{dyn}$), as introduced in Chapter 1, serves as a connection between the soil's behaviour and its effect on the structural response. It integrates the interaction between the foundation and the soil into the structural analysis, thereby accounting for soil-structure interaction, by substituting the soil domain. A *rigid foundation block* will be used to explain the construction of the $\tilde{\boldsymbol{K}}_{dyn}$ and a schematisation of the problem is shown in Figure 2.1. The method can then be extrapolated to more complicated foundations [55].

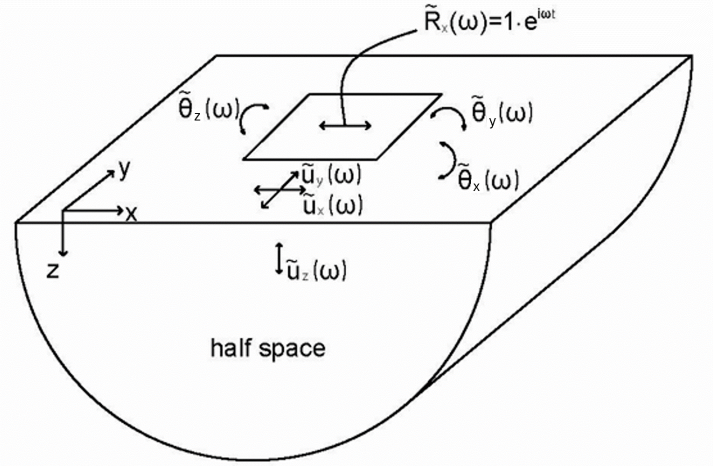


Figure 2.1: General scheme for the derivation of the Dynamic Stiffness Matrix [55].

The derivation of the $\tilde{\boldsymbol{K}}_{dyn}$ is based on the force balance of the foundation (see Eq. 2.1) and the reaction force (see Eq. 2.2):

$$m\ddot{u}_z(t) + R_z(t) = P_z(t) \tag{2.1}$$

$$R_z(t) = \int_0^L \int_0^B \sigma_z(x, y, t)\mathrm{d}x\mathrm{d}y \tag{2.2}$$

where the first one considers the mass (m) of the foundation times acceleration $\ddot{u}$ and the reaction of the soil ($R_z(t)$) when a time-dependent force ($P_z(t)$) is applied to the mass. The second one defines the reaction of the soil ($R_z(t)$) as the integral of the stresses in the soil ($\sigma_z(x, y, t)$) under the area of the foundation (of dimensions L, B).

The Fourier Transform (see Eq.: 2.3) is applied for the transition from the time domain to the frequency domain. The partial differential equations are hereby transformed into algebraic equations which can be solved directly. The time response can be computed if needed by applying the Inverse Fourier Transform (see Eq. 2.4).

$$\tilde{f}(\omega) = \int_{-\infty}^{\infty} f(t) \cdot e^{-i\omega t} dt \tag{2.3}$$

$$f(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \tilde{f}(\omega) \cdot e^{i\omega t} d\omega \tag{2.4}$$

where the tilde symbol in $\tilde{f}(\omega)$ indicates that the quantity is in the frequency domain, and $f(t)$ is in the time domain.

After transforming both equations (2.1, 2.2) the reaction in the frequency domain ($\tilde{R}(\omega)$) can be substituted into Equation 2.1 to obtain:

$$(-m\omega^2 + \tilde{k}_z(\omega))\tilde{u}_z(\omega) = \tilde{P}_z(\omega) \tag{2.5}$$

where, once again, m is the mass, ω is the frequency, $\tilde{k}(\omega)$ is the stiffness, $\tilde{u}(\omega)$ the displacement and $\tilde{P}_z(\omega)$ the load, all in the frequency domain.

The reaction of the soil is given by the stiffness factor $\tilde{k}_z$ in Equation 2.5, which is equivalent to that of frequency-dependent spring and dashpot, i.e. for the real and imaginary parts, respectively. With this equation, it is possible to determine the flexibility $\hat{f}$ (the inverse of the stiffness) of the soil if a unit dynamic force is applied on the foundation and the displacement is computed (see Eq. 2.6) [55].

$$\tilde{u}_z(\omega)) = \frac{\tilde{P}_z(\omega)}{\tilde{K}_z(\omega)} \rightarrow \tilde{f}_z(\omega) = \frac{1}{\tilde{K}_z(\omega)} \tag{2.6}$$

where $\tilde{P}_z(\omega) = 1$ is the unit dynamic force, $\tilde{u}_z(\omega)$ is the displacement, and $\tilde{K}_z(\omega) = -m\omega^2 + \tilde{k}_z(\omega)$ is the dynamic stiffness.

The application of the vertical force rigorously provides a reaction in every degree of freedom in a generic system where coupling can be present. Consequently, the application of a vertical force provides the first column of the dynamic flexibility matrix $\tilde{\boldsymbol{F}}$ (inverse of dynamic stiffness matrix $\tilde{\boldsymbol{K}}_{\boldsymbol{dyn}}$), meaning the response in every direction due to said force (see Eq. 2.7).

$$\hat{f}_z(\omega) = \left[f_{z,z}; f_{z,y}; f_{z,x}, f_{z,\theta_x}; f_{z,\theta_y}; f_{z,\theta_z} \right]^T \tag{2.7}$$

where $\hat{f}$ corresponds to the flexibility in the frequency domain and the components of the vector are the flexibility in every DoF due to the applied force. The same process is then performed with a unitary force in each DoF, as shown in Figure 2.1, to obtain the complete flexibility matrix. The matrix of this problem is of size 6x6, where the columns correspond to the response to a particular force and the rows give the response to a particular degree of freedom. Ultimately, the elements of the matrix will depend on the type, shape and characteristics of the foundation, its level of embedment and the characteristics of the soil.

By knowing the flexibility matrix of the system, the stiffness of the foundation for every degree of freedom and the entire system can be obtained. To do so, the flexibility matrix can be inverted (see Eq 2.8).

$$\tilde{\boldsymbol{K}}_{\boldsymbol{dyn}}(\omega) = \frac{1}{\tilde{\boldsymbol{F}}(\omega)} \tag{2.8}$$

Consequently, the dynamic stiffness matrix $\tilde{\boldsymbol{K}}_{dyn}(\omega)$ can be obtained. This will have the same dimensions as the flexibility matrix and each column corresponds to the reaction force given by and imposed unitary displacement. Moreover, $\tilde{\boldsymbol{K}}_{dyn}$ might be complex in the presence of damping of the soil (see Eq. 2.9).

$$\tilde{\boldsymbol{K}}_{dyn}(\omega) = \Re\{\tilde{\boldsymbol{K}}(\omega)\} + i\Im\{\tilde{\boldsymbol{K}}(\omega)\} \tag{2.9}$$

2.2. Mechanical analogues for Soil-Structure Interaction for "Disc over homogeneous halfspace"

Soil-structure interaction problems present large variability due to the various foundation, soil and superstructure types. The shape of the foundation itself, the type of soil, whether it is layered or not, and the assumptions and simplifications affect the problem statement and the approach to solving it. One of the study cases for which an accurate solution with a simple approach has been found is the *disc over a homogeneous halfspace*. The problem at hand consists of a circular rigid plate on top of a homogeneous soil. Firstly, the plate's dimensions do not allow for flexible behaviour but rather maintain a ratio that ensures rigid behaviour. Secondly, the homogeneous soil is taken into account as a halfspace domain, with a radiation boundary condition, where the waves are not reflected back towards the structure but rather radiate to infinity. Figure 2.2 shows the disc on a homogeneous halfspace.

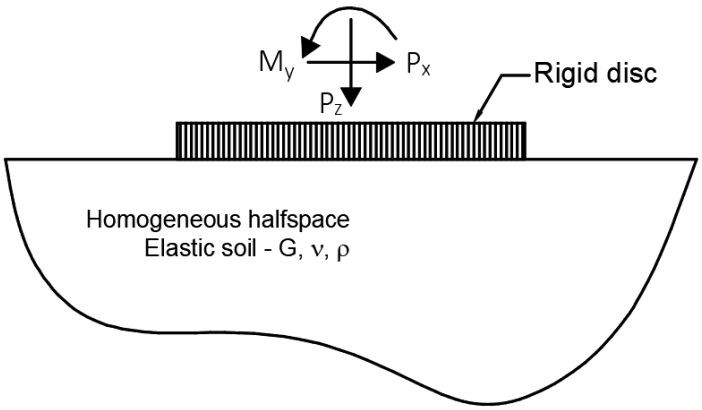


Figure 2.2: Disc on homogeneous halfspace ([15]).

The main soil parameters to be considered in this problem are the shear modulus (G), Poisson's ratio (ν) and mass density (ρ) of the soil. These three allow to calculate further characteristics of the soil, such as the shear wave velocity (c_s). Additionally, the principal parameter of the foundation is the radius of the circular plate (r_0). The objective is to find the disc's displacement in the vertical, horizontal and rocking directions when a dynamic load (P_z, P_x and M_y respectively) is applied. In this work, the torsional response, corresponding to a motion around the vertical axis will be disregarded. Additionally, the moment around the x-axis and the horizontal translation in the perpendicular axis are not explicitly analysed because they are already considered in M_y and P_x due to the system's symmetry condition. Ultimately, the nature of the foundation system, not being embedded in the soil, permits assuming the responses completely uncoupled.

Some general assumptions made for this problem are that the soil is linear, isotropic and homogeneous. These are idealisations of the soil. Firstly, the soil can present non-linear material behaviour in some cases, but this will be outside the scope of the present study. Secondly, the soil medium is in reality layered and each layer has different materials and properties. This means that homogeneity and isotropy are unlikely. Finally, the representation of the soil as a halfspace allows the limitation of the infinite soil into a finite domain with wave radiation at the boundaries. Other assumptions are that only normal stresses are transferred between the foundation and the soil [24] and that the foundation keeps its shape while vibrating because it is much stiffer than the soil [38]. Finally, the disc and the soil are in full contact "continuously, without the formation of a gap or uplift.

The mechanical analogues presented in this work are mainly based on SDoF systems solved in the frequency domain. The equation of motion in the time domain is transformed to the frequency domain by applying the Fourier Transform, such as the one shown in Equation 2.3. The equation of motion in the time domain is similar to the one presented in Equation 2.10.

$$w(\xi,t) = m\ \frac{\partial^2 w(\xi,t)}{\partial t^2} + k\ w(\xi,t) + c\ \frac{\partial w(\xi,t)}{\partial t} = P_{tot}(\xi,t) \tag{2.10}$$

where w is the displacement dependent on time and space, m is the system's mass and k and c are the stiffness and damping coefficients. Finally, P_{tot} is the force exciting the system and also depends on space and time.

Different solutions with various levels of accuracy have been studied and developed. This SSI problem has been investigated since the late 1800's by scientists like Boussinesq and later Cerruti [19]. After that, Lysmer was one of the scientists to study the effects of dynamic loads on a circular plate in the 1970's. He assumed an SDoF system where the mass is the mass of the footing and the radiation conditions of the soil are represented by a spring and a dashpot. This mechanical analogue posed a significantly accurate response to the problem at hand and was studied for this work.

After Lysmer, other researchers have searched for a better, more accurate solution to solve this problem. Pais's and Kausel's [38] presented a solution that proposed a dynamic impedance to approximate the dynamic stiffness, which is outlined in further sections. The last work to be analysed in this work are the models by Wolf [60] [59]. The first model, called Standard Lumped-Parameter model, is an SDoF with a mass, a spring and a dashpot, where the mass and the dashpot coefficients depend on the stiffness of the spring. The second one is known as the Fundamental Lumped-Parameter Model, or the Monkey-tail model and adds a second degree of freedom attached to the top to approximate the foundation's response better. A thorough explanation of these mechanical analogues will be discussed in the following sections.

The availability of multiple solutions to this problem makes it an excellent benchmark for comparative analysis. The models mentioned above will also be compared to a Finite Element model to verify their applicability and accuracy across different frequencies and configurations. This comparison serves to demonstrate the accuracy of the models and ensure the correct functioning of the FE model recursively. Ultimately, the generic approach will also be compared with this benchmark problem, verifying the new method and understanding its suitability.

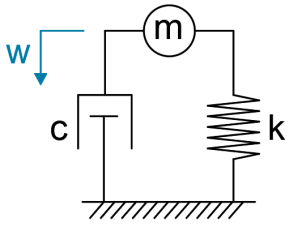
In this thesis, torsional motion will be disregarded. Additionally, due to the symmetric nature of the problem, only three degrees of freedom will be taken into account in the derivation of the dynamic stiffness matrix and other properties. For a circular slab, the translational displacement in either x or y and the rocking motion around these axes are equivalent ($\underline{u}_x(\omega) = \underline{u}_y(\omega)$ and $\underline{\theta}_x(\omega) = \underline{\theta}_y(\omega)$). Consequently, it will only be considered once.

2.2.1. Lysmer's problem

As mentioned earlier, one of the first scientists to provide a solution for the problem at hand was Lysmer. In his dissertation, Lysmer does not restrict the solution to low frequencies, like the ones given by machinery, but makes it applicable across all frequencies. It resolves the issue under steady-state conditions and subsequently utilises the solution to scrutinise a transient load. The model he advocates comprises a single-degree-of-freedom (SDoF) system incorporating mass, stiffness, and dashpot elements, where the dashpot facilitates energy dissipation of the waves generated by the dynamic load. Additionally, the mass, stiffness and damping components are independent of this dynamic load.

To arrive at the solution, Lysmer first analyses a general dynamic system to which a vertical dynamic load $P = P_0 e^{i\omega t}$ is applied. The vertical displacement δ is calculated and then translates this system into a massless system with a spring and a dashpot that is excited with the same vertical dynamic load. Subsequently, he examines a general dynamic system like the first one but adds a mass. In this case, the mass is exited by a force $Q = Q_0 e^{i\omega t}$ and it will generate a response in the system equal to P. By matching the accelerations and differentiating he obtains the displacement of the system with a mass. This analysis revealed that the steady state solution of the system pictured in Figure 2.3 can be found by finding the massless system's displacement.

After analysing the implications of low and high frequencies, the pressure distribution and the displacement of the free surface, he provides a simplified analogue as the one presented in Figure 2.3. The values for the spring's stiffness k and the dashpot's viscosity c are given by the relations in Equations 2.12 and 2.13 respectively ([24], pg. 43). Considering the mass of the footing, the displacement can then be calculated as in Equation 2.14.



$$m = \text{mass of footing} \tag{2.11}$$

$$k = \frac{4Gr_0}{1-\nu} \tag{2.12}$$

$$c = \frac{3.4}{1-\nu} r_0^2 \sqrt{Gr_0} = \frac{0.85\ k\ r_0}{c_s} \tag{2.13}$$

$$w(\omega) = \left[-\omega^2 m + k + i\omega c\right]^{-1} \cdot \tilde{p} \tag{2.14}$$

Figure 2.3: Lysmer model.

The stiffness of the linear elastic spring relates to the characteristics of the soil stiffness, its Poisson's ratio and the radius of the foundation. Similarly, the viscous coefficient considers the shear wave velocity next to the previously mentioned parameters.

The conclusions of his work rely on the demonstration that this analogue is preferred over a complete halfspace model due to its mathematical and computational simplification, where the errors are practically negligible. It is noted that it applies "only as long as the displacements can be considered small compared to the size of the footing" [24] and it invited future researchers to look into other motion modes like horizontal, rocking and torsional.

2.2.2. Pais's and Kausel's model

Pais and Kausel provided their solution for this specific SSI problem. To do so, they used a mixed boundary-value problem and solved the embedded case taking the radiation condition of the soil into account. They also considered different foundation shapes, although only circular discs will be discussed in this work.

This model accounts for the possibility of an embedded foundation (see Fig. 2.4) and thus implies a variation in the equations. They consider the same static stiffness principle as the one proposed by Boussinesq initially and used by Lysmer as explained in Section 2.2.1 [38].

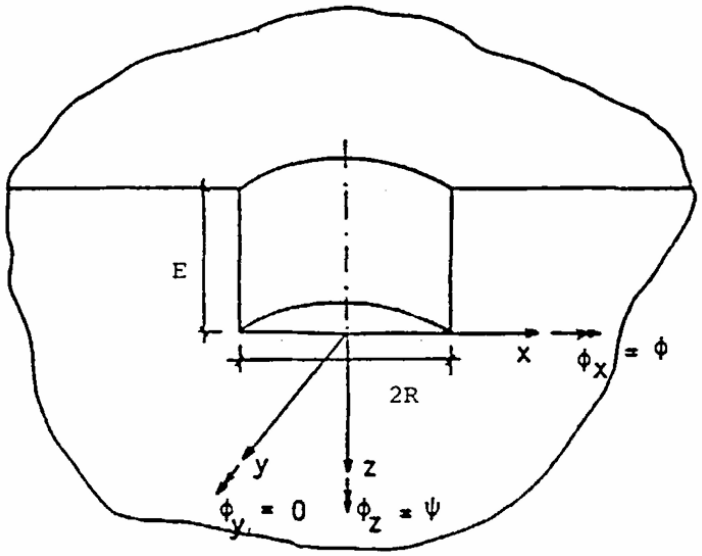


Figure 2.4: Cylindrical embedded foundation [38].

Firstly, as opposed to Lysmer's model, in the derivation of this model, the foundation is considered massless. Additionally, the static stiffness includes the degree of embedment E/R. It is solved for a dimensionless frequency a_0 and considers, for each type of motion n, a static stiffness K_n multiplied by a frequency-dependent dynamic stiffness factor $k_n(\omega)$ and a frequency-dependent damping factor $c_n(\omega)$. The dynamic impedance is calculated as in Equation 2.15 where ξ is the soil damping and includes the effect of a dimensionless frequency $a_0 = \omega r_0/c_s$.

$$Z = K_n(k_n + ia_0c_n)(1 + 2i\xi) \tag{2.15}$$

The equations are all provided in Pais and Kausel's equations below, where $\alpha = \sqrt{(2(1-\nu))/(1-2\nu)}$. They mention that "although k in the vertical and rocking modes depends strongly on the value of ν, for simplicity its influence is not taken into account herein" [38]. This is because, for Poisson's ratios higher than 0.4 and for higher frequencies, the response given by these formulas might not be accurate.

Pais and Kausel's equations [7]

Vertical **Horizontal**

$$K_v = \frac{4Gr_0}{1-\nu}\left(1 + 0.54\frac{E_m}{r_0}\right) \quad (2.16) \qquad K_h = \frac{8Gr_0}{2-\nu}\left(1 + \frac{E_m}{r_0}\right) \quad (2.19)$$

$$k_v = 1 \quad (2.17) \qquad k_h = 1 \quad (2.20)$$

$$c_v = \frac{\pi}{K_v/Gr_0}\left[\alpha + 2\frac{E_m}{r_0}\right] \quad (2.18) \qquad c_h = \frac{\pi}{K_h/Gr_0}\left[1 + (1+\alpha)\frac{E_m}{r_0}\right] \quad (2.21)$$

Rocking

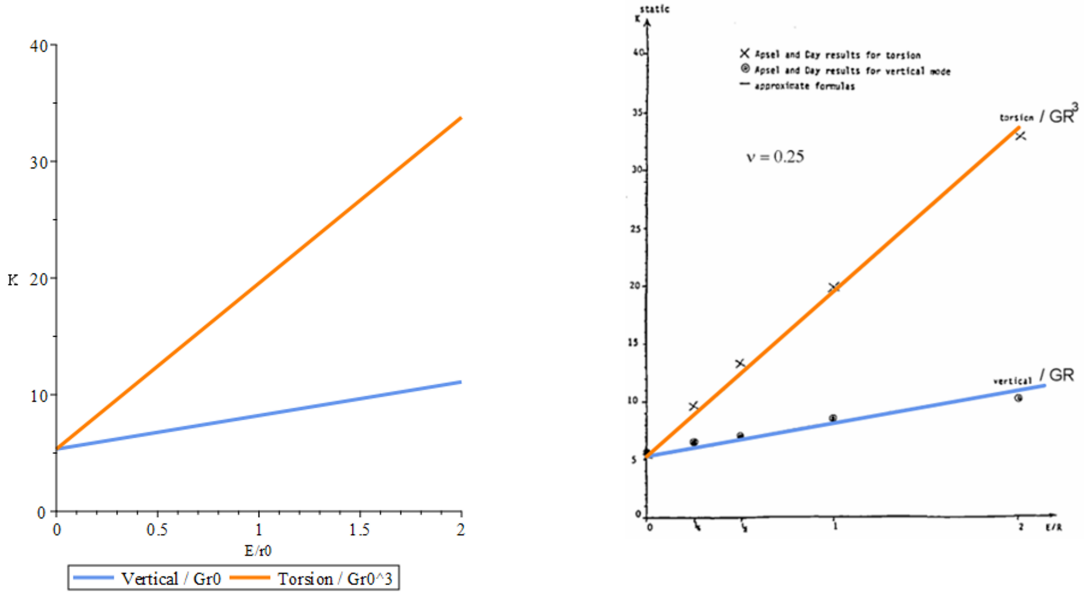
$$K_r = \frac{8Gr_0^3}{3(1-\nu)}\left(1 + 2.3\frac{E_m}{r_0} + 0.58\left(\frac{E_m}{r_0}\right)^3\right) \quad (2.22)$$

$$k_r = 1 - \frac{0.35a_0^2}{1+a_0^2} \quad (2.23)$$

$$b_r = \frac{2}{1 + E/r_0} \quad (2.24)$$

$$c_r = \frac{\pi}{K_v/Gr_0^3}\left[\frac{1}{4}\alpha + \frac{E_m}{r_0} + \frac{1}{3}(1+\alpha)\left(\frac{E_m}{r_0}\right)^3\right] \cdot \frac{a_0^2}{b_r + a_0^2} + 0.84(1+\alpha)\left(\frac{E_m}{r_0}\right)^{2.5}\frac{b_r}{b_r + a_0^2} \quad (2.25)$$

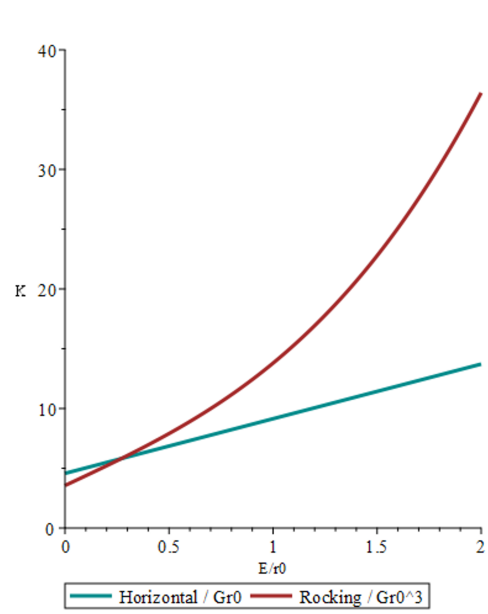
These equations give the components of the mechanical analogue of the circular disc if the formulation for the impedance is used as mentioned before. The static stiffness coefficients (K) for all types of motions are plotted and compared with the ones from the literature for validation (see Figs. 2.5b and 2.6b). In Figure 2.5, the torsional static coefficient is also included, although it is exclusively for visual purposes as it is not studied in the work and is thus marked in a dotted line.



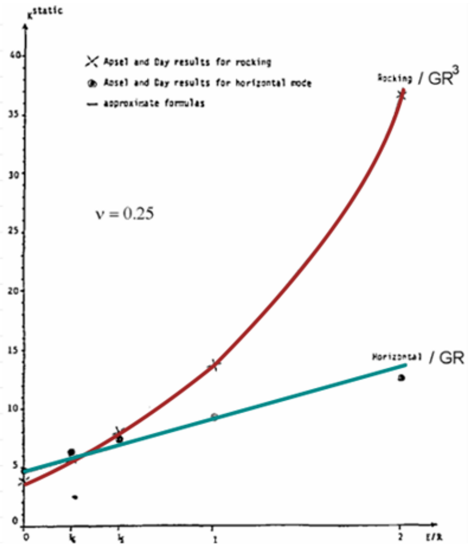
(a) Static stiffness coefficients for vertical (K_v) and torsional (K_t) motions.

(b) Static stiffness coefficients for vertical (K_v) and torsional (K_t) motions from the literature [38].

Figure 2.5: Comparison of static stiffness coefficient K_v and K_t – Calculated vs. graphs from the original publication.



(a) Static stiffness coefficients for horizontal (K_h) and rocking (K_r) motions.



(b) Static stiffness coefficients for horizontal (K_h) and rocking (K_r) motions from the literature [38].

Figure 2.6: Comparison of static stiffness coefficient K_h and K_r – Calculated vs. graphs from the original publication.

The previous graphs show that the vertical (Fig 2.5) and horizontal (Fig. 2.6) static stiffnesses have a linear trend over the embedment relation (E/r_0). The former starts at around five for the embedment coefficient equal to zero for both the calculated one and the literature reference. The latter presents a similar behaviour, it also starts at around 5 and increases with embedment. The rocking static stiffness coefficient (Fig 2.6) has a cubic behaviour in both plots, as expected due to its cubic terms. It can be seen that the studied model complies with the results from the literature.

The correspondence between the plots obtained with the use of the equations and the ones presented in the literature is a reliable indicator that the model has been implemented successfully. It serves as a reference point for further application of the models.

2.2.3. Wolf's Standard lumped-parameter model

As proposed by Lysmer, a simple way to represent the soil-structure interaction that is not time-consuming but can grasp its main characteristics is a standard lumped-parameter model, meaning an SDoF system. This model represents the unbounded soil as a mass-spring-dashpot system with frequency-independent parameters. It can be solved without using very sophisticated software or computational code. The simplification of the soil is represented in Figure 2.7 [60]. Figure 2.7 a) represents the model to solve for the translational motions, where the degree of freedom is the vertical or horizontal displacement and the stiffness and viscosity are linear. On the other hand, Figure 2.7 b) represents the models with rotational degrees of freedom ϑ and can be used for rocking and torsional motions.

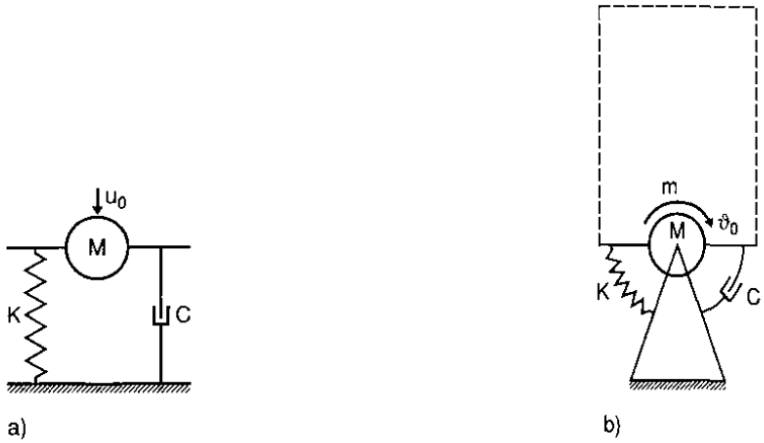


Figure 2.7: Standard Lumped-Parameter Model for a) Translation and b) Rotation ([60], pg. 120).

The stiffness of the spring (K) is the direct static stiffness of the soil. Contrarily, the damping (C) and mass (M) coefficients depend on the system's dynamic response in the low- and medium-frequency range. They are calculated using the following equations, where r_0 is the radius of the foundation, c_s is the shear wave velocity of the soil, K is the stiffness and γ and μ are dimensionless coefficients calculated according to relations defined for different types of foundation.

$$C = \frac{r_0}{c_s}\gamma K \tag{2.26}$$

$$M = \frac{r_0^2}{c_s^2}\mu K \tag{2.27}$$

As can be seen, the mass and damping coefficients depend on the system's stiffness, given by the characteristics of the soil.

TABLE 2-9 STATIC STIFFNESS AND DIMENSIONLESS COEFFICIENTS OF STANDARD LUMPED-PARAMETER MODEL FOR DISK WITH MASS ON HOMOGENEOUS HALFSPACE

		Dimensionless Coefficients of	
	Static Stiffness K	Dashpot γ	Mass μ
Horizontal	$\frac{8Gr_0}{2-\nu}$	0.58	0.095
Vertical	$\frac{4Gr_0}{1-\nu}$	0.85	0.27
Rocking	$\frac{8Gr_0^3}{3(1-\nu)}$	$\frac{0.3}{1+\frac{3(1-\nu)m}{8r_0^5\rho}}$	0.24
Torsional	$\frac{16Gr_0^3}{3}$	$\frac{0.433}{1+\frac{2m}{r_0^5\rho}\sqrt{\frac{m}{r_0^5\rho}}}$	0.045

Figure 2.8: Table for K, γ and μ parameters ([60], pg. 121).

The added mass in the model is used to better fit the response of the soil with that of the foundation, especially at higher frequencies, and does not directly represent the mass of the soil or the disc. Contrary to Lysmer's and similarly to Pais-Kausel's models, this one does not include the mass of the foundation. For the case of a disc on a homogeneous halfspace, the parameters γ and μ are calculated according to Table 2-9 from [60] (see Fig. 2.8) for each type of motion (horizontal, vertical, rocking and torsional, although the latter is disregarded in this work).

These parameters allow to calculate the displacement of the foundation in the soil when a load is applied considering an SDoF system. It can be done with an equation of motion for each type of motion n (horizontal, vertical and rocking) as in Equation 2.28. This equation is in the frequency domain and gives the displacement u_0 for each type of motion n and a particular frequency of analysis ω. M_n, K_n and C_n are the mass, stiffness and damping coefficients for each case and $\tilde{P}$ is the external force applied to the system which will be considered unitary. From this equation of motion, the dynamic stiffness can be extracted and results as in Equation 2.29:

$$\tilde{u}_{0_n}(\omega) = [-\omega^2 M_n + K_n + i\omega C_n]^{-1} \tilde{P}_{0n}(\omega) \tag{2.28}$$

$$\tilde{K} = [-\omega^2 M_n + K_n + i\omega C_n] \tag{2.29}$$

The dynamic stiffness can also be expressed with dimensionless parameters. In Equation 2.30, $k(a_0)$ and $c(a_0)$ are the dimensionless spring and dashpot coefficients respectively (see Eqs. 2.31 and 2.32), and $a_0 = \omega r_0/c_s$ is the dimensionless frequency.

$$S(a_0) = K[k(a_0) + ia_0 c(a_0)] \tag{2.30}$$

$$k(a_0) = 1 - \mu a_0^2 \tag{2.31}$$

$$c(a_0) = \gamma \tag{2.32}$$

The coefficients are plotted for validation in Figure 2.9. It can be seen that $k(a_0)$ is parabola-shaped, while $c(a_0)$ is constant, as expected. This indicates that the behaviour of the stiffness

is frequency-dependent. As the latter increases, the stiffness of the spring decreases until it reaches zero at approximately $a_0 \simeq 2$. After this point, the real part of the dynamic stiffness is negative. However, the absolute value keeps increasing due to the participation of the resting two terms. The mass, being multiplied by the square of the frequency (ω^2), has a constantly increasing participation in the absolute value. The dashpot, multiplied by the frequency (ω) and the imaginary term (i) increases the imaginary part, hence the absolute value, with higher frequencies.

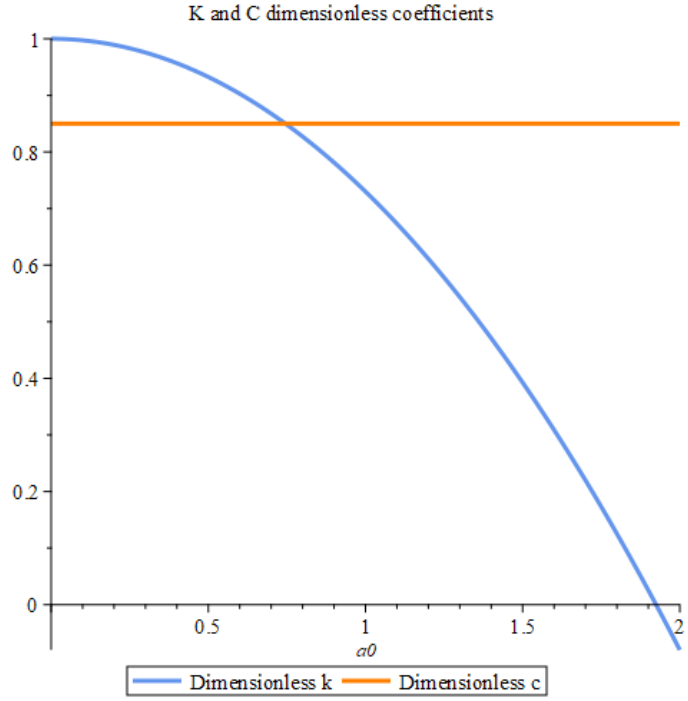


Figure 2.9: $k(a_0)$ and $c(a_0)$ dimensionless coefficients for the Standard lumped-parameter model.

2.2.4. Wolf's Fundamental lumped-parameter model

This model, like the previous ones, considers a simple system to represent the soil-structure interaction of the problem at hand. It proposes two approaches for the additional degree of freedom. The first one attaches an extra spring and dashpot in series to the left of the central mass. The second one, also known as the *monkey-tail* model, examines the model with a mass connected to a dashpot. In this case, the monkey-tail model will be studied (see Fig. 2.10).

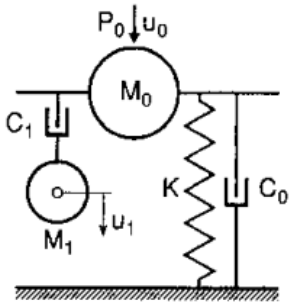


Figure 2.10: Fundamental Lumped-Parameter Model - Monkey-tail ([60], pg. 122).

This system is solved similarly to the previous one. However, the coefficients and the equation of motion vary. The latter is shown in Equation 2.33, which is the solution of the response and it is formulated to obtain more accurate results than the Standard Lumped-Parameter model above. In this case, the displacement u_0 is calculated for each type of motion n, where again $\tilde{P}_{0n}$ is the applied force and K_n is the static stiffness of the soil for each kind of motion. Additionally, M_0 is the central mass, C_0 is the damping in parallel to the stiffness and M_1 and C_1 are the mass and damping coefficient of the extra degree of freedom.

$$\tilde{u}_{0_n}(\omega) = \left[K \left[1 - \frac{\frac{\omega^2 M_1}{K}}{1 + \frac{\omega^2 M_1^2}{C_1^2}} - \frac{\omega^2 M_0}{K} + i\omega \left(\frac{M_1}{C_1} \frac{\frac{\omega^2 M_1}{K}}{1 + \frac{\omega^2 M_1^2}{C_1^2}} + \frac{C_0}{K} \right) \right] \right]^{-1} \cdot \tilde{P}_{0n}(\omega) \tag{2.33}$$

The mass and dashpot coefficients are calculated similarly to the previous model (see Eqs. 2.34, 2.35, 2.36 and 2.37). In the same way, the dimensionless coefficients γ_0, γ_1, μ_0, μ_1 are calculated using the equations shown in Table 2-10 from [60] shown in Figure 2.11.

$$C_0 = \frac{r_0}{c_s} \gamma_0 K \tag{2.34}$$

$$M_0 = \frac{r_0^2}{c_s^2} \mu_0 K \tag{2.35}$$

$$C_1 = \frac{r_0}{c_s} \gamma_1 K \tag{2.36}$$

$$M_1 = \frac{r_0^2}{c_s^2} \mu_1 K \tag{2.37}$$

In the table, it can be seen that the static stiffness remains the same for each kind of motion. So far, all the mechanical analogues use the same static stiffness, except for the embedment coefficient in Pais-Kausel's model. Additionally, in this case, the dashpots and masses dimensionless coefficients depend mainly on Poisson's ratio.

TABLE 2-10 STATIC STIFFNESS AND DIMENSIONLESS COEFFICIENTS OF FUNDAMENTAL LUMPED-PARAMETER MODEL (MONKEY-TAIL ARRANGEMENT) FOR DISK ON HOMOGENEOUS HALFSPACE

	Static Stiffness K	Dimensionless Coefficients of Dashpots γ_0	Dashpots γ_1	Masses	μ_0	Masses μ_1
Horizontal	$\frac{8Gr_0}{2-\nu}$	$0.78 - 0.4\,\nu$	——		——	——
Vertical	$\frac{4Gr_0}{1-\nu}$	0.8	$0.34 - 4.3\,\nu^4$	$\nu < \frac{1}{3}$ $\nu > \frac{1}{3}$	0 $0.9\left(\nu - \frac{1}{3}\right)$	$0.4 - 4\,\nu^4$
Rocking	$\frac{8Gr_0^3}{3(1-\nu)}$	——	$0.42 - 0.3\,\nu^2$	$\nu < \frac{1}{3}$ $\nu > \frac{1}{3}$	0 $0.16\left(\nu - \frac{1}{3}\right)$	$0.34 - 0.2\,\nu^2$
Torsional	$\frac{16Gr_0^3}{3}$	——	0.29		——	0.2
		(0.017)	(0.291)		——	(0.171)

Figure 2.11: Table for K, γ and μ parameters ([60], pg. 123).

As for the previous model, the dynamic stiffness matrix can be expressed with dimensionless stiffness (Eq. 2.38) and damping (Eq. 2.39) coefficients, using the same expression as for the previous model (Eq. 2.30). In this case, the dimensionless stiffness and damping are obtained considering the dimensionless coefficients of both degrees of freedom in the model.

$$k(a_0) = 1 - \frac{\mu_1 a_0^2}{1 + \frac{\mu_1^2}{\gamma_1^2} a_0^2} - \mu_0 a_0^2 \tag{2.38}$$

$$c(a_0) = \frac{\mu_1}{\gamma_1} - \frac{\mu_1 a_0^2}{1 + \frac{\mu_1^2}{\gamma_1^2} a_0^2} + \gamma_0 \tag{2.39}$$

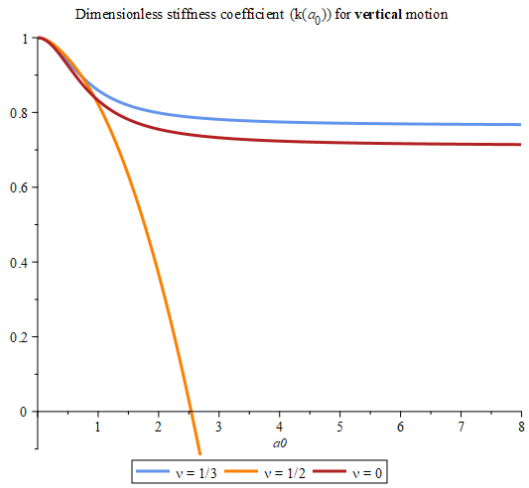
The coefficients are plotted and compared to those in the literature for validation. Figures 2.12, 2.13, 2.14 and 2.15 show, on the left, the dimensionless stiffness and damping coefficients calculated using the equations and, on the right, the one from the literature. It can be observed that, for all Poisson's ratios considered, the results are in perfect agreement, indicating that the equations are applied correctly.

For the dimensionless vertical stiffness coefficient (see Fig. 2.12) it can be observed that for $\nu = 0.5$, the stiffness follows a parabolic shape behaviour, where it goes in the negative domain after approximately $a_0 \simeq 2.5$. This means that with increasing frequency the stiffness term of the system decreases and even gives negative values. This shows that for incompressible soil (a valid approximation for saturated soils), the stiffness coefficient becomes negative. However, for a Poisson ratio $\nu = 1/3$, the response becomes quasi-asymptotic with increasing frequency, as well as for $\nu = 0$. This indicates that with lower values of ν the vertical stiffness is not as

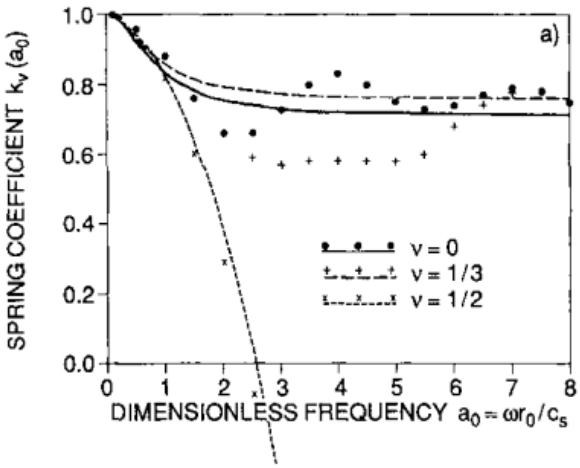
affected by the increasing frequency and, contrarily, shows an almost frequency-independent behaviour.

Similarly, Figure 2.13 shows the calculated damping coefficient and the graphs from the literature. For all Poisson's ratios, the damping becomes asymptotic after approximately $a_0 \simeq 1.5$. The graphs indicate that for lower frequencies, the damping increases from 0.5 to approximately 1 (except for incompressible soil) and remains constant for higher frequencies.

For rocking motion (see Fig. 2.14), the behaviour of the stiffness coefficient is more stable for lower Poisson's ratios, while it severely drops for incompressible soil. On the other hand, the damping coefficient is more stable for all Poisson's ratios in the rocking motion and behaves almost asymptotically after $a_0 \simeq 2$.

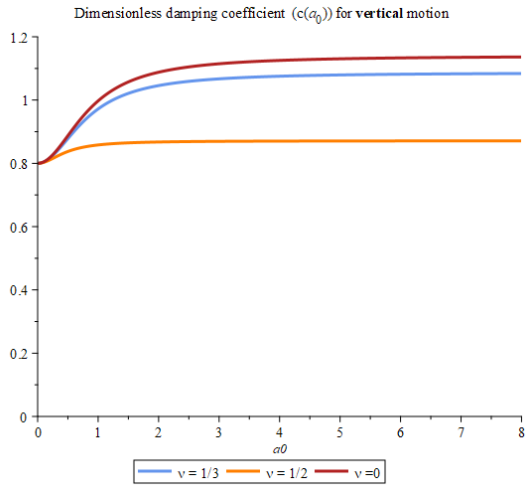


(a) $k_v(a_0)$ coefficient for vertical motion.

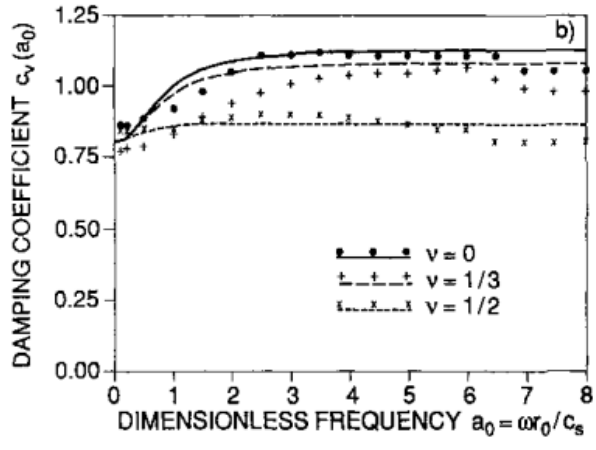


(b) $k_v(a_0)$ coefficient for vertical motion from the literature [60].

Figure 2.12: Comparison of dimensionless vertical stiffness coefficient $k_v(a_0)$ – Calculated vs. graphs from the original publication.

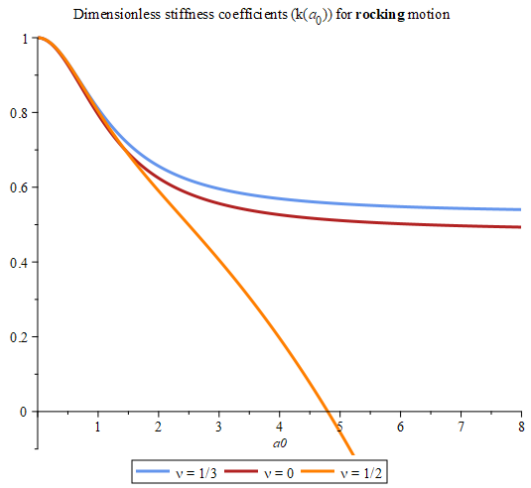


(a) $c_v(a_0)$ coefficient for vertical motion.

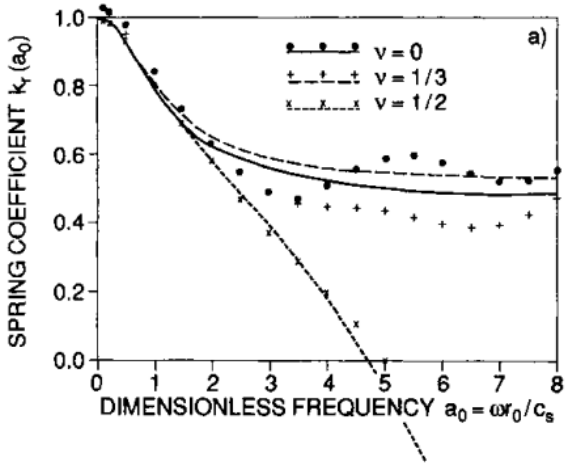


(b) $c_v(a_0)$ coefficient for vertical motion from the literature [60].

Figure 2.13: Comparison of dimensionless vertical damping coefficient $c_v(a_0)$ – Calculated vs. graphs from the original publication.

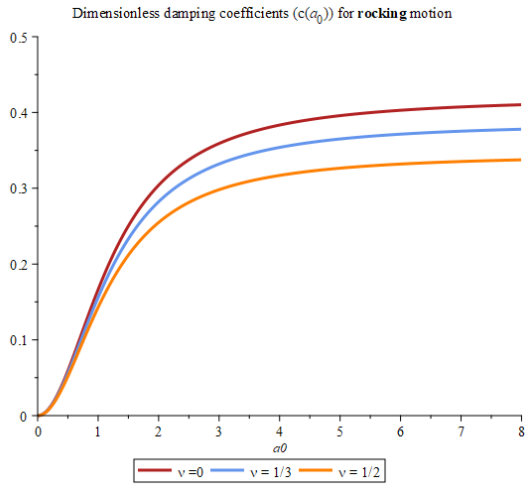


(a) $k_r(a_0)$ coefficient for rocking motion.

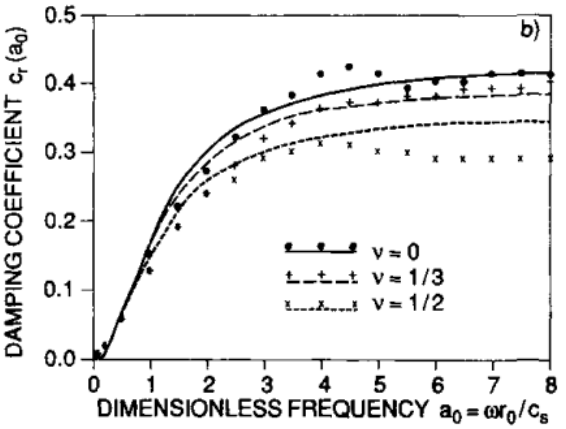


(b) $k_r(a_0)$ coefficient for rocking motion from the literature [60].

Figure 2.14: Comparison of dimensionless rocking stiffness coefficient $k_r(a_0)$ – Calculated vs. graphs from the original publication.



(a) $c_r(a_0)$ coefficient.



(b) $c_r(a_0)$ coefficient from the literature [60].

Figure 2.15: Comparison of dimensionless rocking damping coefficient $c_r(a_0)$ – Calculated vs. graphs from the original publication.

2.3. Modelling approaches for Soil-Structure Interaction for "Pile in soil under axial vibrations"

Piles are widely used in civil engineering for foundations. Consequently, their interaction with the surrounding soil has been broadly studied and analysed. Several scientists have studied different numerical and analytical approaches. There are mainly 3 classical approaches. The first is the *beam on Winkler-foundation*, contemplates the resistance of the soil as a "set of independent, linear, or nonlinear distributed springs" [25] and adds masses and dashpots to account for the inertia and wave radiation respectively. The second one refers to the *finite element formulations*, where axisymmetric models are used. The third and last one uses *elastic continuum-type formulations*, where three-dimensional equations of motion are solved [25]. However, some hybrid approaches have been studied recently for the solution of piles [23] [13]. Novak's formulation, which will be looked into in this work, falls into the first category and includes inertia and energy-dissipation capacity.

Additionally, pile-soil interaction can be analysed from different perspectives, depending on the effect of interest. "Pile dynamics finds its applications in machinery foundations, pile driving and in structures exposed to dynamic loads such as wind or earthquake" [31]. The interest in this topic relies upon the effect of the pile-soil interaction on the stiffness of the pile, the damping in the soil and the stiffness of the superstructure.

An important remark to this mechanical analogue for piles is that it gives a local response for slender piles. This means that the springs and dashpots attached to the pile that account for the interaction between this and the soil are not coupled or connected and that the pile is considered infinitely long. In other words, the response given by each spring and dashpot is local and individual and not affected by that of the adjacent ones, and the model might not apply to the response of compact piles. More advanced models consider the pile's global response, where the interaction between the springs is also accounted for or where different L/D ratios are evaluated. Ultimately, Novak's model gives an accurate solution for specific cases and cannot be used to solve any type of pile embedded in soil.

2.3.1. Novak's model for dynamic stiffness of piles

This type of foundation is significantly affected by the pile's dimensions and slenderness and the friction between the pile and the soil. The main assumption for this model is that the soil reactions at the foundation base and surface are the same and that the soil reactions on the side of the foundation are "equal to those of an independent layer overlaying the soil beneath the footing" [34]. Additionally, the soil is constructed by "independent infinitesimally thin horizontal layers that extend to infinity" [37] as proposed in [36]. The foundation is cylindrical and embedded, vertical and unaffected by surrounding piles. This last point states that the pile-group effect is neglected in this model.

This problem states that the pile is embedded in the soil resting on a rigid bedrock, which defines a fixed boundary condition at the bottom of the pile and only allows radiation conditions in the r-direction. This means that displacements and rotations in the z-, r-, and rotational directions are equal zero or the displacement of the bedrock itself. This is the case described by Novak [37]. The arrangement is depicted in Figure 2.16, where the springs and dashpots represent the impedance given by the soil against the pile under vertical motion.

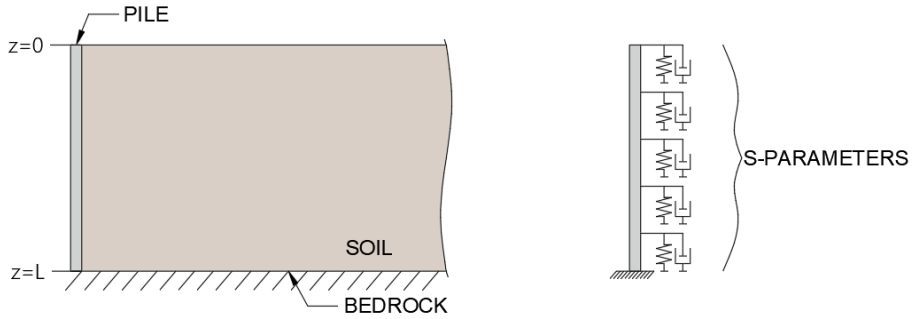


Figure 2.16: Diagram and representation of pile in soil over bedrock for vertical motion.

The displacements of interest are measured at the cap of the pile. Firstly, the vertical displacement w consequence of a vertical load, which also yields a vertical normal reaction R_w. The horizontal displacement u and the reaction of the soil R_x and the rocking angle ψ and reaction R_ψ are coupled and the analysis would need to consider this. However, in this thesis, only the vertical motion will be considered.

Vertical vibrations

The reaction of the soil along the vertical axis z given when the pile is subjected to a vertical excitation is expressed in Equation 2.40. In the equation, G is the shear modulus of the soil, and S_{w1} and S_{w2} are the impedance functions of the soil, dependent on the dimensionless frequency a_0 (see Eq. 2.41 and Eq. 2.42, respectively).

$$G(S_{w1} + iS_{w2})w(z,t) \tag{2.40}$$

$$S_{w1}(a_0) = 2\pi a_0 \frac{J_1(a_0)J_0(a_0) + Y_1(a_0)Y_0(a_0)}{{J_0}^2(a_0) + {Y_0}^2(a_0)} \tag{2.41}$$

$$S_{w2}(a_0) = \frac{4}{{J_0}^2(a_0) + {Y_0}^2(a_0)} \tag{2.42}$$

In the impedance functions, $J_0(a_0)$ and $J_1(a_0)$ are Bessel functions of the first kind of order zero and one respectively, and $Y_0(a_0)$ and $Y_1(a_0)$ are Bessel functions of the second kind of order zero and one respectively. They were derived by solving the complex displacement of an infinitely thin soil layer, considering the wave propagation of a 1D problem along the radial direction [37] [2]. The behaviour of these impedance functions is plotted and compared to the ones presented in the literature [37] (see Fig. 2.17) for validation purposes.

Considering the pile as a rod and using the impedance functions to account for the soil's reaction, the displacement of the pile along its length can be calculated with the equation of motion in Equation 2.43.

$$\mu\frac{\partial^2 w(z,t)}{\partial t^2} + c\frac{\partial w(z,t)}{\partial t} - E_p A\frac{\partial^2 w(z,t)}{dz^2} + G(S_{w1} + iS_{w2})w(z,t) = 0 \tag{2.43}$$

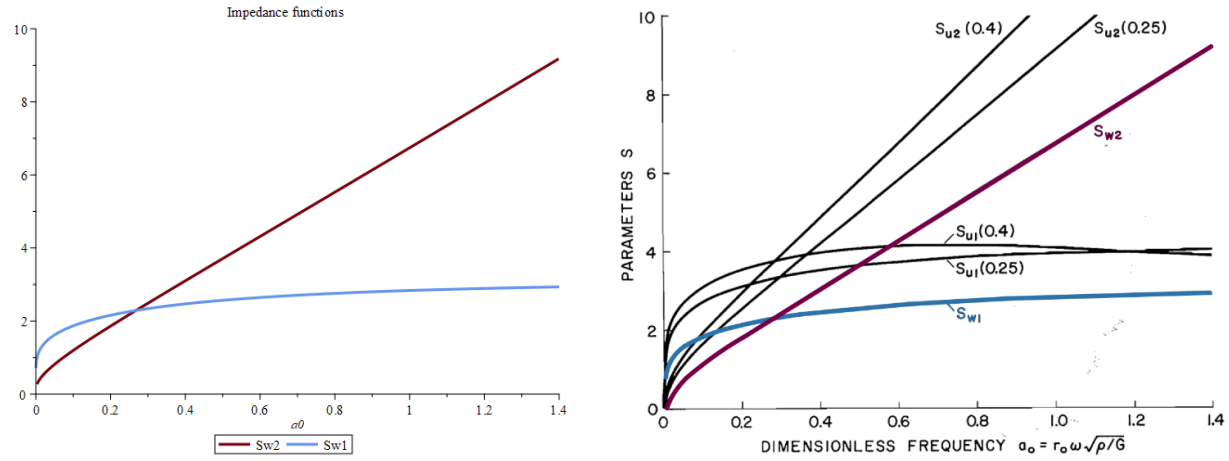
where μ is the mass of the pile per unit length, c is its damping coefficient related to the energy dissipated in the pile material, which will be assumed to be zero, E_p is the elastic modulus and A is the cross-sectional area of the pile. For this work, the pile was considered a concrete pile, which is representative of engineering practice. Henceforth, the pile's material properties are the density $\rho_p = 2400$ [kg/m ^{3}] and the elasticity $E_p = 3.2e10$ [Pa].

This equation of motion can be transformed into an ordinary differential equation in the frequency domain with an assumed solution of the shape $w(z,t) = w(z)e^{i\omega t}$. This yields a general solution that can be solved with the following boundary conditions: zero displacement (see Eq. 2.46) at the bottom and the normal force reaction at the top (see Eq. 2.45).

$$N(x) = EA\frac{dw}{dz} \tag{2.44}$$

$$N(z=0) = -F \tag{2.45}$$

$$w(z=L) = 0 \tag{2.46}$$



(a) Calculated.

(b) From literature ([37], fig.2, page 576).

Figure 2.17: Comparison of the S parameters for vertical soil's reaction to piles.

By using the impedance functions presented previously and comparing them to the graphs found in the literature, it can be assured that the equations are well implemented for further analysis. It can be observed from the plots in Figure 2.17a that they replicate the ones presented in the literature, shown in Figure 2.17. The coefficient S_{w1} represents the stiffness of the soil and its behaviour is relatively constant at around 2. On the other hand, S_{w2} presents a quasi-linear performance, especially if the curvature at the lowest frequencies is disregarded, and it represents the damping characteristics of the soil's reaction. This impedance function of the soil is added to the equation of motion of the pile as a stiffness (and damping due to the imaginary part) term.

2.4. Conclusions

The available mechanical analogues for the disc over homogeneous halfspace and the pile embedded in the soil exhibit superior quality. This analysis indicates that they can be used in practice with the assurance that a FE model is not needed at least for the purpose of describing the dynamic stiffness from the soil-foundation system provided to the superstructure. Moreover, these results ensure their adequacy as benchmarks for further steps in this work. However, these simple types of foundations are rarely found in practice nowadays, and there is a lack of accurate analogues for more complicated arrangements. Conclusively, the development of a computational approach is valuable for engineering practice to guarantee high-quality analogues for any type of foundation system.

3

Computational methods in SSI: optimisation, Machine Learning and Neural Networks

The main objective of this work is to determine whether it is possible to obtain equivalent models to the SSI problems explained in Chapter 2 using classical computational approaches combined with Artificial Intelligence. To do so, several previous steps will be taken. Firstly, a conventional optimisation approach will be studied for a simpler problem: a 3DoF system. Eventually, more complex tools, like Feedforward Neural Networks, will be applied to SDoF, 2DoF and 3DoF systems as benchmark cases to demonstrate their potential for the problem at hand. In the end, the goal is to apply Feedforward Neural Networks and optimisation tools to SSI problems to derive impedance functions and test their accuracy. Before delving into these interesting topics, basic concepts like Artificial Intelligence, Machine Learning and Neural Networks will be introduced.

3.1. Artificial Intelligence, Machine learning and Neural Networks

The term *Artificial Intelligence* (AI) was first mentioned in the 1950s at a computer science conference in Dartmouth, where scientists proposed that computers could be programmed to think and reason in the same way humans do [46]. In essence, AI is a discipline that aims to solve problems using creativity and by learning from previous knowledge, mimicking human cognitive processes. Early examples of AI include Shallow Neural Networks (1960s), Recurrent Neural Networks (1980s), and Unsupervised Deep Learning (1990s) [11]. These instances all belong to the subset of AI known as *Machine Learning* (ML), which focuses on "machines learning from data without being explicitly programmed" [51]. ML utilises *datasets* to train machines through *algorithms*. Various Machine Learning methods exist, including the aforementioned types, with *Neural Networks* (NN) being another prominent example. Figure 3.1 illustrates the classification hierarchy within the field of Artificial Intelligence.

Neural Networks is a "machine learning algorithm inspired by biological neural networks" [43]. Like human neural networks, a computational (also called Artificial) Neural Network (ANN or simply NN) is composed of neurons that transport information from one to the next. These connections are given by connecting channels. These are called "weighted channels because each of them has a value attached to it" [46]. They resemble the neural synapses of the human brain.

Among the various available models defined as Neural Networks, the most frequently used is the *Feedforward Neural Networks* (FNN) – also known as *Multilayer Perceptrons* (MLPs) [16]–. The name refers to the direction of the information flow, which is always forward, as there are no feedback connections where the output of the model can be fed back into itself [12].

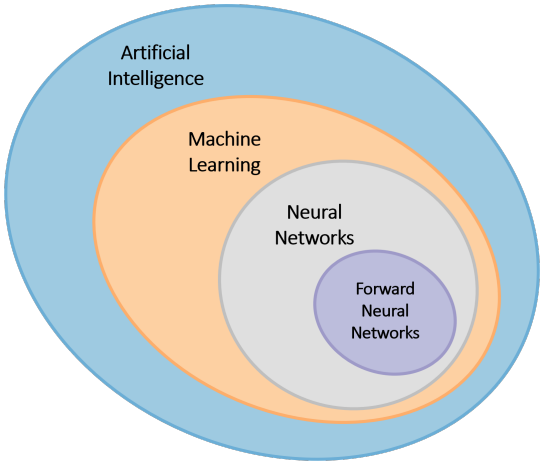


Figure 3.1: Diagram of AI umbrella, including ML, NN and FNN [11].

3.1.1. Feedforward Neural Networks

A generic Feedforward Neural Network (FNN) is illustrated in Figure 3.2. The figure distinguishes the various layers of an FNN. The first layer is the *input layer*, where the number of neurons corresponds to the number of input parameters. The subsequent layers, four in this case, are known as *latent* or *hidden layers* and consist of *neurons*, depicted as blue dots. The number of layers indicates the network's depth, and the number of neurons in each layer is not necessarily constant. Furthermore, each layer is connected to the next by *weights*, which are typically initialised randomly and subsequently determined by the network itself. The final layer is the *output layer*, representing the number of output parameters generated by the Neural Network [44].

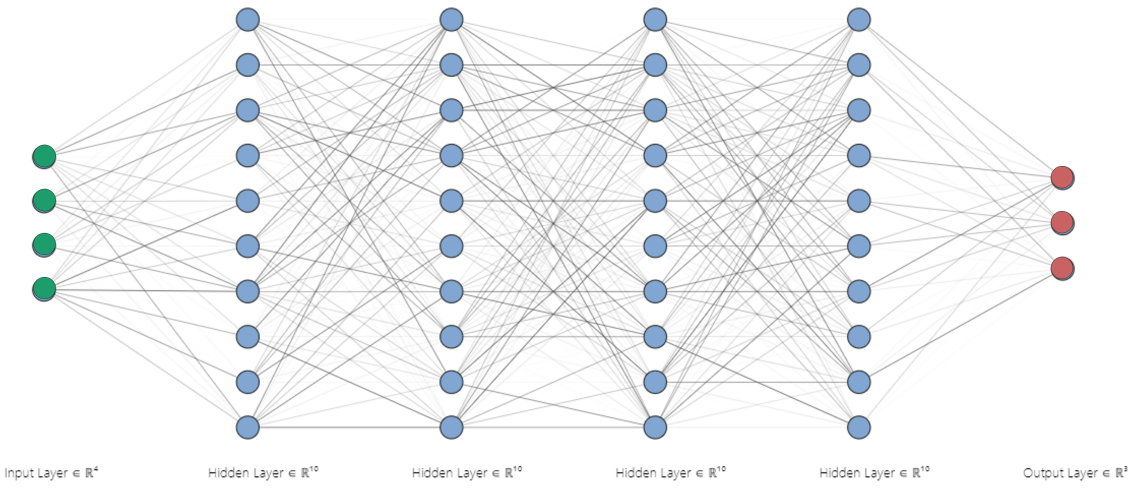


Figure 3.2: Example of Forward Neural Network.

ML models can be categorised as supervised or unsupervised. The former refers to the models whose goal is to predict a target variable based on a set of inputs, in which the inputs are labelled but the relationship between them is unknown. Contrarily, the latter's goal is to discover patterns or relationships in the unlabeled input dataset where the target is not defined [51]. In this work, *supervised learning* ML will be used, given that the dataset and expected outcome are labelled.

Supervised FNN are built up by *parameters* and *hyperparameters*. The former are learnt during the training process and determined by the NN itself. The latter are specified by the user before entering the learning algorithm [12] and require calibration through a validation process [3].

Parameters

The performance of a Neural Network is governed by its parameters. Among these, the linking *weights* between neurons across layers are particularly significant. Greater weights between two neurons signify that the associated substantially impacts the subsequent layer and its output. Conversely, lower weight values suggest a lesser effect on the following layer [12]. In Figure 3.3a, the weights are represented by w_{j1} to w_{jD}. The second parameter of Neural Networks is the *bias*. This parameter is a constant offset that refines the connection through the layers. It is attributed to the neurons to give more freedom although it can be omitted. The effect of the weights and bias to the predicted value is defined in Equation 3.1:

$$\hat{y} = \boldsymbol{w}^T \boldsymbol{x} + b \tag{3.1}$$

where $\hat{y}$ is the predicted output, $\boldsymbol{w}$ is the weight that multiplies $\boldsymbol{x}$, which represents the input and b is the so-called bias. It describes the linear transformation between two layers. Finally, the number of parameters is given by the number of neurons and hidden layers, which are some of the hyperparameters.

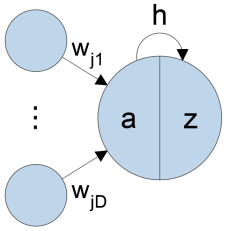
Hyperparameters

As previously mentioned, the hyperparameters must be calibrated and set by the user using previous knowledge or optimisation. The objective is to avoid over- or underfitting a model. The relevance relies on ensuring the correct result of the network. On the one hand, an overfitting model indicates that it follows the original dataset extremely closely and will not be applicable if the inputs are slightly modified. On the other hand, an underfitting model will not give enough information in the output and the result will be too inaccurate.

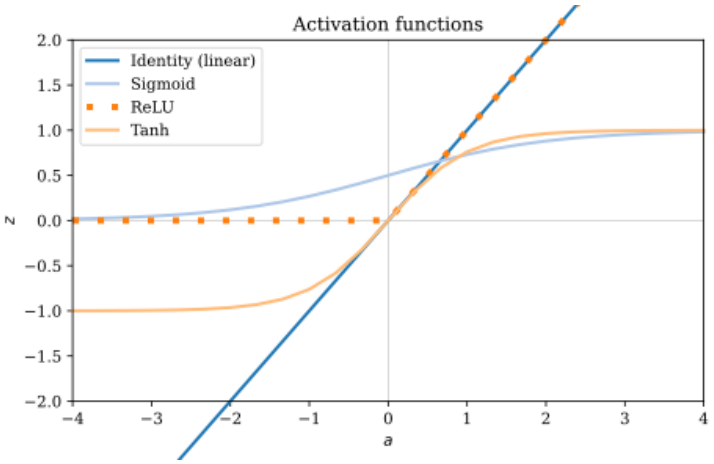
The model hyperparameters are the ones that determine the *architecture* of the Neural Network, meaning the shape. The two main hyperparameters are the *number of neurons* per layer, usually represented with N, and the *number of hidden layers* in the Neural Network, marked as L. These two have a great impact on the behaviour of the Neural Network as they determine the number of parameters and these need to be balanced.

Apart from the number of layers and neurons, the *activation function* is a hyperparameter that must be selected. This can be chosen depending on the application and is responsible for introducing non-linearity to the problem. The activation functions (h) do so by passing the value in a neuron (a), obtained from the linear biased information from the previous layer (with neurons w_{j1} to w_{jD}), through it to obtain a second neuron value (z) (see Fig. 3.3a and Eq. 3.2) [44]. Additionally, some of the available activation functions are shown in Figure 3.3b.

$$z_{ij} = h(a_{ij}) \tag{3.2}$$



(a) Activation function representation [44].



(b) Most common activation functions [3].

Figure 3.3: Activation functions.

Training parameters to control the model complexity

To control the training process, more choices have to be made. One of these is the rate at which the weights are updated, in other words how quickly the minimisation is done [8], defined as *learning rate* (η). Another one is the *batch size*, which is the number of training examples from the training set used in one iteration, where the number of iterations will depend on the number of samples in the training dataset and the batch size. The *epochs* is then the number of times the entire training dataset is passed through the network. This means that "we can divide the dataset of 2000 examples into batches of 500 and it will take 4 iterations to complete 1 epoch" [47].

Another important training choice is the solver or *optimiser*, which is the algorithm through which the loss function is minimised. The most common solvers are the "stochastic gradient descent", mostly called *sgd* and the *adam* solver, which is an adaptation *sgd* solver. Additionally, the *loss function* quantifies the difference between the predictions of a model and the target with, most commonly, a squared loss function. However, it can be modified if needed. The squared loss is defined in Equation 3.3 [3]:

$$L(t, y(\boldsymbol{x})) = [y(\boldsymbol{x}) - t]^2 \tag{3.3}$$

where L indicates the loss function, t is the targeted value and $y(\boldsymbol{x})$ is the predicted value.

Finally, the Neural Network's regularisation can be done with different techniques. Two will be considered in this work. The first consists of applying a regularisation factor λ that penalizes the weights between the neurons to avoid overfitting [41] [63]. It is determined by the user and is usually turned off during testing [8]. It is applied with the technique L2, which "puts a penalty on weights and prevents them from taking too large values unless supported by the data" [3]. This term is added in the error calculation as in Equation 3.4:

$$E(\boldsymbol{w}) = \frac{1}{2}\sum_{n=1}^{N}\left(t_n - \boldsymbol{w}^T\phi(x_n)\right)^2 + \frac{\lambda}{2}\boldsymbol{w}^T\boldsymbol{w} \tag{3.4}$$

where E is the error, t_n is the target, $\boldsymbol{w}$ are the weights, $\phi(x_n)$ is the input dataset and λ is the regularisation factor.

The last regularization technique is *early stopping*. This technique consists of breaking the training process of the NN when the validation loss reaches a valley or has no improvement after a determined number of epochs. For instance, if there is no improvement in the validation

loss for 10 epochs, the training loop is stopped. This ensures that the NN is not trained until overfitting.

Training, validation and testing datasets

The target variables of the NN are made up of samples, also called instances. This information must not be shown entirely to the Neural Network during training because the resulting model must be flexible enough. For that purpose, the dataset used in the NN is divided into training, validation and testing datasets, which are subsets of the original dataset. The rule of thumb is to divide them into 70%, 20% and 10% respectively, although the proportions can vary.

As the name indicates, the first 70% is used to train the Neural Network. Then 20% is used to calibrate the model hyperparameters. Finally, the last 10% of the samples in the testing dataset are hidden from the NN for the entire process and only used to understand the behaviour of the trained and validated NN. The training dataset should be used in the Neural Network with a random shuffle to improve the training process [3].

3.2. Optimisation and Neural Networks for dynamic stiffness identification

Being an unexplored method due to the complexity of the SSI systems, the computational approach should be previously evaluated with simpler physical models and through simpler methods. This will facilitate a more profound comprehension of the tools and anticipate potential drawbacks. Different simple systems will be evaluated and various computational approaches will be used. In the following subsections the analysed systems, the data-driven approaches and the results will be presented. Both the analysed systems and the methods will gradually increase in complexity.

3.2.1. Conventional optimisation approach for a 3DoF system

In the field of dynamics, discrete masses, springs, and dashpots serve as fundamental elements in systems to model the majority of linear dynamical systems. Being discrete, these systems allow to compute and calculate the system's response directly. These elements represent the element's mass, stiffness and damping of the system and, with the equations of motion, the exact response of these systems can be derived. The objective of this section was to approximate the response of a discrete 3DoF system using conventional optimisation tools. This serves as an introduction to data-driven approaches and optimisation methods with simple systems.

The system consists of three discrete masses (3DoF) connected by springs and dashpots with one degree of freedom each: displacement in x. A representative image can be seen in Figure 3.4.

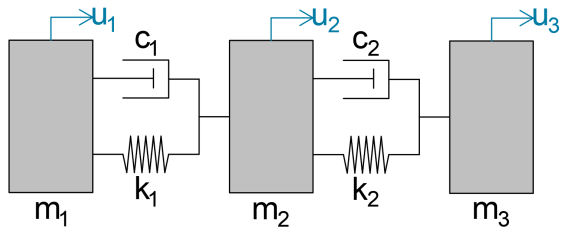


Figure 3.4: 3DoF system.

The system's equation of motion in the frequency domain is shown in Equation 3.5 below:

$$-\omega^2 \underline{\underline{M}}\ \underline{\tilde{u}} + \underline{\underline{K}}\ \underline{\tilde{u}} + i\omega \underline{\underline{C}}\ \underline{\tilde{u}} = \underline{\tilde{p}} \tag{3.5}$$

where ω is the frequency, $\underline{u}$ is the displacement vector and $\underline{\underline{M}}$, $\underline{\underline{C}}$ and $\underline{\underline{K}}$ are the system matrices of the mass, stiffness and damping coefficients:

$$\underline{\underline{M}} = \begin{bmatrix} m_1 & 0 & 0 \\ 0 & m_2 & 0 \\ 0 & 0 & m_3 \end{bmatrix} \quad \underline{\underline{C}} = \begin{bmatrix} c_1 & -c_1 & 0 \\ -c_1 & c_1 + c_2 & -c_2 \\ 0 & -c_2 & c_2 \end{bmatrix} \quad \underline{\underline{K}} = \begin{bmatrix} k_1 & -k_1 & 0 \\ -k_1 & k_1 + k_2 & -k_2 \\ 0 & -k_2 & k_2 \end{bmatrix} \tag{3.6}$$

From Equation 3.5, the flexibility matrix ($\underline{\underline{\tilde{F}}}$) can be obtained. This matrix represents the behaviour of the system when a specific force ($\tilde{p}$) is applied to it. In a forward problem, the multiplication of a flexibility matrix and the external force gives the frequency-dependent response of the system ($\underline{\tilde{u}}$), which is also called the Frequency Response Function (FRF). If the response is plotted, the peaks indicate the frequencies at which the displacement of the

system is amplified, indicating resonance at the system's natural frequencies. The flexibility matrix is:

$$\underline{\underline{\tilde{F}}} = (-\omega^2 \underline{\underline{M}} + \underline{\underline{K}} + i\omega \underline{\underline{C}})^{-1} \tag{3.7}$$

Optimisation

For the inverse approach, the Least-Squares tool from the scipy library in Python was applied [53] [52]. The objective was to find the mass ($\underline{\underline{M}}$), damping ($\underline{\underline{C}}$) and stiffness ($\underline{\underline{K}}$) matrices with values and responses similar to the original. In other words, to approximate the system's (MCK) matrices' values for an unknown 3DoF system using a computational approach. The FRF was obtained from a system with random parameters and then the system matrices were approximated using Least-Squares estimation parameters. The objective function (see Eq. 3.8) consists of minimizing the difference between the real and imaginary displacements ($\underline{u}$) given by the original system matrices against the real and imaginary displacements ($\underline{u'}$) obtained with approximated system matrices. Given that the original matrices to which the system will be approximated are symmetric, so will the approximated system matrices.

$$|\underline{u'}^{Re,Im} - \underline{u}^{Re,Im}| \simeq \underline{0} \tag{3.8}$$

As an additional validation, the peaks of the FRF should coincide with the natural frequencies (ω_n) obtained through modal analysis. For such purpose, the eigenvalues and eigenvectors of matrix A, resulting from the multiplication of the inverse mass matrix ($\underline{\underline{M}}^{-1}$) and the stiffness matrix ($\underline{\underline{K}}$), were calculated:

$$\underline{\underline{A}} = \underline{\underline{M}}^{-1} \cdot \underline{\underline{K}} \rightarrow \begin{cases} \lambda_n = \text{eigVl}[\underline{\underline{A}}] \rightarrow \omega_n = \sqrt{\lambda_n} \\ \Phi_n = \text{eigVc}[\underline{\underline{A}}] \end{cases} \tag{3.9}$$

where λ_n results from the eigenvalues[1] (eigVl) and its square root is the system's real-valued natural frequencies (ω_n), and the latter results from the eigenvectors[2] (eigVc) and represents the modal shapes (Φ_n) of each natural frequency.

Figure 3.5 depicts the response of the original system with random M, C, and K matrices, where three peaks can be observed. Each peak corresponds to a natural frequency, indicating the system has 3DoF and possesses light damping.

[1] An eigenvalue of an NxN matrix $\underline{\underline{A}}$ is a scalar such that the equation $\underline{\underline{A}}\underline{v} = \lambda \underline{v}$ has a non-trivial solution [26]

[2] An eigenvector of an NxN matrix $\underline{\underline{A}}$ is a non-zero vector $\underline{v}$ such that $\underline{\underline{A}}\underline{v} = \lambda \underline{v}$, for some scalar λ [26]

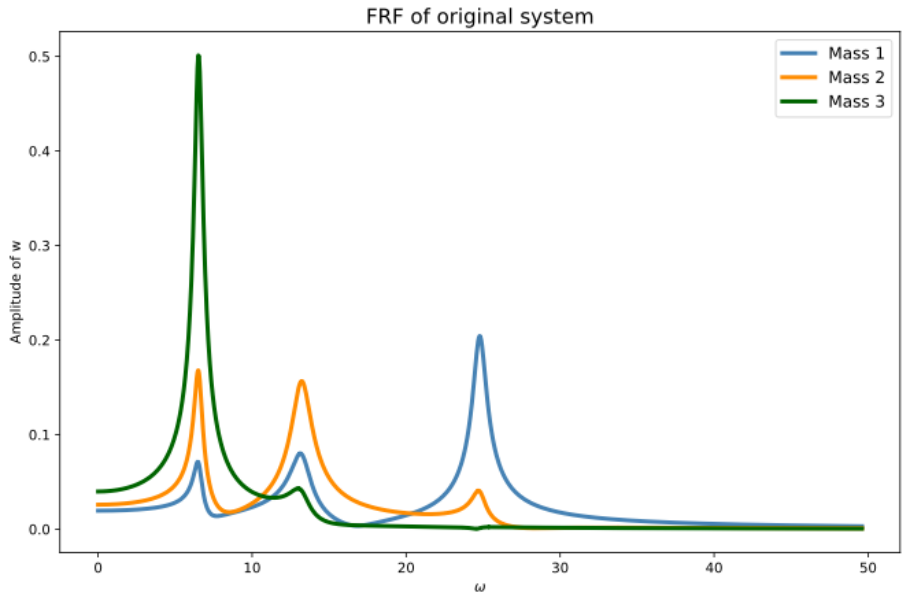


Figure 3.5: FRF of original 3DoF system.

By identifying the number of peaks in the original FRF, it is possible to determine the number of degrees of freedom and, consequently, the sizes of the matrices for the approximation problem: 3x3. The Least-Square function was then set up and evaluated by defining initial values. The results of the approximation are shown in the following figures (see Figs. 3.6, 3.7, 3.8). They respectively show that real, imaginary and absolute results of the FRF for both the original and approximated system are extremely similar. This demonstrates that this optimisation method works adequately.

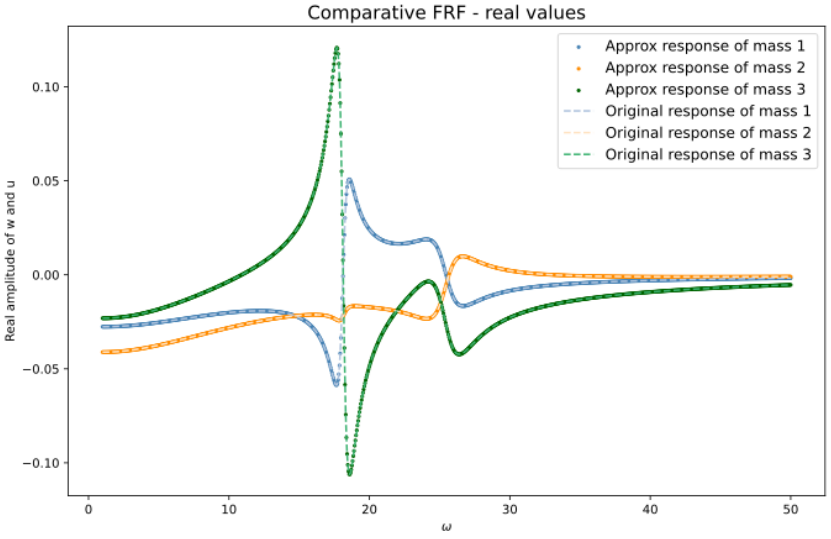


Figure 3.6: Comparison of real part.

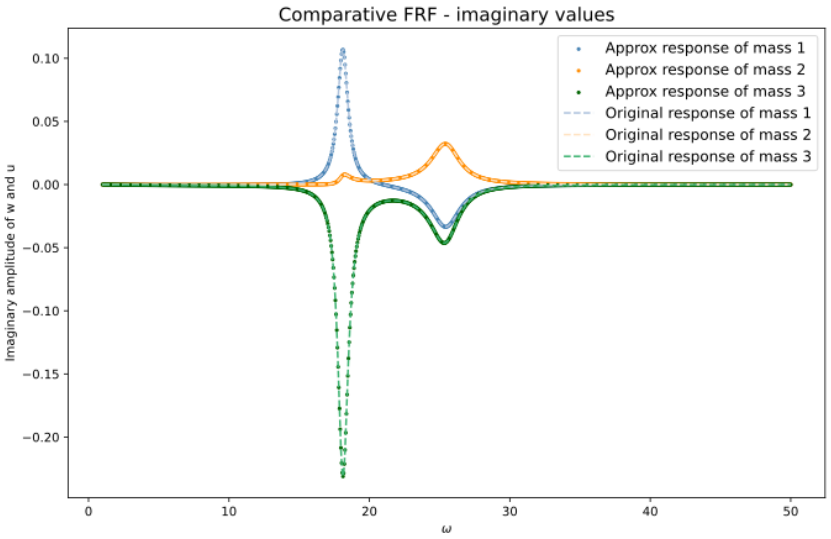


Figure 3.7: Comparison of imaginary part.

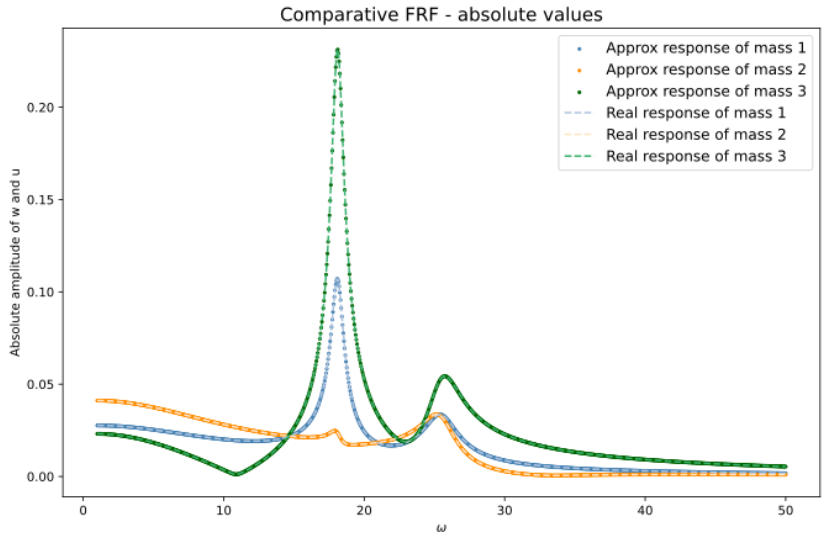


Figure 3.8: Comparison of absolute value.

The error between the displacement of the original system (u) and the displacement of the approximated system (u') is in the order of magnitude of 10^{-14}. This indicates that the optimisation method is appropriate to obtain an approximated model. However, the algorithm's behaviour and ability to achieve convergence depend on the initial guess. To assess the robustness of the approximation, various sets of initial guesses were tested, demonstrating the method's capability for effective optimisation.

3.2.2. Feedforward Neural Networks for SDoF and 2DoF systems

After understanding the scope and limitations of conventional optimisation methods, FNN are investigated. This section's objective is to grasp the particularities of FNN with a similar SDoF system and eventually scale it to 2- and 3DoF systems. Firstly, all the combinations of the options for parameters and hyperparameters were evaluated for the SDoF system. The options are shown in Table 3.1. For this problem, the inputs are the points in the FRF, meaning the

system's response to some of the frequencies in the range of analysis. On the other hand, the outputs are the system's response predicted by the NN after the training process. In other words, some of the points of the FRF are used to train the NN in order to predict the displacement for other frequencies. The objective of this section was to demonstrate that an FRF of a system can be reconstructed using NN. Once again, this is a benchmark problem used as a bridging step.

Model hyperparameters	
Number of hidden layers	$[3, 4, 5, 6]$
Number of neurons	$[20, 25, 30, 35, 40, 45, 50, 55]$

Training choices	
Solver	sgd, adam
Activation function	identity, logistic, tanh, ReLU

Table 3.1: Parameters and hyperparameters for initial approach.

These parameters and hyperparameters were looped through to determine the best combination and thus the best Neural Network to approximate an SDoF. The original system is depicted in Figure 3.9 and corresponds to a randomly chosen system with mass $M = 5.4$, stiffness $K = 1.38$ and damping $C = 0.42$.

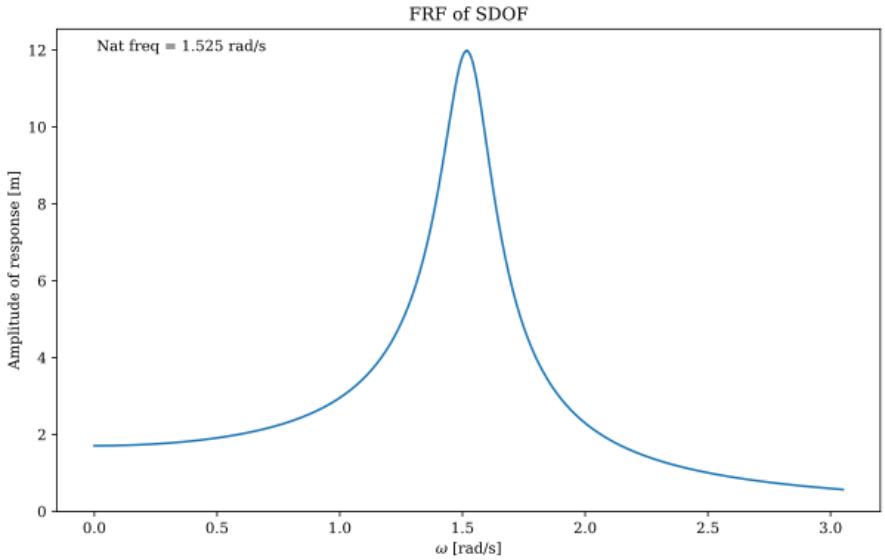


Figure 3.9: Original SDoF.

There were 68 possible combinations with a score higher than 0.99. The score is calculated as $[1 - (u/v)]$ where u is the residual sum of squares between the real value and the prediction and v is the total sum of squares between the real value and the mean of the real values and has a maximum value of 1 [45]. It is calculated in the validation set. The least number of hidden layers needed was 4, with 40 neurons. This architecture required an *adam* solver and ReLU activation function. However, an FNN with 20 neurons in 6 layers was also possible. This demonstrates that different architectures and hyperparameters can give an accurate solution for the same problem. Figure 3.10 shows the solution of the test set for the problem using the parameter combination with the highest score.

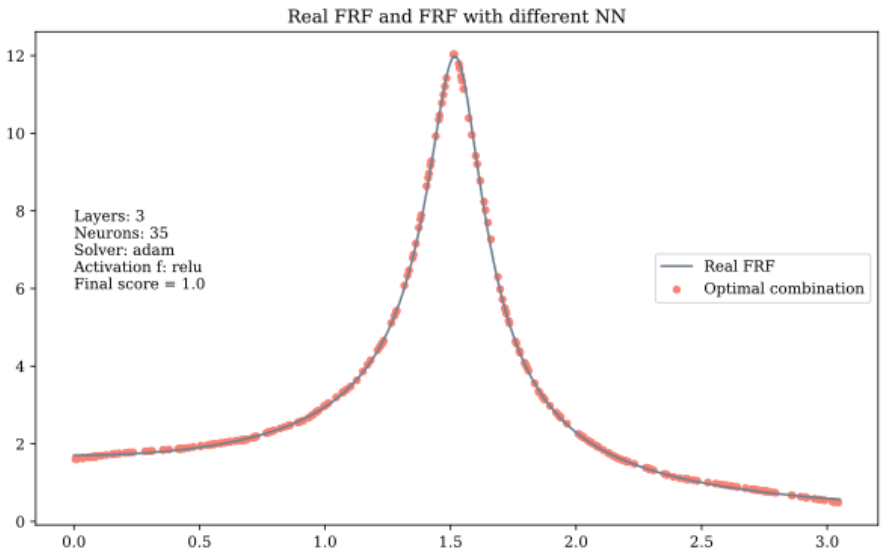


Figure 3.10: Approximated SDoF.

The same procedure was done with a 2DoF system, where the FRF now presents two peaks, corresponding to the two natural frequencies of the system. The same ranges for the model and the training parameters were used in FNN to obtain approximate models. Figure 3.11 shows the FRF of the original system and all the possible solutions for the problem at hand. This demonstrates that, in Machine Learning, there is not a unique solution but several for the same problem. Different Neural Network architectures allow accurate prediction. The best model would be the one with the least validation error. In these cases, this is still not looked into, as these models were used to understand the behaviour and capabilities of NN.

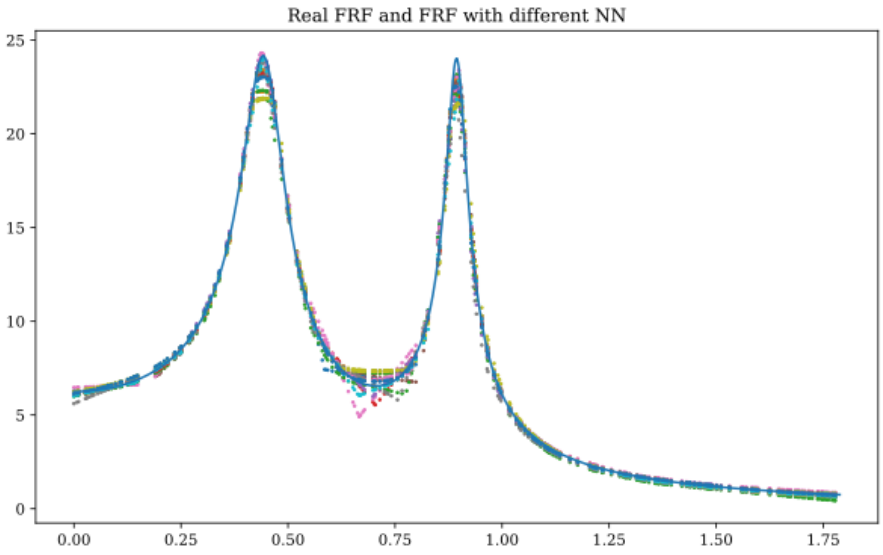


Figure 3.11: Comparison of the original 2DoF system and the system approximated through Neural Networks.

This first approach to NN using FNN confirmed that FRFs can be approximated using this method. In this case, the input of the FNN was the displacement of the dynamic system and the output was the approximated displacement of the input. Consequently, Figure 3.12 shows an illustrative example of the Neural Network architecture used for all the cases presented in

this section. The common hyperparameters in all iterative solutions were the single-neuron input and single-neuron output layers. The number of layers and the neurons varied in the ranges presented in Table 3.1, as well as the activation functions and solvers. The Figure is merely an example of one of the possible architectures with three layers.

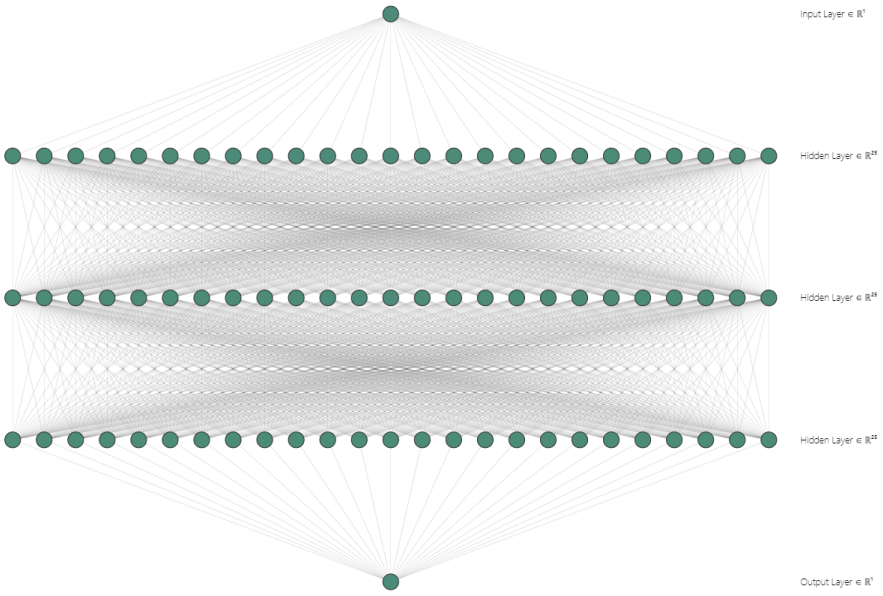


Figure 3.12: One possible NN architecture for the analysis of SDoF, 2DoF and 3DoF systems.

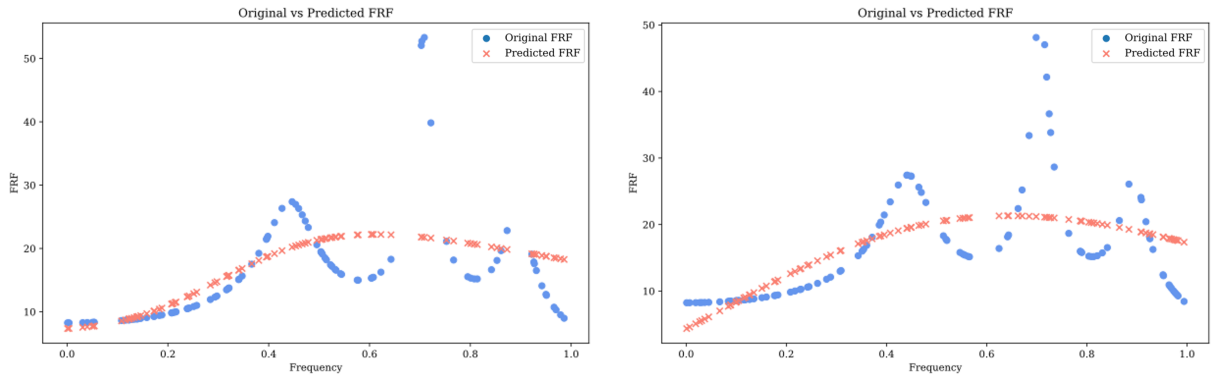
3.2.3. Feedforward Neural Networks for 3DoF system

The last step before tackling NN for SSI problems is solving 3DoF systems with PyTorch, a library in Python that can handle automatic differentiation [40]. Firstly, it can store the operations done in the process, making it possible to go backwards. Additionally, it can differentiate equations. The main benefits of this tool are that it is integrated with the conventional libraries of Python, like NumPy and SciPy and that it provides more flexibility in the ML and NN.

As a simple sensitivity analysis, different NN architectures were run for a 3DoF system using Pytorch. The results are shown to help understand the effect of each hyperparameter. All the analysed options have 1 input and 1 output vectors and include early stopping as a regularization technique. They use *adam* as the solver, a learning rate of 0.001, and a division of 70-20-10 for training, validation and testing respectively. The main adjustments are made in the number of layers and neurons per layer. However, the activation function is also changed.

1 hidden layer

Figure 3.13 shows the results of the approximation using a 1-hidden-layer NN with different hyperparameters (number of neurons and activation functions – ReLU and *tanh*–). The one on the left shows the result with 4 neurons, and the one on the right with 128, both with *tanh*. It can be observed that, regardless of the number of neurons, one hidden layer is not enough for the NN to capture the shape of the FRF with a *tanh* activation function. This depicts that more than one layer is surely needed.



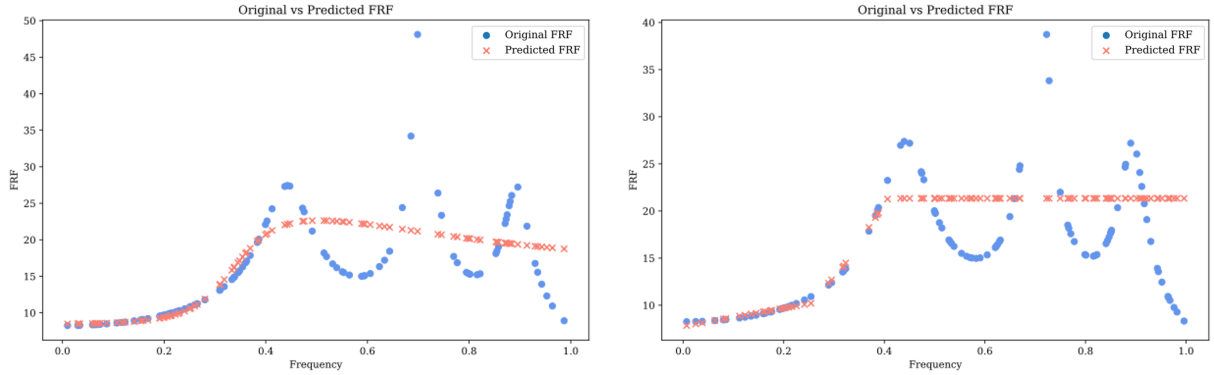
(a) 4 neurons and tanh activation function.

(b) 128 neurons and tanh activation function.

Figure 3.13: Predictions with the test set for an NN with 1 hidden layer.

3 hidden layers

After trying 1-hidden-layer NN, more layers were introduced. This section plots the results of Neural Networks with 3 hidden layers and different activation functions. Figure 3.14 shows the results for 4-neuron 3-hidden-layer Neural Networks. The one on the left results from using a *tanh* activation function, while the one on the right used ReLU. The behaviour of the NN in the lower frequency range is closer to the original FRF in both cases. However, neither gives good results for higher frequencies. An important remark is that the ReLU activation function presents a linear behaviour in the high-frequency range while the results with the *tanh* activation function are curved.

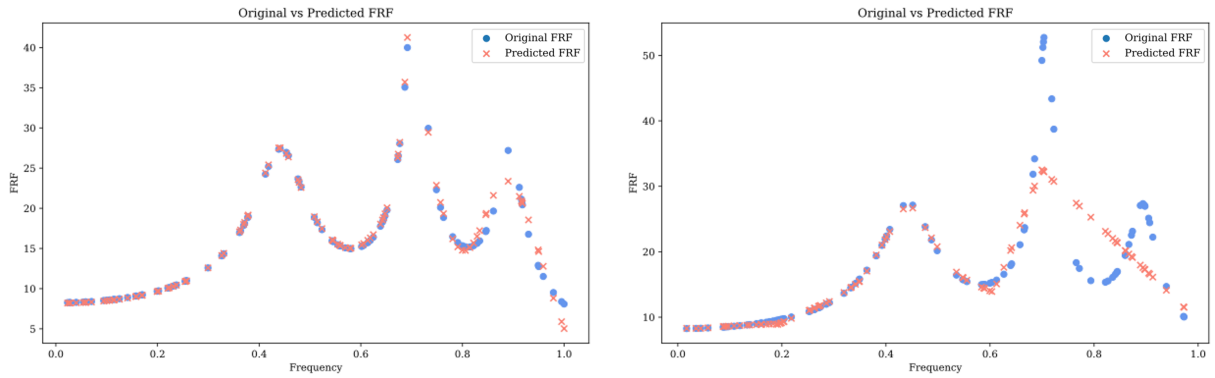


(a) 4 neurons and tanh activation function.

(b) 4 neurons and ReLU activation function.

Figure 3.14: Predictions with the test set for an NN with 3 hidden layers.

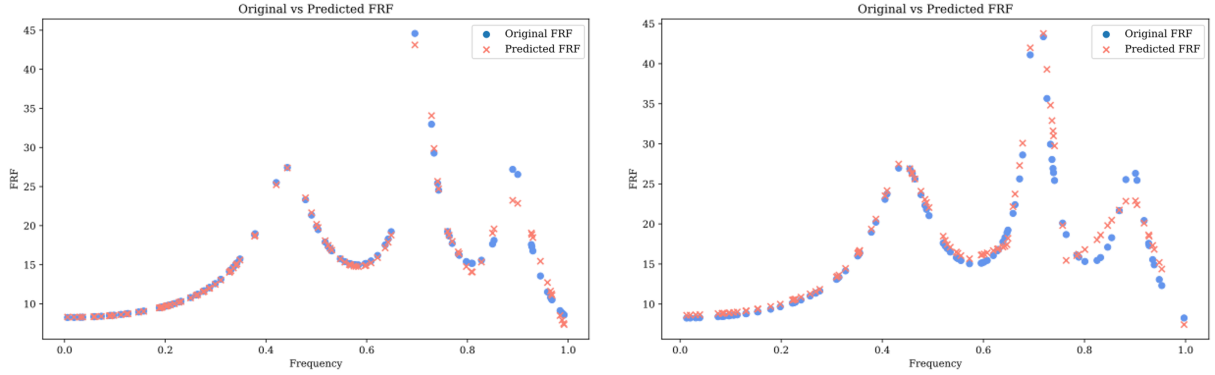
Augmenting the number of neurons enhances the Neural Network's capacity to find better solutions (refer to Fig. 3.15). However, the activation function also plays a crucial role. A configuration of three layers with 32 neurons and the *tanh* activation function closely approximates the shape of the original FRF (see Fig. 3.15a). Conversely, using the same architecture with the ReLU activation function does not yield a precise solution, as it tends to linearise at higher frequencies (see Fig. 3.15b).



(a) 32 neurons and tanh activation function. **(b)** 32 neurons and ReLU activation function.

Figure 3.15: Predictions with the test set for an NN with 3 hidden layers, 32 neurons each.

Finally, making layers with 64 neurons each does not give a significant improvement (visually), as can be observed in Figures 3.16. With *tanh* activation function (see Fig. 3.16a), the results are quite similar to the 32-neuron network. On the other hand, twice the neurons do improve the NN's behaviour with ReLU activation function (see Fig. 3.16b). It can be concluded that 3-layer NN can provide good solutions but the correct number of neurons per layer and activation function are needed.

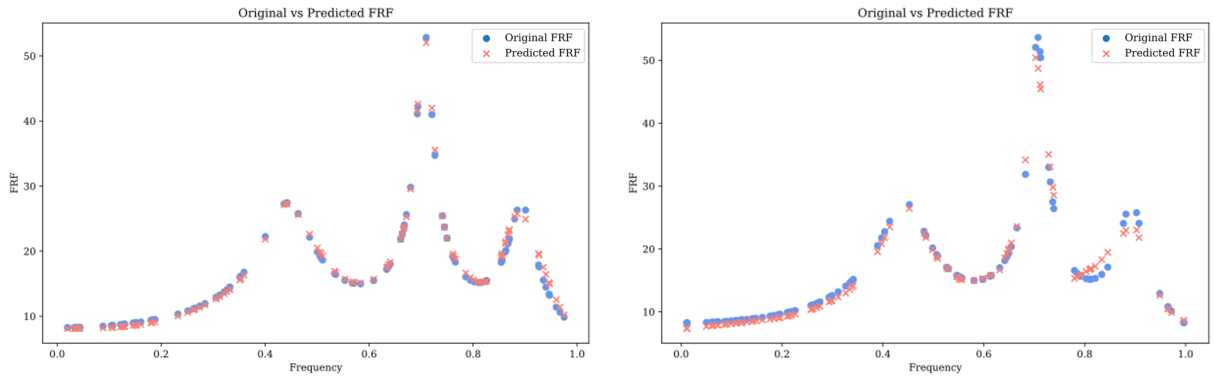


(a) 64 neurons and tanh activation function. **(b)** 64 neurons and ReLU activation function.

Figure 3.16: Predictions with the test set for an NN with 3 hidden layers, 64 neurons each.

4 hidden layer

Finally, 4 layers were introduced in the NN. Figure 3.17 shows the results for these cases. The figure on the left (Fig. 3.17a), which shows the results for a 4-layer 64-neuron NN with *tanh*, exhibits that the behaviour of the network is appropriate for this case. This model captures the last peak better than the 64-neuron 3-layer NN with *tanh* activation function. However, the figure on the right (Fig. 3.17b) indicates that ReLU activation function can also have an adequate performance but requires deeper NN.



(a) 64 neurons and tanh activation function.

(b) 64 neurons and ReLU activation function.

Figure 3.17: Predictions with the test set for an NN with 4 hidden layers.

This simplistic initial analysis shows that the number of layers probably impacts greater than the number of neurons per layer. This means that making the network deeper makes it more robust. Finally, the activation function has an important role in the behaviour of the network, and thus should not be chosen lightly. A complete analysis of this problem would require a full sensitivity analysis and a comparison of the training and validation errors which will be performed in the full problem.

3.3. Conclusions

Not only are there various computational approaches available for solving dynamic problems, but also a wide variety of parameters that play important roles and can be adjusted and fine-tuned in their application. The key lies in choosing the correct combination of criteria that allows for more flexibility while maintaining accuracy. After evaluating the available methods separately, the combination of optimisation and Neural Networks will be investigated to address the problem at hand: obtaining the entries of the system matrices based on the soil and foundation characteristics provided. In the following chapter, Neural Networks and optimisation methods will be used to solve the aforementioned SSI problems assessing their suitability and achieve the intended computational approach.

4

A Neural Network-based method for the discovery of mechanical analogues in SSI

4.1. General approach

The core of this work is to propose a generic approach for solving SSI problems with approximate impedance functions. It is based on a reduced model of the real problem, where the former is built using several computational tools (see Fig. 4.1). The objective of the reduction is to have a smaller model size that solves soil-structure interaction more efficiently without a major loss of accuracy. The response that will be sought is that of the foundation subjected to a frequency-dependent vertical force ($p_z(\omega)$), horizontal force ($p_x(\omega)$) and rocking moment ($M_y(\omega)$) applied independently. These forces are depicted in Figure 4.1 in both the advanced FE and the reduced model. The SSI problems investigated using this generic approach were disc over a homogeneous halfspace and pile in soil. Due to the axisymmetric nature of these cases, the force in the y-direction (p_y) and the rocking moment over the x-axis (M_x) can be neglected.

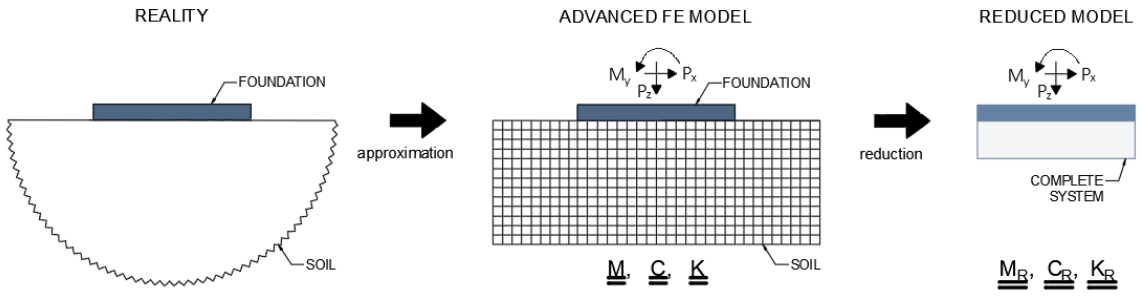


Figure 4.1: General problem statement of the proposed approach.

Ultimately, the objective is to derive reduced matrices that are substantially smaller in size compared to those generated by an advanced Finite Element model, thereby providing a more computationally effective yet accurate solution.

Advanced model

An advanced FE model can represent the real system. Said system can be any foundation on or in the soil, where the former is a bounded element with a higher rigidity than the latter, which is an unbounded domain. Large-size mass ($\underline{\underline{M}}$), damping ($\underline{\underline{C}}$) and stiffness ($\underline{\underline{K}}$) matrices are obtained from the FE model, correspondent to all the nodes that conform it. These matrices are used within the model to obtain the displacement to a given force, as shown in Equation 4.1.

$$(-\omega^2 \underline{\underline{M}} + i\omega \underline{\underline{C}} + \underline{\underline{K}}) \cdot \underline{\tilde{u}}_a(\omega) = \underline{\tilde{p}}_a(\omega) \tag{4.1}$$

In this equation of motion, the force ($\underline{\tilde{p}}_a(\omega)$) and the displacement ($\underline{\tilde{u}}_a(\omega)$) are frequency-dependent vectors. The former is applied to the system to compute the response. The latter is comprised of the responses in each degree of freedom to every force. In other words, the coupled horizontal u, vertical w and rotational φ displacements given by the independent forces $\tilde{p}_x(\omega)$, $\tilde{p}_z(\omega)$ and $\tilde{M}_y(\omega)$ (see Eq. 4.2).

$$\underline{\tilde{u}}_a(\omega) = \begin{bmatrix} \underline{\tilde{u}}_n(\omega) \\ \underline{\tilde{w}}_n(\omega) \\ \underline{\tilde{\varphi}}_n(\omega) \end{bmatrix} \quad \text{and} \quad \underline{\tilde{p}}_a(\omega) = \begin{bmatrix} \underline{\tilde{p}}_x(\omega) \\ \underline{\tilde{p}}_z(\omega) \\ \underline{\tilde{M}}_y(\omega) \end{bmatrix} \tag{4.2}$$

As discussed in Section 2.1, the reaction in every degree of freedom to each force corresponds to a column in the flexibility matrix's construction. Equivalently, the matrix in Equation 4.3 shows the displacement per degree of freedom per force in columns, which correspond to the displacement vectors from the advanced model.

$$\begin{bmatrix} u_x & u_z & u_M \\ w_x & w_z & w_M \\ \varphi_x & \varphi_z & \varphi_M \end{bmatrix} \begin{bmatrix} \tilde{p}_x(\omega) \\ \tilde{p}_z(\omega) \\ \tilde{M}_y(\omega) \end{bmatrix} = \begin{bmatrix} \underline{\tilde{u}}(\omega) \\ \underline{\tilde{w}}(\omega) \\ \underline{\tilde{\varphi}}(\omega) \end{bmatrix} \tag{4.3}$$

Due to the complex nature of the dynamic problem and the coupled response of the motions, the vectors $\underline{\tilde{u}}(\omega)$, $\underline{\tilde{w}}(\omega)$, $\underline{\tilde{\varphi}}(\omega)$ are composed of the real and imaginary parts of all the combinations. In other words, $\underline{\tilde{u}}(\omega)$ contains the value of the horizontal displacement due to the horizontal force $\tilde{p}_x$ for every frequency, denoted as $\underline{u}^R_{x,x}$ for the real part, where the first x marks the displacement and the second x marks the direction of the force. This term is a vector in itself because there is a solution for each frequency. Equation 4.4 shows the shape of the total vector.

$$\begin{bmatrix} \underline{u}^R_{x,x} & \underline{u}^I_{x,x} & \underline{u}^R_{x,z} & \underline{u}^I_{x,z} & \underline{u}^R_{x,M_y} & \underline{u}^I_{x,M_y} & \underline{w}^R_{z,x} & \cdots & \underline{w}^I_{z,M_y} & \underline{\varphi}^R_{\varphi,x} & \cdots & \underline{\varphi}^I_{\varphi,M_y} \end{bmatrix}^T \tag{4.4}$$

This extensive displacement vector is obtained from an advanced FE model for each frequency. The goal of the reduced model is to replicate this vector and retrieve the $\underline{\underline{M}}$, $\underline{\underline{C}}$ and $\underline{\underline{K}}$ matrices of the system in an inverse procedure. However, to reduce the number of DoFs and make the process more efficient, a reduced model will be sought.

Reduced model

The reduced model, attained using Neural Networks and optimisation tools, uses reduced mass ($\underline{\underline{M}}_R$), damping ($\underline{\underline{C}}_R$) and stiffness ($\underline{\underline{K}}_R$) matrices instead of the complete matrices of the advanced FE model. The objective is that the advanced FE and the NN-based reduced

models provide roughly equal displacement responses. The equation of motion of the reduced model is Equation 4.5.

$$(-\omega^2 \underline{\underline{M}}_R + i\omega \underline{\underline{C}}_R + \underline{\underline{K}}_R) \cdot \underline{\tilde{u}}_R(\omega) = \underline{\tilde{p}}_R(\omega) \tag{4.5}$$

Similar to the advanced FE model, the reduced displacement vector ($\underline{\tilde{u}}_R(\omega)$) contains the response in every degree of freedom to every frequency-dependent force ($\underline{\tilde{p}}_R(\omega)$) (see Eq. 4.6) but has fewer DoFs. In the end, the goal is to reduce the size of the matrices to a few DoFs to make the process more efficient (see Eq. 4.7).

$$\underline{\tilde{u}}_R(\omega) = \begin{bmatrix} \underline{\tilde{u}}_1(\omega) \\ \underline{\tilde{w}}_1(\omega) \\ \underline{\tilde{\varphi}}_1(\omega) \end{bmatrix} \quad \text{and} \quad \underline{\tilde{p}}_R(\omega) = \begin{bmatrix} \underline{\tilde{p}}_x(\omega) \\ \underline{\tilde{p}}_z(\omega) \\ \underline{\tilde{M}}_y(\omega) \end{bmatrix} \tag{4.6}$$

$$\text{size} \begin{pmatrix} \underline{\underline{M}} \\ \underline{\underline{C}} \\ \underline{\underline{K}} \end{pmatrix} >> \text{size} \begin{pmatrix} \underline{\underline{M}}_R \\ \underline{\underline{C}}_R \\ \underline{\underline{K}}_R \end{pmatrix} \tag{4.7}$$

Proposed method

The proposed model is built with a hybrid two-step approach based on Neural Networks and optimisation algorithms. The combination of these algorithms yields a reduced model that can solve any SSI problem depending on the dataset used for training, validating and testing the Neural Network.

The first step is to **generate data** for a specified SSI problem using an advanced FE model. It is noted here that if other alternatives to FE models exist for a specific problem, they are equally applicable and may be preferable from a computational point of view. However, an FE model is used herein as the most widely applicable and generic approach for the purpose of this thesis. This will provide a displacement vector such as the one in Equation 4.4 that can be introduced in the Neural Network. These displacements are the *targets* in the FNN that the algorithm trains upon and tries to approximate. Using a cyclic process in Python with a communication package between COMSOL and Python [28], the response of the configurations for specific SSI cases was obtained. The soil and foundation parameters that were swept through are the ones that represent the most relevant change in the results and will be outlined for each SSI problem in the subsections below. This part of the work demanded intense computational power and thoughtful decisions regarding the appropriate ranges for each parameter. The ranges were chosen to provide realistic values applicable to this engineering practice. In the case of the frequency, the range was limited to 0 to 50 Hz because this is more than sufficient for Earthquake Engineering practice. After all, earthquakes possess energy in much lower frequency ranges and this is the subject of interest [42] [49]. The shape of each foundation was also established considering adequate pragmatic implementations.

Once the dataset is generated, stored and arranged, it is introduced into the Neural Network. The latter takes the soil's and foundation's characteristics as inputs and finds the connection between those and the displacement. In short, the second step is to build and train the Neural Network to **obtain the predicted frequency response function**. This could be done after training, validating and testing the Neural Network to ensure the hyperparameters are chosen correctly.

This predicted response $\hat{w}$ (where the hat symbol represents that a value is predicted), obtained from the NN is introduced in an optimisation algorithm to retrieve the **reduced matrices**: $\underline{\underline{M}}_R$, $\underline{\underline{C}}_R$, $\underline{\underline{K}}_R$ by minimising the difference between the predicted and the optimised

displacements (see the objective function in Eq. 3.8). In the end, these matrices are the system matrices that can be appended to the superstructure matrices and solve the SSI problem and structural analysis in either frequency and, more importantly, time domain. The process is schematically described in Figure 4.2, and the details of the process are explained in Sections 4.2 and 4.3 for specific cases.

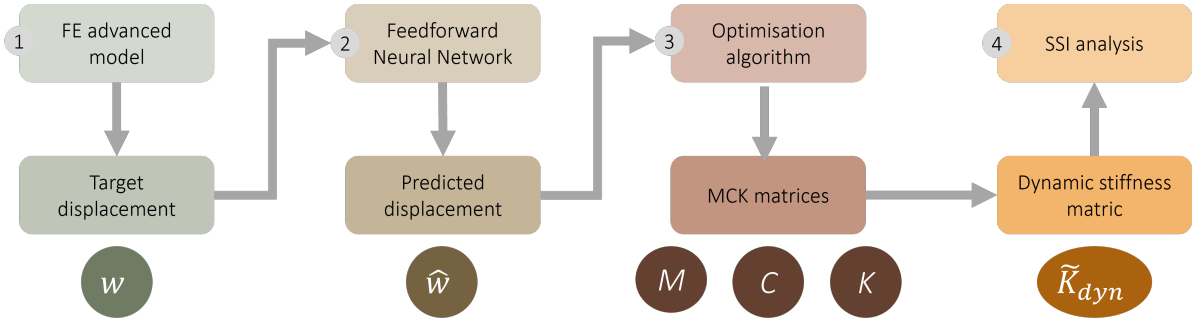


Figure 4.2: General approach of the proposed method.

4.2. Neural Network and optimisation approach for "Disc over homogeneous halfspace"

Solving a dynamic system to obtain a system's mass, stiffness and damping from the FRF required a two-step approach using NN and conventional optimisation techniques. The first step of the approach uses an NN schematically presented in Figure 4.3. The five input neurons in this NN are r_0, ν, ρ, G and ω. Aside from the frequency, these are the foundation's and soil's parameters used in the dataset generation because they are the main parameters that define the behaviour of this problem. These inputs, after scaling, are introduced as features to the NN. The complete feature dataset was an array of 1232 combinations per frequency. This represented a total of 61600 rows and 5 columns. The rows corresponded to all the combinations of the input parameters and the columns corresponded to each parameter (see Eq. 4.8).

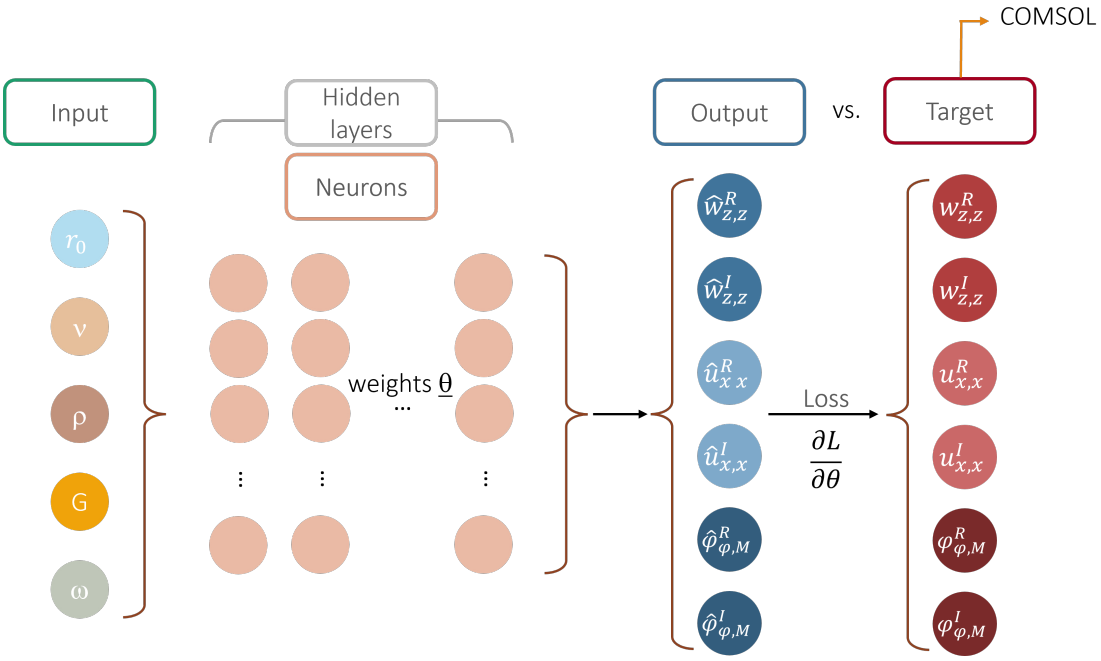


Figure 4.3: Neural Network architecture for Disc over homogeneous halfspace.

$$\begin{array}{c} \\ \text{Combination 1} \\ \text{Combination 2} \\ \text{Combination 3} \\ \cdots \\ \text{Combination 61600} \end{array} \begin{array}{c} \begin{array}{ccccc} r_0 & \nu & \rho & G & \omega \end{array} \\ \begin{bmatrix} 0.2 & 0.2 & 1400 & 2e7 & 6.283 \\ 0.2 & 0.2 & 1400 & 2e7 & 12.566 \\ 0.2 & 0.2 & 1400 & 2e7 & 18.849 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 5.2 & 0.495 & 2000 & 2.6e8 & 314.159 \end{bmatrix} \end{array} \tag{4.8}$$

The network's output was the predicted displacement of the system with specific r_0, ν, ρ, G characteristics for all three analysed motions for every frequency ω. This output is concurrently adjusted during the training process by comparing it with the targets, which are the points generated with the FE model (COMSOL) explained in Section 4.2.1. The output is 6 neurons. Each neuron corresponds to the real and imaginary responses of all 3 types of motions (vertical w, horizontal u and rotational φ). The output is stacked as shown in Equation 4.4. In this case, the coupled responses are equal to zero. Given that the system is symmetrical and the

disc is lying on the soil and not embedded, the response can considered decoupled, leaving a response vector as shown in 4.9.

$$\hat{w}(\omega) = \begin{bmatrix} \hat{u}^R_{x,x} & \hat{u}^I_{x,x} & \hat{w}^R_{z,z} & \hat{w}^I_{z,z} & \hat{\varphi}^R_{\varphi,M_y} & \hat{\varphi}^I_{\varphi,M_y} \end{bmatrix}^T \tag{4.9}$$

where $\hat{u}^R_{x,x}$ is the real and $\hat{u}^I_{x,x}$ the imaginary components of the horizontal predicted response due to a horizontal force; $\hat{w}^R_{z,z}$ is the real and $\hat{w}^I_{z,z}$ the imaginary components of the vertical predicted response due to a vertical force and $\hat{\varphi}^R_{\varphi,M_y}$ is the real and $\hat{\varphi}^I_{\varphi,M_y}$ the imaginary components of the rocking predicted response due to a rocking force. Each of these outputs corresponds to one frequency of analysis.

The architecture was varied in order to find the optimal combination by ranging through several neurons and the number of layers. Additionally, the weight decay (λ-value) was varied to avoid overfitting. Eventually, the optimal NN was defined.

As explained in Section 4.1, the predicted displacements were used as an input into the optimisation step (see Fig. 4.4). This step was performed with the Least-Squares method of the scipy library, as investigated in Section 3.2.1. In this step, the difference between the displacement obtained from the NN and the one approximated with the optimisation method should tend to be zero, as shown in Equation 3.8.

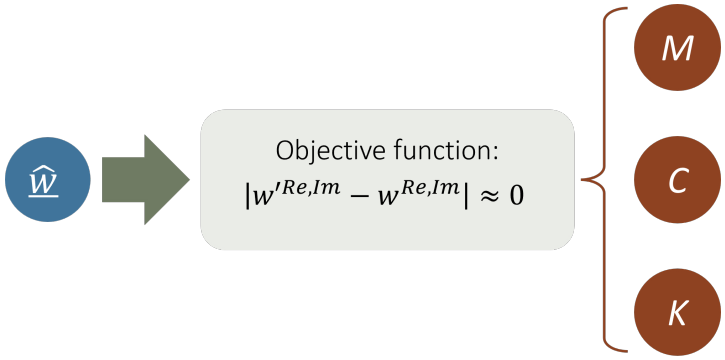


Figure 4.4: Optimisation process from predicted displacement to MCK matrices for Disc over homogeneous halfspace.

The optimisation algorithm was set up to optimise for one or 2DoFs, which can be determined by the user. In other words, the predicted displacement can be approximated using systems with an arbitrary number of degrees of freedom. An optimisation with more degrees of freedom may yield a more accurate response for responses with defined peaks or concave shapes. This leads to the final step of the process; obtaining the sought MCK matrices used to solve SSI systems in structural analysis.

4.2.1. Finite Element Method model

The first step in the application of this method is obtaining the dataset to feed the NN and train it. For this reason, an FE model was built to generate the data. However, the limitations of the FE models should also be taken into consideration. To begin with, the model was built with an already existing FEM software, which brings limitations in the adaptability of the built model. It was set up with a 2D-axisymmetric configuration using cylindrical coordinates (r, θ, z). This type of model allows the expression of 3D models in a 2D domain, reducing the computational expense. To ensure equivalency, simpler 2D-axisymmetric models were compared and validated with equivalent 3D models. In any case, this dimension reduction

requires particularly chosen conditions to assure comparability to 3D models. These will be outlined below.

Boundary conditions

The boundary conditions and configuration of the model were selected for optimal performance and are illustrated in Figure 4.5. The model must be symmetrical around an axis, necessitating a boundary condition that ensures axial symmetry (indicated in orange). Additionally, to accurately represent the soil's halfspace, the bottom and far-right boundaries must permit wave radiation and prevent reflections back into the domain (indicated in light blue). Furthermore, the top layer should be free of any prescribed stresses or displacements (marked in purple). Lastly, the loads were applied according to the case. For the vertical motion, a point load was applied in the negative z-direction on the symmetry axis. The horizontal load as distributed load over the area, referred to as boundary load. The rocking motion was considered with a ring load on the perimeter of the disc. All the loading conditions are marked in green on the figure although only the corresponding load was used for each case.

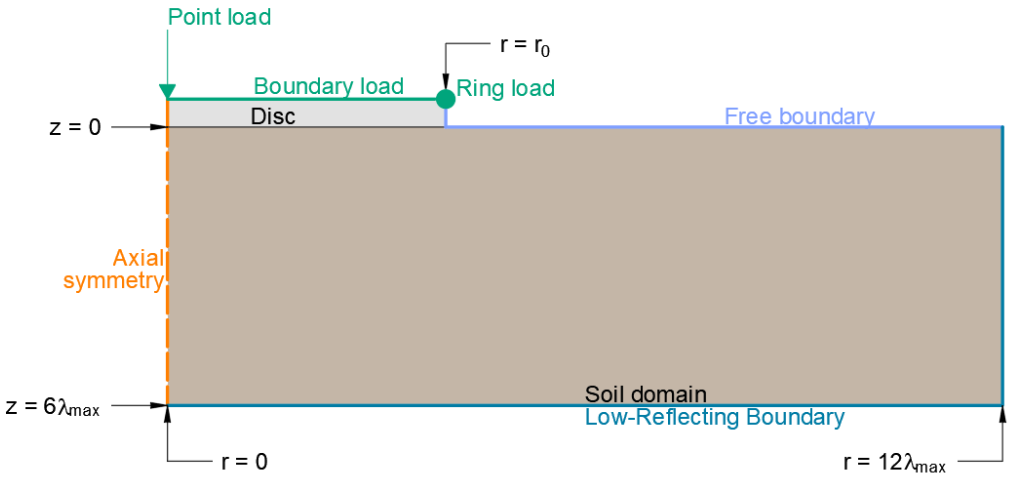


Figure 4.5: Boundary and loading conditions in FE model.

Soil domain

The soil domain's size had to be adjusted to ensure that the radiation boundary worked properly and that energy was not being reflected at the boundary back into the domain. If the domain size is not selected appropriately, it may fail to replicate the propagation of waves away from the foundation and towards infinity. Consequently, the boundary condition may not function properly, leading to wave reflections and potentially altering the results. The sensitivity analysis proved that using a domain of $12\lambda_{max}$ for the radial direction and $6\lambda_{max}$ in the z direction gave sufficiently accurate results. However, in some cases, especially when analysing the horizontal motion, larger domains were needed (up to $30\lambda_{max}$).

The wavelength relates the frequency of analysis with the shear wave velocity of a soil, and the conditions of the soil give the latter. This means that the domain's size is determined by the maximum wavelength given by the smallest frequency in the range of analysis (see Eq. 1.2).

Additionally, a hysteretic damping ratio ($\xi = 0.001$) was applied to the shear modulus of the soil. It was considered constant throughout the frequencies and was added to provide stability to the system. In the FE model, a damped shear modulus was used: $G_d = G(1 + 2i\xi)$.

Mesh size

The minimum and maximum sizes of the mesh were crucial to guarantee that the model grasped the whole information of all frequencies. The sensitivity analysis proved that using at least 8 elements and at most 20 elements per wavelength were enough to properly analyse the wave motion. In this case, the determining wavelength was the shortest, corresponding to the highest frequency in the range of analysis. This means that:

- $\text{mesh}_{\text{min}} = \lambda_{min}/20$
- $\text{mesh}_{\text{max}} = \lambda_{min}/8$

The sensitivity analysis confirmed that a mesh with sizes ranging between these values for every frequency range of analysis was enough to grasp the response of the problem at hand. Moreover, finer mesh led to higher computational expense without rendering more accurate results.

Application of the loading conditions

2D-axisymmetric models are, as indicated in their name, symmetrical. This means that motions in the vertical direction can be implemented directly. However, a horizontal displacement or motion implicates an asymmetrical behaviour, as it does not act on either a vertical, radial or circumferential axis but a mix of these. Therefore, the loading conditions for each case (vertical, horizontal and rocking) were applied giving specific values to the axial, radial and azimuthal directions in order to reconstruct the needed characteristics, especially for horizontal and rocking.

A horizontal load around a circular perimeter asks for vectors that, instead of acting in the radial or tangential direction, are adjusted at every point of the circumference to be in the same direction. Geometrically, this can be compounded by a cosine function in the radial direction and a negative sine function along θ considering the first horizontal circumferential mode. This way, adding both real components at, for example, $\theta = 0$, $p_{H_r} = \cos(0) = 1$ in the positive direction of r, so outwards and $p_{H_\theta} = \sin(0) = 0$ giving a resulting vector of 1. Given that the FEM software describes the loads as $e^{-im\theta}$ where m is the circumferential mode, the load can be reconstructed as shown below. Similarly, the rocking motion needs the first circumferential mode. In this case, to give a total moment of 1 in the disc, independently of its radius, the load must be reconstructed as shown below. It is also built up geometrically, considering a negative direction in z, $m = 1$ and a factor of $-2/r_0$ in order to ensure the desired total moment.

$$p_V = \begin{bmatrix} p_{V_r} \\ p_{V_z} \end{bmatrix} = \begin{bmatrix} 0 \\ -1 \end{bmatrix} \qquad p_H = \begin{bmatrix} p_{H_r} \\ p_{H_\theta} \\ p_{H_z} \end{bmatrix} = \begin{bmatrix} 1 \\ -i \\ 0 \end{bmatrix} \qquad p_R = \begin{bmatrix} p_{R_r} \\ p_{R_\theta} \\ p_{R_z} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -2/r_0 \end{bmatrix}$$

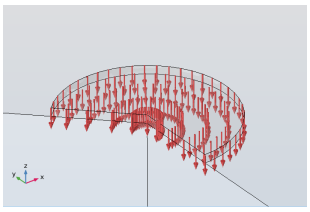


Figure 4.6: Vertical load.

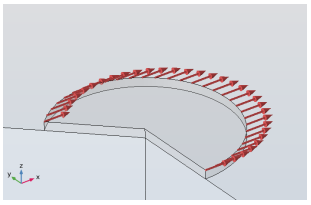


Figure 4.7: Horizontal load.

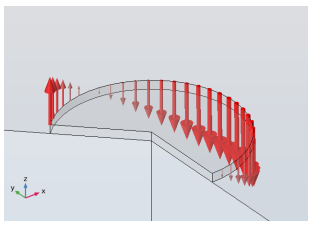


Figure 4.8: Rocking load.

The vertical load was actually applied as a point load at the axis and the horizontal as a boundary load, and Figures 4.9 and 4.11 are used merely for graphical representation. On the other hand, the rocking load was applied as a ring load, as this one allowed the reconstruction of a unit moment for the rocking analysis. Finally, the horizontal and rocking cases were done using the first circumferential mode with horizontal and vertical loads, respectively. This allowed the geometrical construction of the loads as seen in the images with the characteristics in the vectors above them.

4.2.2. Comparison of analogue models and FE model

Recent advancements in computational capacity have significantly enhanced the accuracy of engineering analysis. In the field of soil-structure interaction, larger domains and finer meshes can be modelled thus facilitating more accurate results. The increasing computational expenses can now be withstood by commercial computers that were unavailable a few decades ago. Understanding the applicability of the analogues can be done nowadays by comparing the results of each motion to the ones obtained from an FE model. All the models for each case are contrasted below, including the relative error calculated between the FE model and each of the analogues. The parameters used for this analysis are shown in Table 4.1.

Radius	$r_0 = 0.5$ [m]
Shear modulus	$G = 2.59 \cdot 10^7$ [Pa]
Density	$\rho = 1800$ [kg/m^3]
Poisson's ratio	$\nu = 0.33$ [-]
Damping ratio	$\xi = 0.001$ [-]

Table 4.1: Parameters of the Disc models used for analogues comparison.

Vertical motion

Figure 4.9 shows that this motion is well captured by the analogues in comparison with the FE model. It seems that it is Lysmer's or Pais-Kausel's models that follow the results from the FE model better. The average error of Lysmer's model is the lowest, followed by Pais-Kausel's, with a value of $\bar{\varepsilon} = 0.024$ and $\bar{\varepsilon} = 0.025$ respectively, as can be seen in Figure 4.10. It can also be observed that the variation increases for higher frequencies.

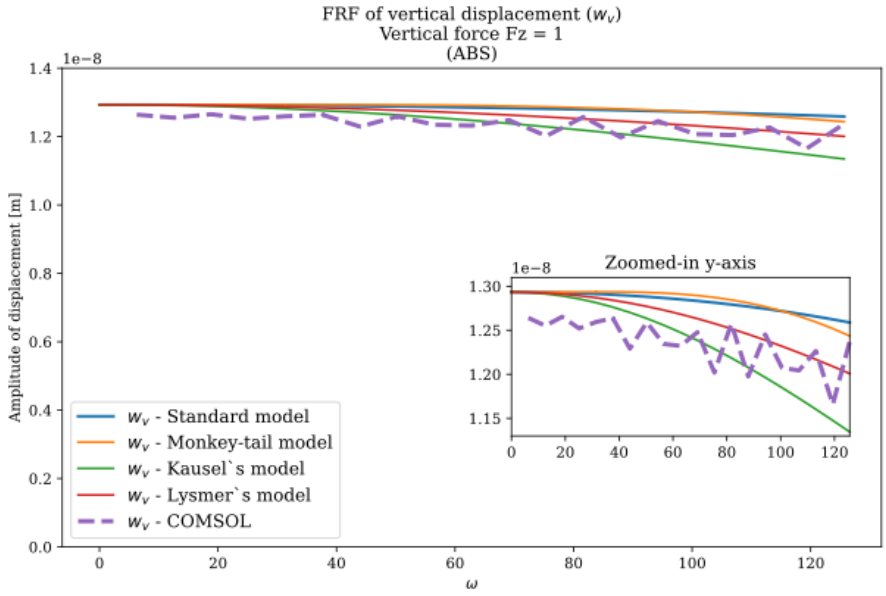


Figure 4.9: Comparison of the analogue models and the FE model for vertical motion.

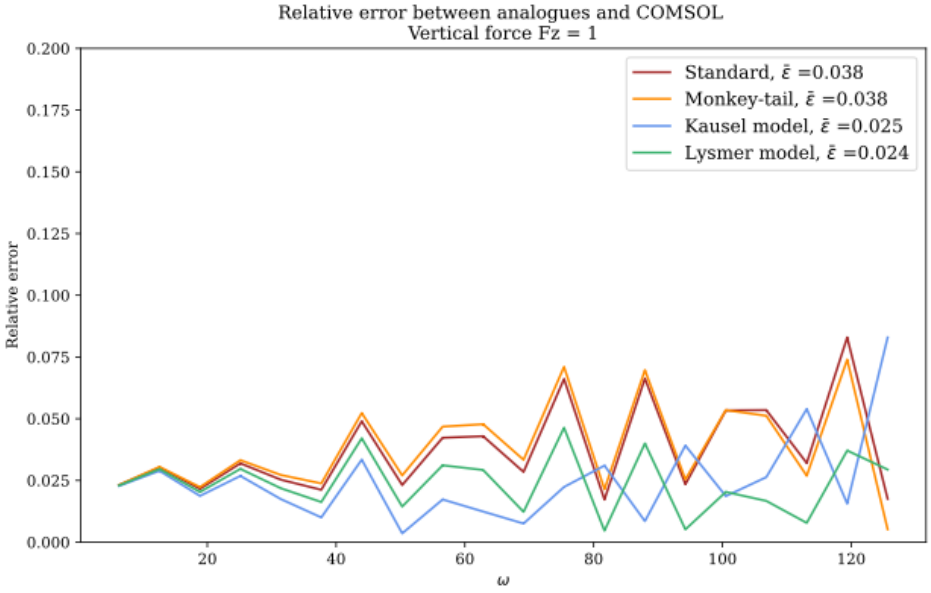


Figure 4.10: Relative error of the mechanical analogues with respect to the FE model for the vertical motion.

Horizontal motion

Figure 4.11 depicts that the analogues are more accurate for the horizontal case than for the vertical case. However, the horizontal loading is captured better by Wolf's standard model, with an error of $\bar{\varepsilon} = 0.007$, compared to the other two models with an average error of $\bar{\varepsilon} = 0.010$. The relative error for every frequency can be observed in Figure 4.12.

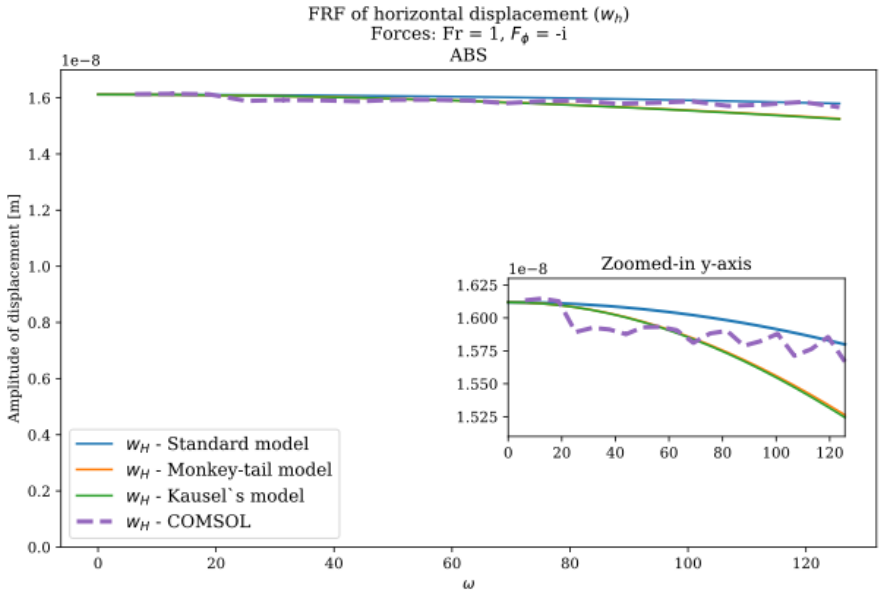


Figure 4.11: Comparison of the analogue models and the FE model for horizontal motion.

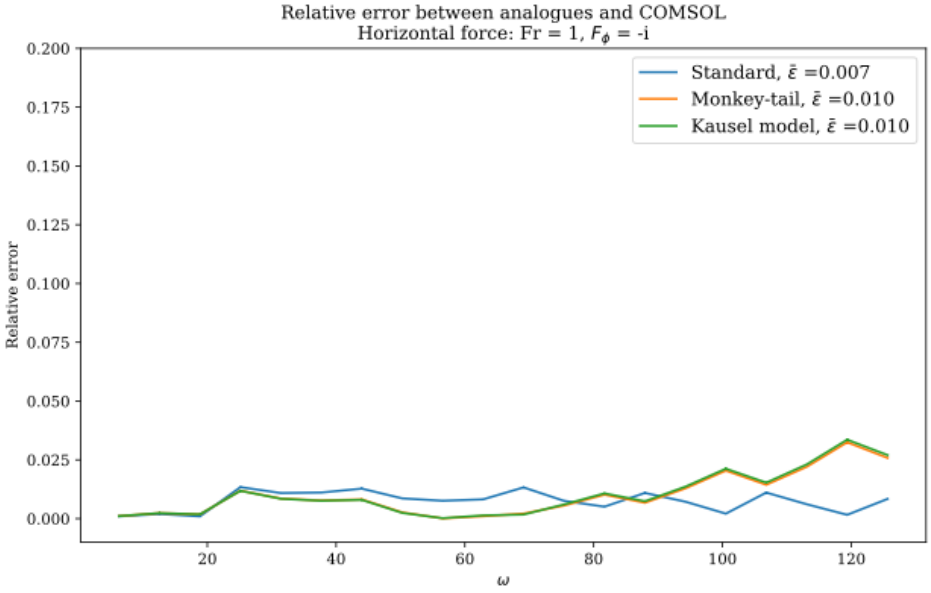


Figure 4.12: Relative error of the mechanical analogues with respect to the FE model for the horizontal motion.

Rocking motion

The rocking motion presents a greater offset between the analogues and the FE model, as can be observed in Figure 4.13. The best-suited analogues appear to be Pais-Kausel's with an average error of $\bar{\varepsilon} = 0.033$ closely followed by Wolf's Monkey-tail model, as compared to Wolf's Standard model with an average error of $\bar{\varepsilon} = 0.043$ when analysing the relative errors depicted in Figure 4.14

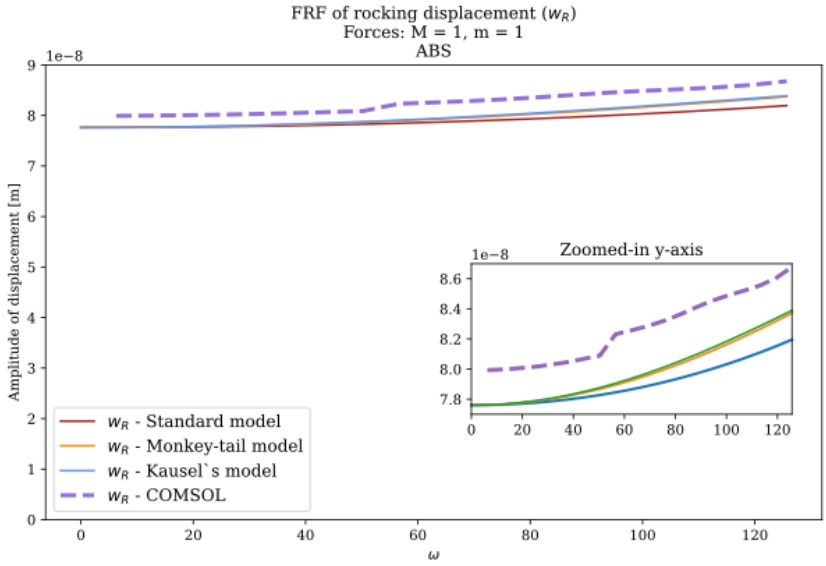


Figure 4.13: Comparison of the analogue models and the FE model for rocking motion.

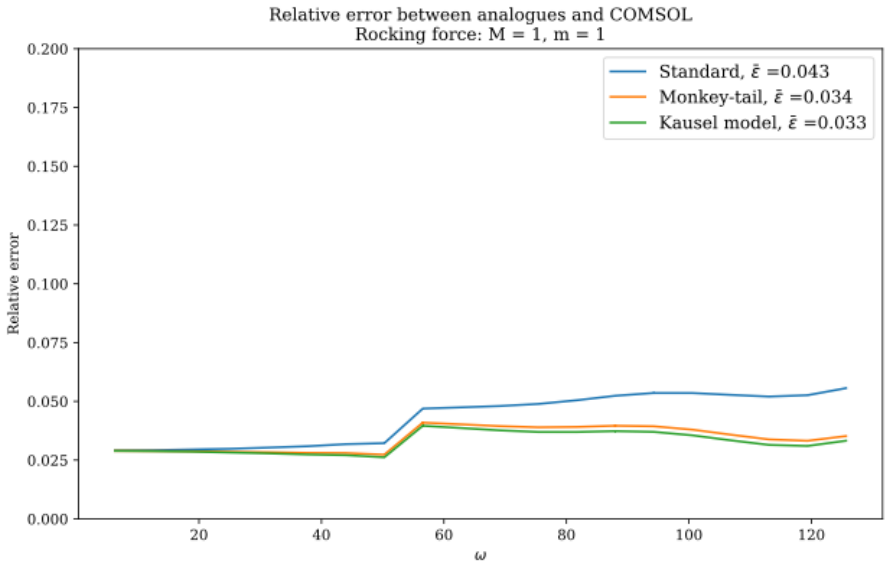


Figure 4.14: Relative error of the mechanical analogues with respect to the FE model for the rocking motion.

It can be observed in the graphs above that the vertical and horizontal models are close and applicable to the chosen characteristics of the disc and the soil, while the rocking motion shows a greater error. However, all motions show an error significantly below 5%, which is acceptable from an engineering perspective.

This comparison between models confirms that this is a good benchmark problem for the study of a computational approach. Moreover, it retroactively indicates two things. On one hand, the mechanical analogues developed by Lysmer, Pais-Kausel and Wolf are significantly accurate for this SSI problem. It confirms that a disc over a homogeneous soil can be represented as an SDoF system successfully. On the other hand, this indicates that the FE model, assuming that the mechanical analogues had previously been validated and could be accepted as correct, was built correctly and can be used for the next steps of the work.

4.2.3. Generation of input data

The recurrent validation of the analogues and the FE model allowed the characteristics of the model to be considered adequate to generate the dataset for the NN. Given that the foundation parameter of interest in this case is the radius of the disc, its variation will give sufficient insight into the behaviour of this type of substructure in terms of geometry. For the soil, the Poisson's ratio, the density of the soil and the G-modulus will be varied in different combinations to obtain a wide range of results. For each combination, a model will be set up and run for frequencies from 1 to 50 Hz with a step of 1 Hz. The ranges for each variable are presented in Table 4.2.

Parameters		Amount
$r_0 =$	$[0.2, 0.7, 1.2, 1.7, 2.2, 2.7, 3.2, 3.7, 4.2, 4.7, 5.2]$ [m]	11
$\nu_{soil} =$	$[0.2, 0.3, 0.4, 0.495]$ [-]	4
$\rho_{soil} =$	$[1400, 1600, 1800, 2000]$ [kg/m^3]	4
$G_{soil} =$	[2.0e7, 6.0e7, 1.0e8, 1.4e8, 1.8e8, 2.2e8, 2.6e8] [Pa]	7
	Total number of combinations	1232
	Total number of analyses per motion (50 frequencies)	61600

Table 4.2: Ranges of values for each parameter for each type of motion for the dataset generation of Disc over homogeneous halfspace.

The generation of the dataset for vertical motion took approximately 10 hours of 7 different processes running in parallel. On the other hand, the horizontal motion took around 60 hours for 10 parallel processes and the rocking motion around 26 hours for 7 processes. This allowed to generate the points to feed the Neural Network and obtain the generic solution for this type of SSI problem. Figure 4.15 shows the COMSOL results of a 0.2 m radius disc for all soil combinations under a vertical load. Figures 4.16 and 4.17 show the results for the same radius in horizontal and rocking motions, following the same colour, linestyle and linewidth code.

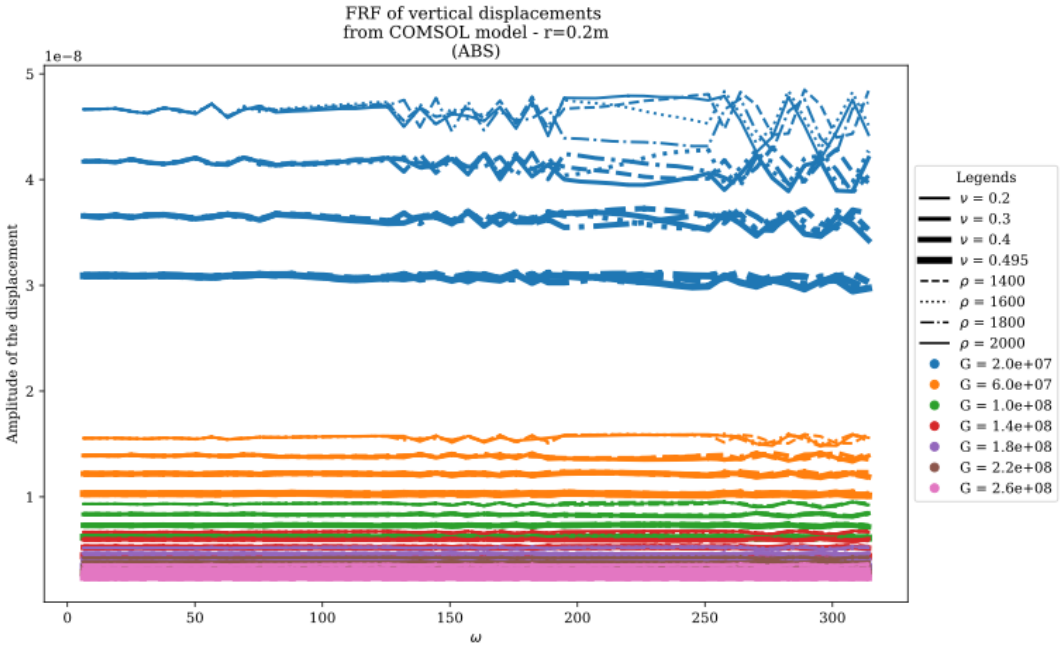


Figure 4.15: Results of vertical motion for all soil combinations with $r_0 = 0.2$ m.

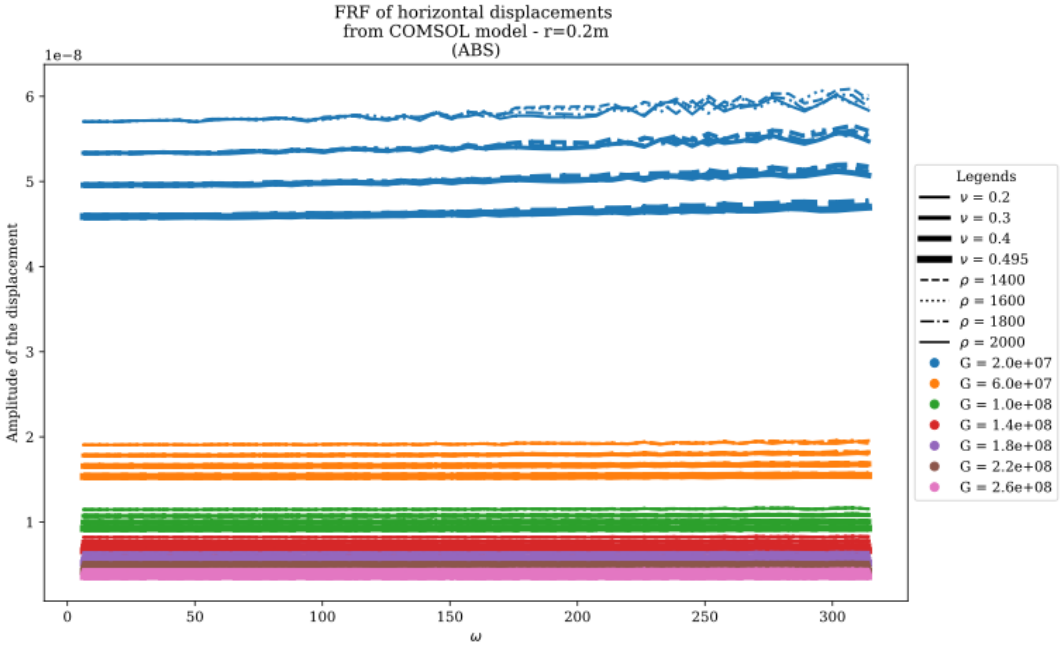


Figure 4.16: Results of horizontal motion for all soil combinations with $r_0 = 0.2$ m.

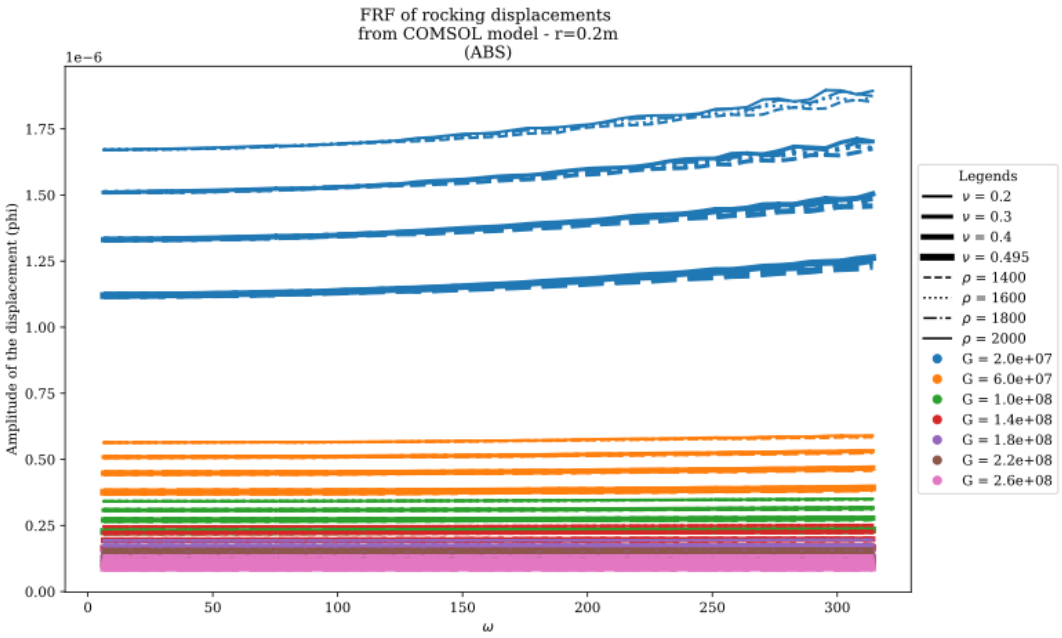


Figure 4.17: Results of rocking motion for all soil combinations with $r_0 = 0.2$ m.

It can be observed from these graphs (Figs. 4.15, 4.16, 4.17) that the shape of the results is similar among the analysed motions. The order of magnitude for both vertical and horizontal is 10^{-8}. Contrastingly, the rocking motion has results in the order of magnitude of 10^{-6}. However, it must be noted that the degree of freedom for the first two is translation and is measured in meters, while the degree of freedom of the rocking motion is rotational (φ - see Fig. 2.7) and it is measured in radians.

It is perceived from these graphs that softer soil, i.e. with lower shear modulus, gives a greater displacement than stiffer soils. Additionally, this is the characteristic that determines the behaviour the most, as the displacement can be grouped in defined regions established by the shear modulus, marked with the color. The second most influencing parameter is Poisson's ratio. Within the responses for the same shear modulus, the next big differences are observed in the thickness of the line, which indicates the Poisson's ratio of said case.

Firstly, the blue lines, corresponding to soft soil, clearly mark greater displacements than the rest (see Fig. 4.15). Secondly, the thinnest line distinctly presents greater displacements than the thicker lines. This means that a lower Poisson's ratio relates to bigger deformations, while a higher Poisson's ratio resists them better. This is congruent to the effect of this soil characteristic. For instance, a soil with a high ν would be less compressible. Therefore it is logical to find greater deformations with more compressible soils.

Ultimately, the density of the soil does not have a significant effect on the behaviour of the soil. It can be observed in all figures that the different linestyles present practically the same displacements for the same shear modulus and Poisson's ratio. This indicates that its role in this SSI problem is less critical than that of G and ν. This was to some degree expected because the range of analysis for ρ is not as wide as for G, both in this study and generally, so the effect of the latter is foreseen to be indeed greater.

These results, and the ones generated for all the other radii in the range of analysis, were used in the definition of the optimal Neural Network for the proposed approach. The results for vertical, horizontal and rocking motions for other radii can be observed in Appendix A.

4.2.4. Sensitivity analysis for the Neural Network and optimisation approach

The method's applicability was defined by comparing the targeted and predicted FRFs of the NN, and the FRF produced by the reduced M, C, and K obtained from the optimisation process. Firstly, the optimal NN must be defined. To do so, and based on the findings of the preliminary analysed problems, different combinations of hyperparameters were evaluated. The hyperparameters are presented in Table 4.3. Each proposed NN was run for 300 epochs to define the best architecture. The *adam* solver and ReLU activation function were tested.

Model hyperparameters	
Number of hidden layers	$[3, 4]$
Number of neurons	$[16, 32, 64, 128]$
Training choices	
Solver	*adam*
Activation function	*tanh*, ReLU
Weight-decay (λ)	[1e-9, 1e-8, 1e-7, 1e-6, 1e-5, 1e-4, 1e-3, 1e-2, 1e-1, 1, 10]

Table 4.3: Model hyperparameters and training choices for finding the optimal NN for Disc over homogeneous halfspace.

The process to choose the best NN consisted of varying the model hyperparameters and building the NN with the architecture resulting from different combinations, in total 13 NN. The lowest validation loss determined the ideal number of layers and neurons from the trial set. Firstly, the effect of the activation function was evaluated. Figure 4.18 shows the decrease of the validation loss per epoch in 3-layer NN with a different number of neurons in each layer. It can be observed that the ReLU activation function reaches a lower validation error with fewer epochs than *tanh* when comparing the same layer size.

This indicates that ReLU is more efficient for this problem. Additionally, for most applications, ReLU activation function is usually more robust [21], as it is computationally effective and well-suited for problems with large data. It is preferred because it has been observed that it converges faster, is computationally efficient, provides a more stable backpropagation and has been empirically proven to address most ML problems successfully [10]. Therefore only ReLU will be considered for further analysis.

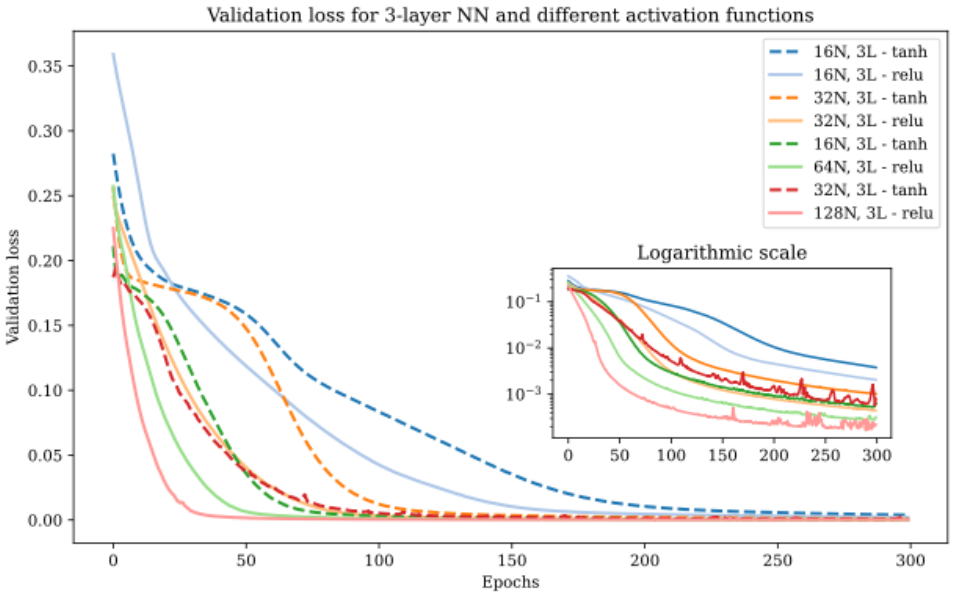


Figure 4.18: ReLU vs. tanh activation functions in 3-layer NN with different number of neurons per layer.

After defining the activation function, 4-layer Neural Networks were also evaluated. The number of neurons was again varied to compare the behaviour of the validation loss with each epoch for the different architectures. Figure 4.19 shows these results. It can be seen that the parameter that affects the most in the results is the number of neurons and not the number of layers. A larger number of layers does not yield a faster decrease in the validation loss, which means this case does not require a very deep NN. Consequently, a 3-layer NN will be considered from now on.

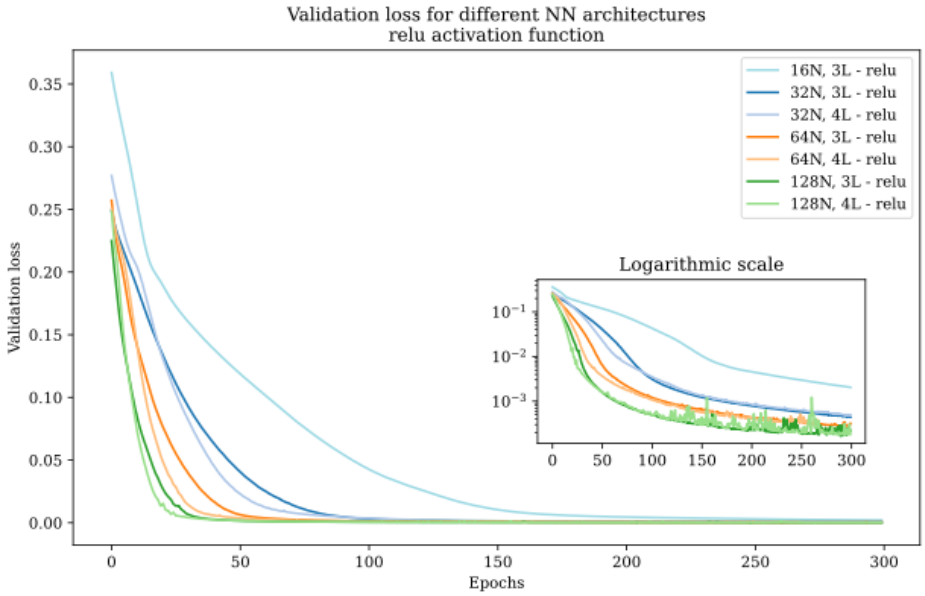


Figure 4.19: Validation loss of NN with ReLU activation function and varying number of layers and neurons.

Finally, to define the best architecture, the minimum validation losses of each 3-layer NN must be considered. The architecture with the smallest minimum validation loss can be determined as the optimal architecture. Table 4.4 shows these results. It can be observed that

a 3-layer 128-neuron NN has the lowest historical validation error. This was chosen as the optimal architecture and will be considered for further analysis.

Number of neurons	Minimum validation loss
16	2.0460e-3
32	4.4179e-4
64	2.8087e-4
128	1.6647e-4

Table 4.4: Minimum validation losses for different number of neurons in a 3-layer NN.

Overfitting models cannot predict accurate solutions for unseen sets, meaning they can only predict for the sets they were trained on. This is not the desired outcome in ML because the objective is to predict the response of new systems. Therefore, the model was run for the different λ-values in Table 4.3 for 1200 epochs. With this, it was possible to determine the weight-decay (λ) that yielded the best model, which does not under- or overfit. Figure 4.20 shows the validation loss after 1200 epochs against the different evaluated λ-values.

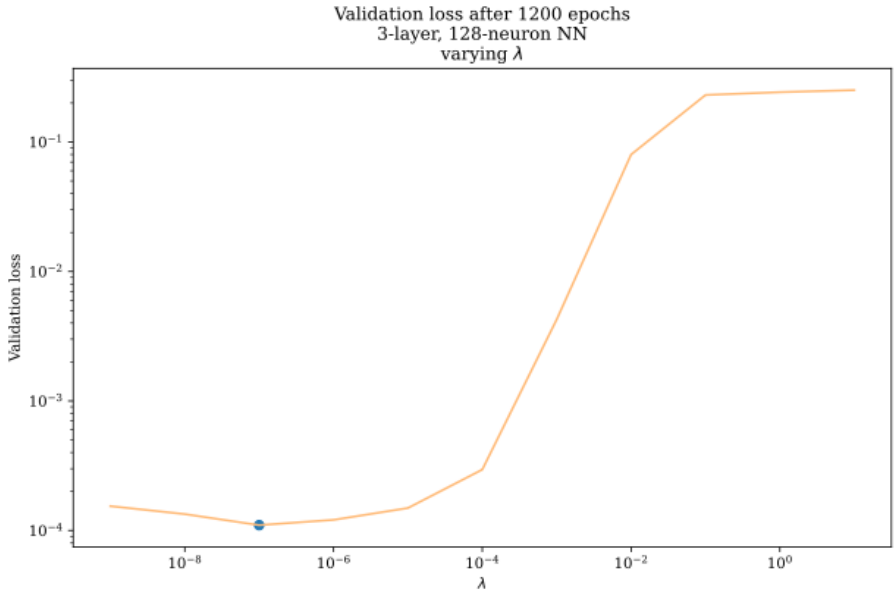


Figure 4.20: Validation loss of 3-layer, 128-neuron NN with ReLU activation function for different λ-values after 1200 epochs.

This figure shows that the NN requires a weight-decay of $\lambda = 1e-7$ in the *adam* optimizer to perform adequately. In general, in a curve of validation loss against λ, the lowest point shows the optimal model. From there to the left, the model would underfit; and from there to the right, the model would overfit. With the lowest values given for λ in this case, the validation loss does not increase significantly to the left of the lowest point. This is attributed to the large size of the available dataset. However, higher values of λ do present overfitting, as the validation loss increases for λ greater than 1e-7. Conclusively, this graph indicates that the optimal model is obtained using $\lambda = 1e-7$. This weight-decay hyperparameter will be the one to avoid overfitting and underfitting.

Optimal NN

After the parametric study, the optimal NN could be defined. The best Neural Network for this problem resulted from the hyperparameters in Table 4.5.

Number of hidden layers	3
Number of neurons	128
Activation function	ReLU
Weight-decay (λ)	1e-7

Table 4.5: Optimal hyperparameters for Disc of homogeneous halfspace.

Figures 4.21, 4.23 and 4.25 show the target values from the testing set and the values predicted by the trained NN for vertical, horizontal and rocking motions respectively. It can be observed that the results align with accuracy. Moreover, Figure 4.27 shows only a few points of the scaled results of all three motions to perceive more detail. Once again, it is clear that the NN predicts correctly. Finally, Figures 4.22, 4.24 and 4.26 show a zoomed-in plot of the lower regions of the results for all the motions, marked with an orange line to the left in the previous plots.

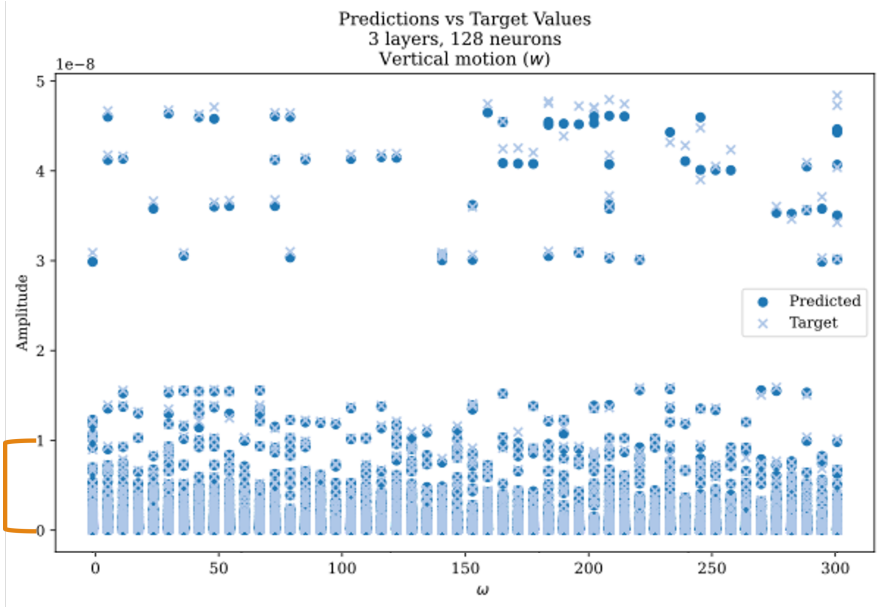


Figure 4.21: Target and predicted FRF from trained NN - Vertical motion.

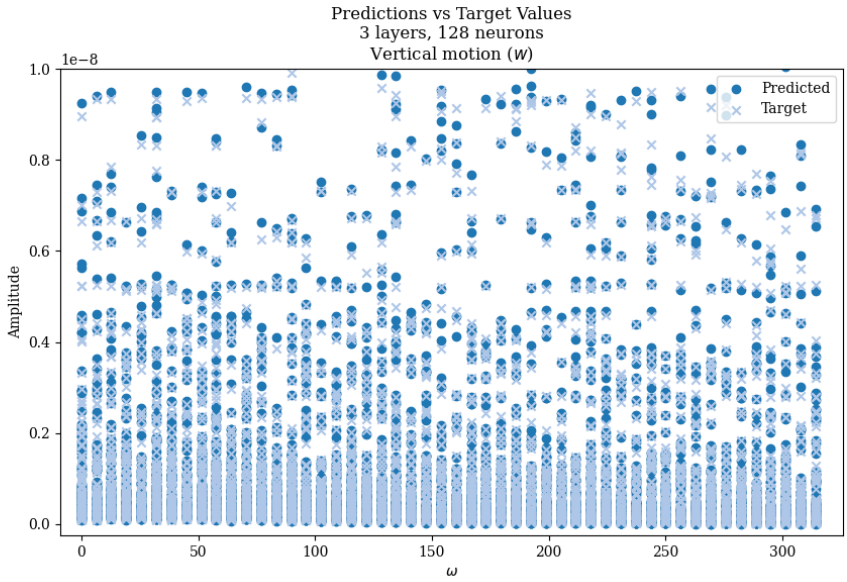


Figure 4.22: Target and predicted FRF from trained NN (zoomed in) - Vertical motion.

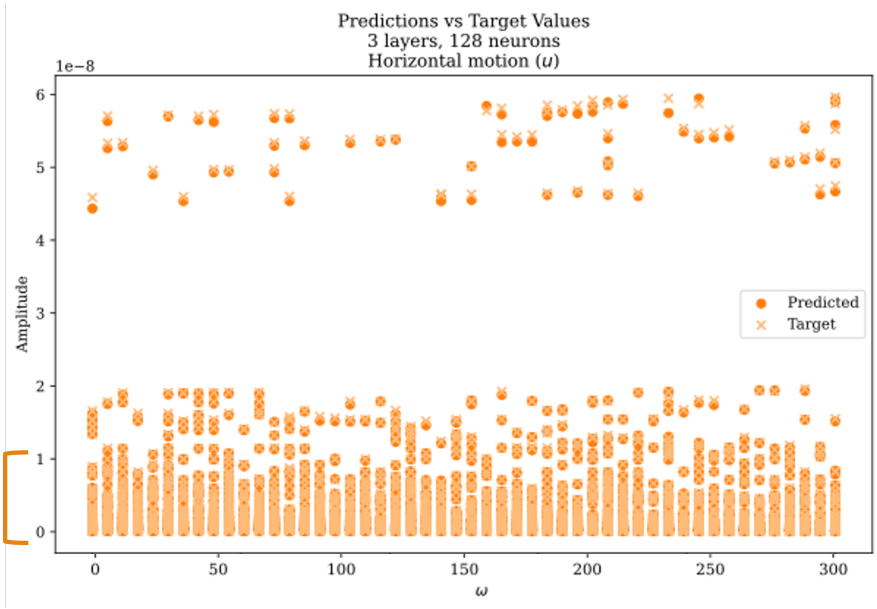


Figure 4.23: Target and predicted FRF from trained NN - Horizontal motion.

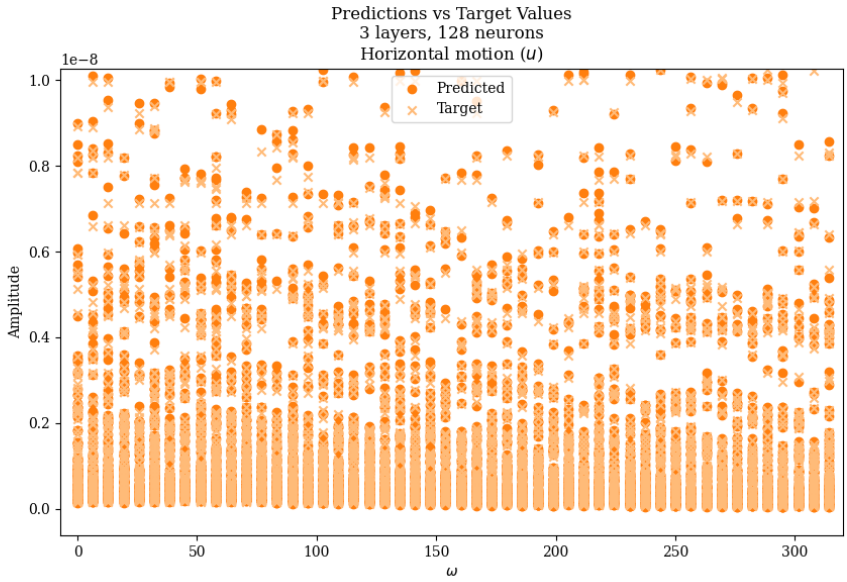


Figure 4.24: Target and predicted FRF from trained NN (zoomed-in) - Horizontal motion.

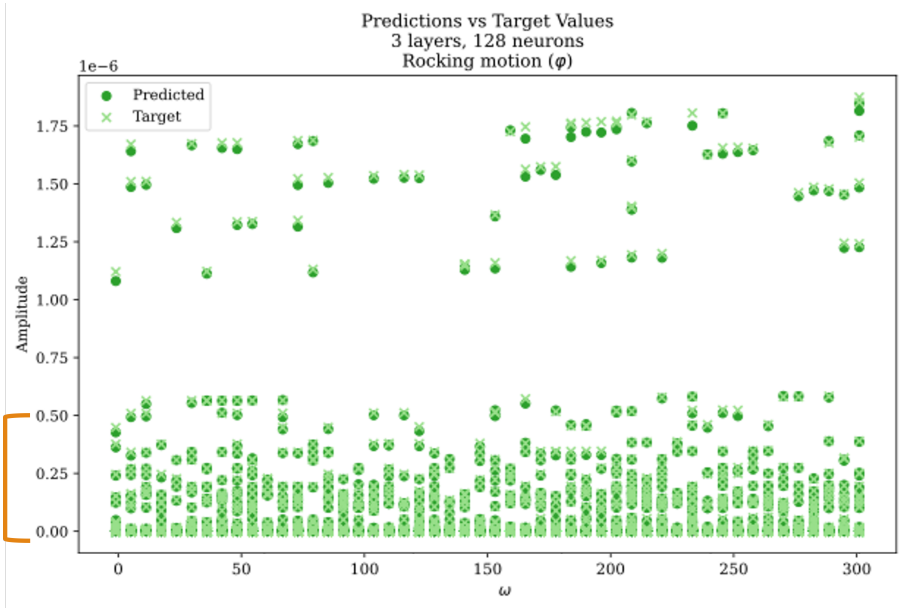


Figure 4.25: Target and predicted FRF from trained NN - Rocking motion.

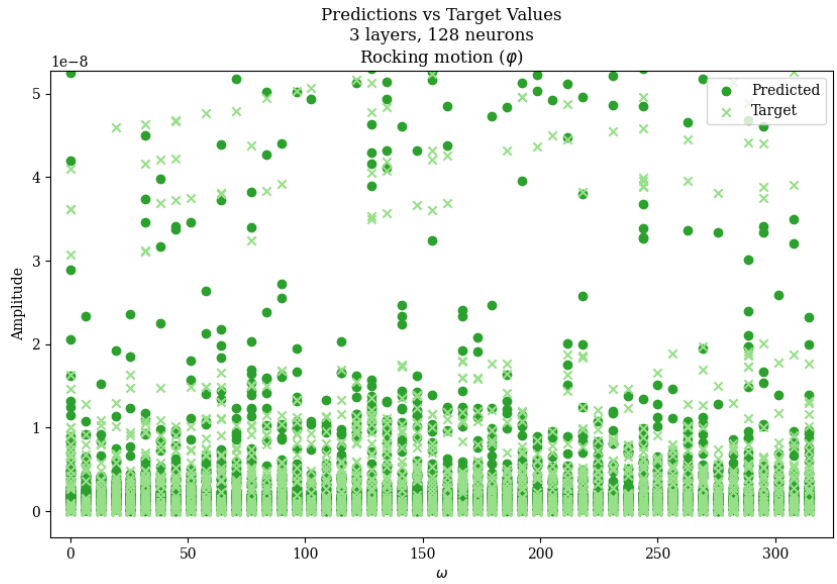


Figure 4.26: Target and predicted FRF from trained NN (zoomed-in) - Rocking motion.

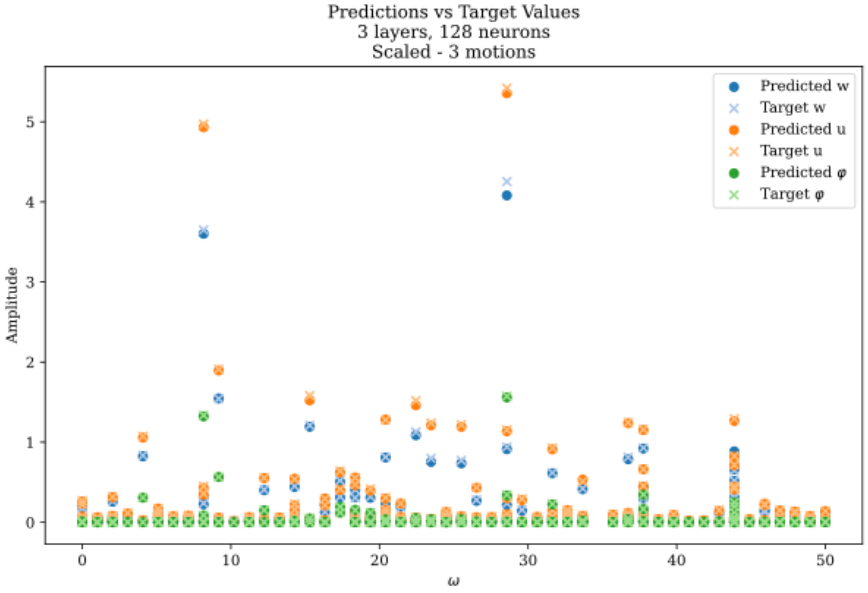


Figure 4.27: Target and predicted FRF from trained NN - All motions, scaled.

Optimisation

After predicting the FRF with the Neural Network, it is possible to introduce it into the optimisation process and obtain the system's mass, stiffness and damping matrices for later SSI analysis. The algorithm tries to approximate the values of the mass, stiffness and damping by comparing the displacements. To do so, the correct testing inputs have to be introduced in the NN.

Until this point, the NN was trained, validated and tested on individual points corresponding to a soil-foundation case for a specific frequency ω. However, to fully evaluate a particular system, the testing features must correspond to an SSI system in all the frequencies. For such reason, an array that contains the r_0, ν, ρ and G for the system and all the frequencies of

analysis must be input to evaluate a system and obtain the complete FRF. One important advantage of this NN architecture that considers ω as an input, is that any frequency, and not only the ones used in the training process, can be evaluated. An example of an input array for testing is shown in the matrix below (Eq. 4.10).

$$\begin{bmatrix} r_0 & \nu & \rho & G & \omega \\ 0.35 & 0.25 & 1500 & 2.3e6 & 0 \\ 0.35 & 0.25 & 1500 & 2.3e6 & 3.1416 \\ 0.35 & 0.25 & 1500 & 2.3e6 & 6.2832 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0.35 & 0.25 & 1500 & 2.3e6 & \omega_n \end{bmatrix} \tag{4.10}$$

1DoF optimisation Figures 4.28, 4.29 and 4.30 show the target, predicted and optimised FRF (vertical, horizontal and rocking, respectively). The first one corresponds to the information generated with the FE model. The second one is the predicted response from the test dataset created for a specific case. The third one is the optimised FRF calculated using the mass, stiffness and damping coefficients obtained in the optimisation process.

The comparison between the target and predicted FRF indicates that, although the results are close, the accuracy of the NN could still be improved. The 1DoF optimisation process proves its applicability by delivering the system's matrix such that it gives a response that follows the predicted FRF closely. However, for other soil and foundation characteristics, the prediction and optimisation are not as precise and indicate there is room for finer tuning of both processes. The comparison between the real and imaginary parts of the target, predicted and optimised FRF are shown in Appendix B. Other cases are also included.

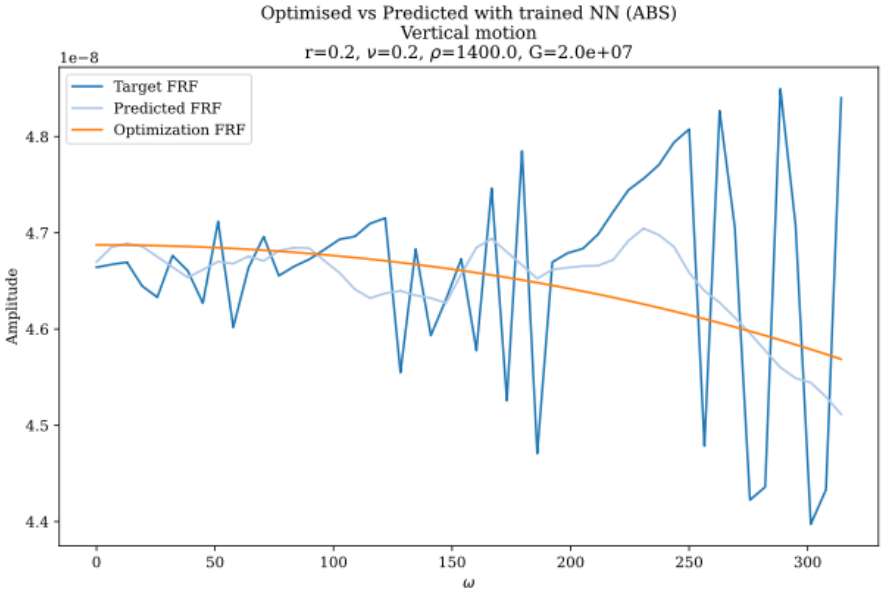


Figure 4.28: Comparison of absolute FRF of vertical motion with 1DoF optimisation.

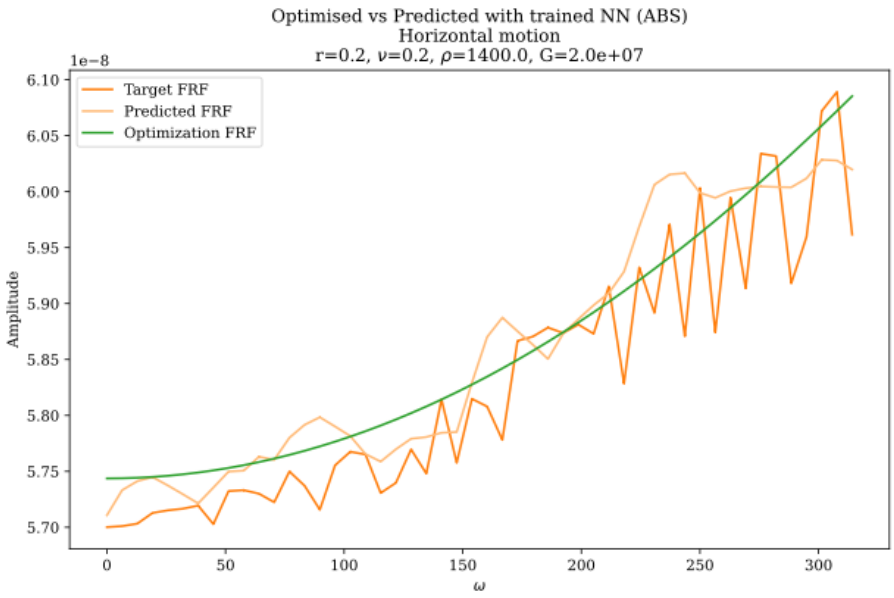


Figure 4.29: Comparison of absolute FRF of horizontal motion with 1DoF optimisation.

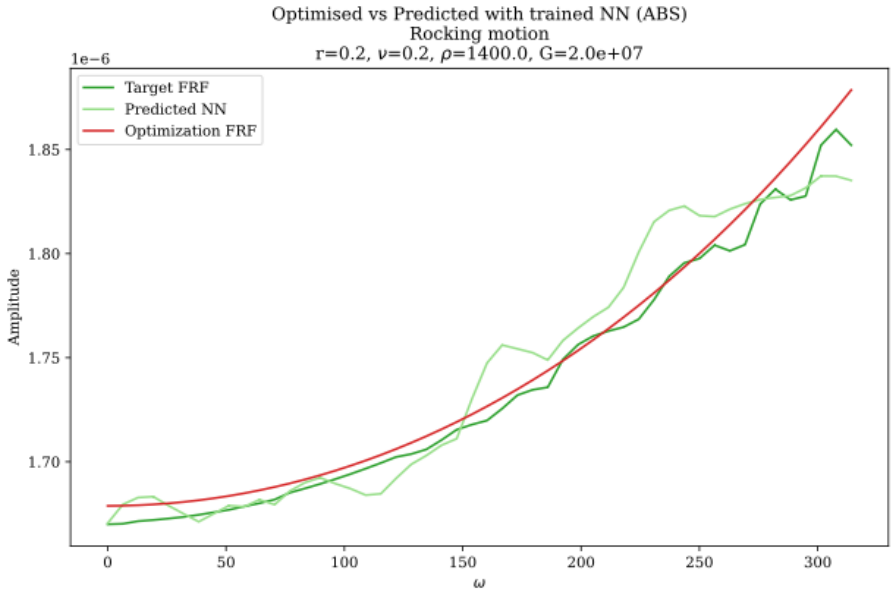


Figure 4.30: Comparison of absolute FRF of rocking motion with 1DoF optimisation.

2DoF optimisation Instead of obtaining only one value for mass (M), one for stiffness (K) and one for damping (C), it can be upgraded to more DoFs if needed. This means that the result for each coefficient is now a NxN matrix. In other words, the result is three (for M, C, and K) NxN matrices that provide the FRF of the system. Figure 4.31 shows the optimised FRF obtained with a 2DoF optimisation matrix for the vertical motion.

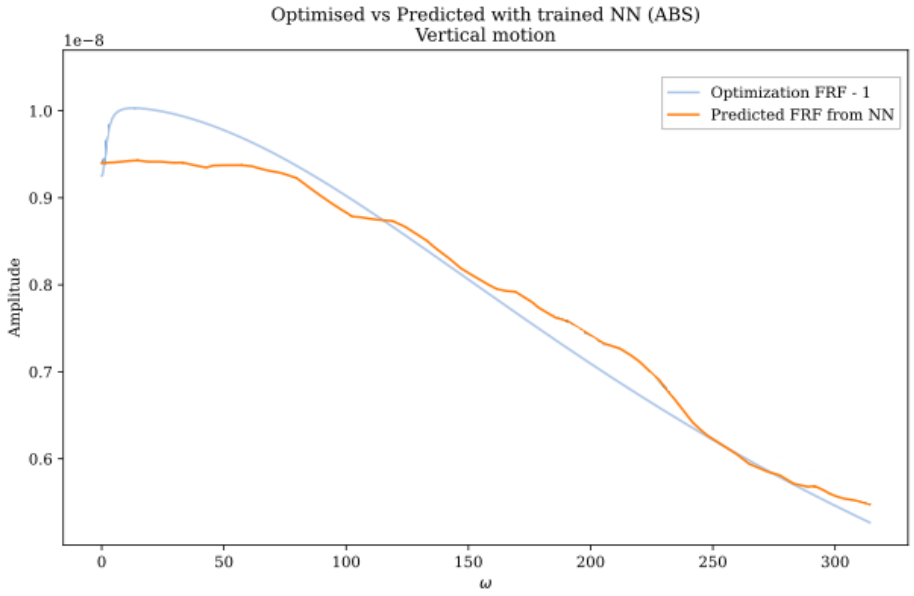


Figure 4.31: Comparison of absolute FRF of vertical motion with 2DoF optimisation.

For the vertical motion, the 2DoF optimisation results in an FRF close to the predicted one. In this case, the approach seems applicable. However, for the horizontal and rocking responses, the optimised FRFs do not follow the predicted FRFs closely and the results are not included in this report. This indicates that the optimisation process is not fully robust for more than 1 DoF and in certain cases leads to erroneous results. However, given that the NN predicts correctly, finer tuning in the optimisation process for higher DoFs could provide a better analogue and is an interesting point for future works.

4.3. Neural Network and optimisation approach for "Pile in soil under axial vibrations"

The method was applied to a pile embedded in soil under vertical motion to demonstrate that the approach can be used for other foundations. This case presents a significantly contrasting behaviour to that of the disc over soil. The main difference lies in the geometry of the foundation, as the length of the pile is an important characteristic. Additionally, the embedment has a significant effect on the response as the soil affects the foundation not only from beneath but also along the sides. Consequently, an extra input neuron in the NN is included to account for the length of the pile. A scheme of the structure of this NN is shown in Figure 4.32.

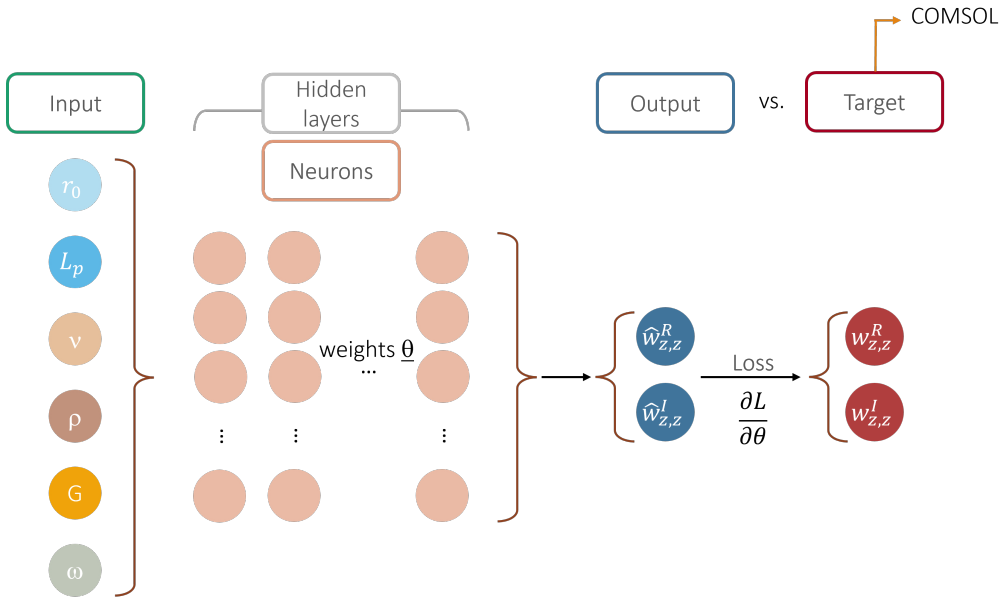


Figure 4.32: Neural Network architecture for Pile embedded in soil.

The input array for this case contains the combination of unique pairs of radius and length defined for the pile and the same soil parameters as in the previous case. Each of these combinations is analysed per frequency, resulting in an array such as the one shown in Equation 4.11:

$$\begin{array}{c} \\ \text{Combination 1} \\ \text{Combination 2} \\ \text{Combination 3} \\ \text{...} \\ \text{Combination 78400} \end{array} \begin{array}{c} \begin{array}{cccccc} r_0 & L & \nu & \rho & G & \omega \end{array} \\ \begin{bmatrix} 0.25 & 4 & 0.2 & 1400 & 2e7 & 6.283 \\ 0.25 & 4 & 0.2 & 1400 & 2e7 & 12.566 \\ 0.25 & 4 & 0.2 & 1400 & 2e7 & 18.849 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 2 & 50 & 0.495 & 2000 & 2.6e8 & 314.159 \end{bmatrix} \end{array} \tag{4.11}$$

where r_0 and L are the radius and the length of the pile, and ν, ρ and G are the soil parameters as in the previous case. ω is the frequency.

In this thesis, only the vertical motion of the pile will be analysed. For that reason, the NN has two output neurons: the real $\hat{w}^R_{z,z}$ and imaginary $\hat{w}^I_{z,z}$ components of the vertical response. In the end, the response vector is as in Equation 4.12 for each frequency:

$$\hat{w}(\omega) = \left[\hat{w}^R_{z,z}, \hat{w}^I_{z,z}\right]^T \tag{4.12}$$

Following the same procedure as for the disc over soil, an FE model with its specific characteristics was built to generate the training dataset to be introduced in the NN. Then the architecture of the NN was varied to obtain the optimal option. However, making use of the acquired knowledge from the previous case, fewer hyperparameters will be considered in the process. For example, the *tanh* activation function will not be looked into. Finally, the optimisation process was used to retrieve the mass, stiffness and damping matrices of the system for the full SSI analysis.

4.3.1. Finite Element Method model

This type of foundation was also modelled using the FEM software COMSOL. The scheme is shown in Figure 4.33. This model is built following the same steps as outlined for the "Disc over homogeneous halfspace" in Section 2.2. It is also built as axisymmetric, which means that the pile is circular and has an axis of symmetry. The top boundary is defined as the Free Boundary. The load was applied as a point load on the symmetry axis in the negative z-direction.

The bottom boundary has a constrained displacement: $u_z = 0$, to resemble the bedrock that would impede the vertical displacement of the pile and the soil. In this case, the sensitivity analysis indicated that the domain size needed to ensure the radiation of the waves was slightly bigger than in the previous case. The reason for this is that the soil domain for the disc presented an equivalence to a spherical space, given that the radiation condition was present both in the side and bottom boundary conditions. In other words, the waves generated in a localised area were attenuated in a 3D space. On the contrary, in the pile model, the soil had the same dimension as the pile in the z-direction resulting in only cylindrical and not spherical attenuation. All in all, the attenuation in a spherical domain is greater than that of a cylindrical domain, necessitating a larger domain for the pile than for the disc. The required radial length of the domain was from $r = 16\lambda_{max}$ to $r = 20\lambda_{max}$ for vertical motion, depending on the frequency of analysis.

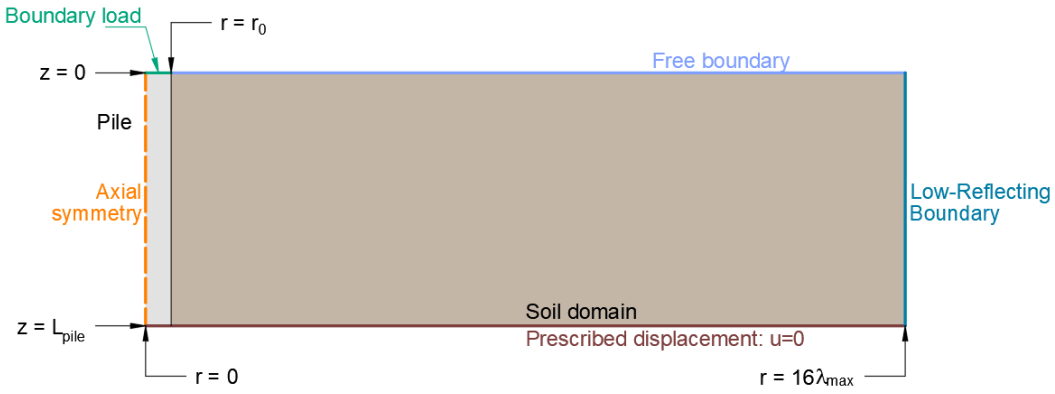


Figure 4.33: Boundary and loading conditions in FE model for *Pile in soil over bedrock.*

Figure 4.33 schematically shows the conditions of the model. This configuration was used to run the case for frequencies from 1 Hz to 50 Hz, like in the case of the disc. The range width is enough to capture any possible excitation generated by an earthquake. Lastly, the material of the piles was defined as concrete, which is representative of the Civil Engineering practice.

4.3.2. Comparison of analogue models and FE model

It is essential to determine the applicability of these analogues and understand their limitations. Consequently, the results given by the analogues will be compared to Finite Element models in the graphs below. The parameters and characteristics of the models for this comparison are stated in Table 4.6. They consider a type of soil with a shear wave velocity of 120 m/s. The material for the pile was defined as C35 concrete as prescribed in the Eurocode [48].

Characteristics of the pile	
Radius	$r_0 = 0.5$ [m]
Length	$L_p = 40$ [m]
Young's modulus	$E_{pile} = 3.2 \cdot 10^{10}$ [Pa]
Density	$\rho_{pile} = 2400$ [kg/m^3]

Characteristics of the soil	
G-modulus	$G = 2.59 \cdot 10^7$ [Pa]
Density	$\rho = 1800$ [kg/m^3]
Poisson's ratio	$\nu = 0.33$ [-]
Damping	$\xi = 0.001$ [-]

Table 4.6: Parameters of the Pile models used for analogues comparison.

Vertical motion

The first plot (Fig. 4.34) shows the response of the pile according to Novak's solution. It can be seen that the displacement at the bottom is zero, as was expected from the problem statement. Additionally, the greatest displacement is given by the static excitation ($\omega = 0$) and decreases with increasing frequency. Comparatively, the results from the FE model (see Fig. 4.35) also show zero displacements at the bottom, as expected. In this case, the static response was not computed. However, as a first analysis, the shapes and orders of magnitude of the results are similar to Novak's results.

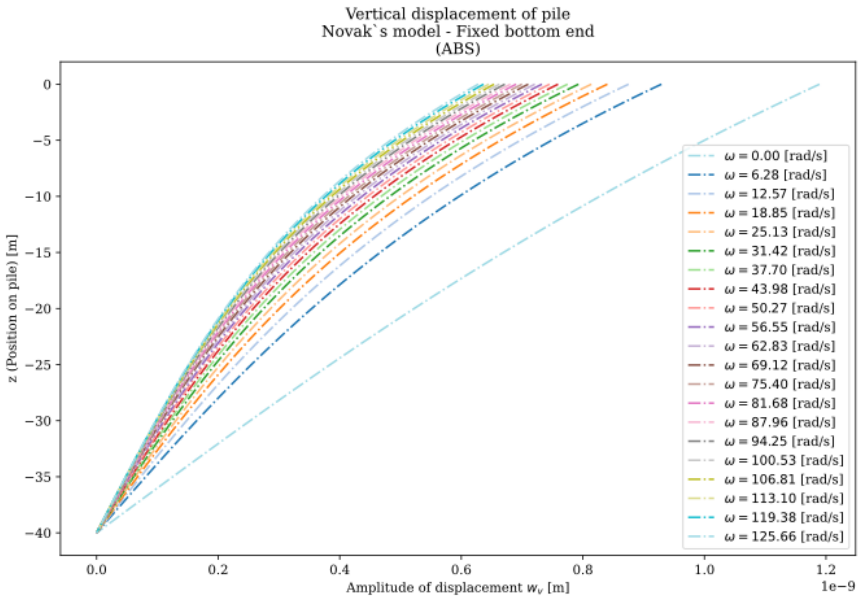


Figure 4.34: Novak's solution for a pile in soil over bedrock.

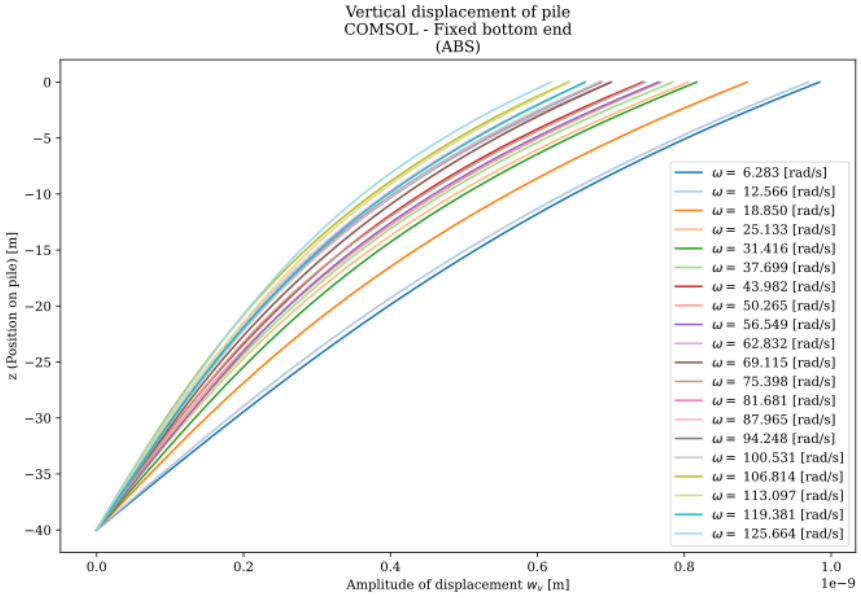


Figure 4.35: Response of a pile with fixed bottom end in FE.

The results for the pile over its length for two arbitrary individual frequencies are depicted in Figures 4.36 and 4.37. The former exhibits the comparison for a low frequency with the response at $f = 1$ Hz or $\omega = 6.28$ [rad/s]. The latter shows the response for a higher frequency of $f = 20$ Hz or $\omega = 125.66$ [rad/s]. These plots demonstrate that Novak's analogue is almost a perfect fit for this configuration. The relative error is $\bar{\varepsilon}_1 = 0.041$ and $\bar{\varepsilon}_{20} = 0.028$ for $f = 1$ and $f = 20$ respectively. Although other frequencies are not explicitly reported, the errors are in the same order of magnitude.

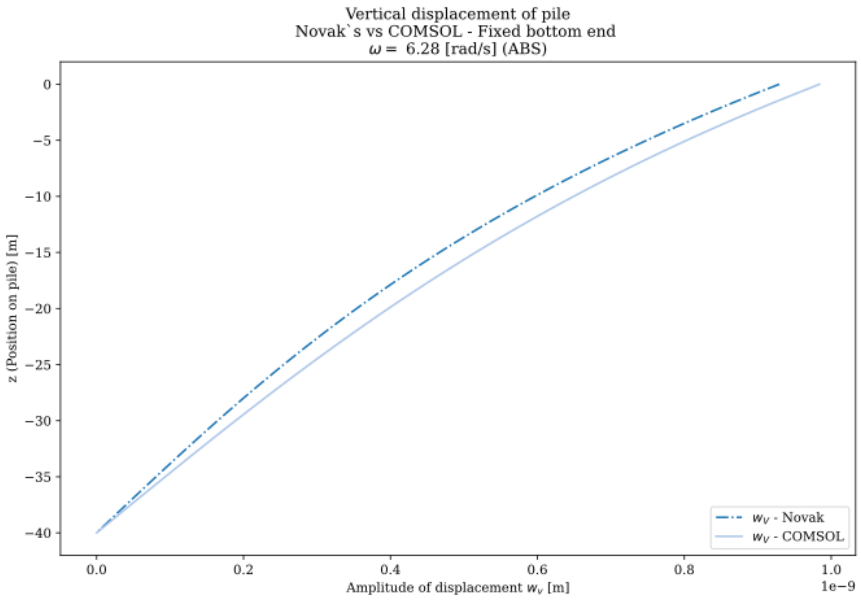


Figure 4.36: Response of a pile with fixed bottom end for $\omega = 6.28$ rad/s, $f = 1$ Hz.

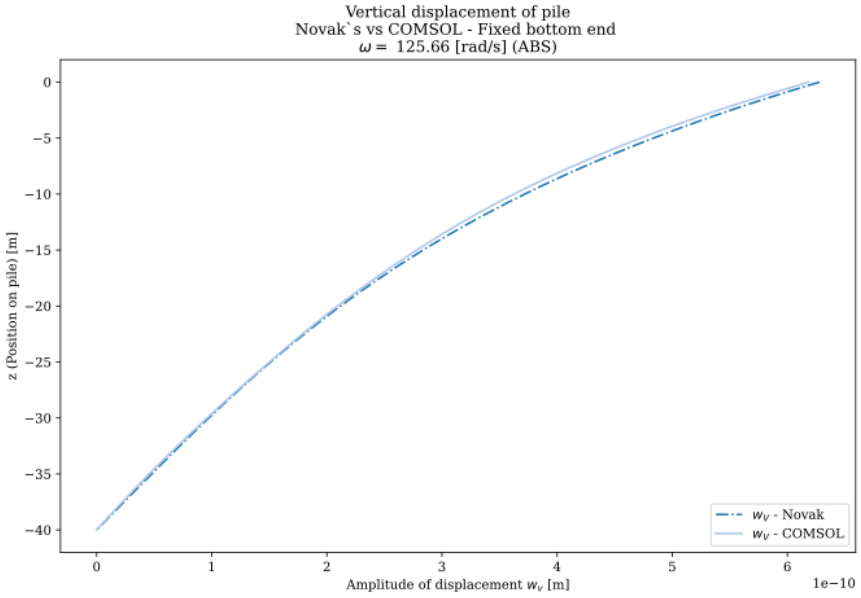


Figure 4.37: Response of a pile with fixed bottom end for $\omega = 125.66$ rad/s, $f = 20$ Hz.

The graphs in Figures 4.35, 4.36 and 4.37, show the response of the pile along its length for one or several frequencies and compare the response from COMSOL with the response given by Novak's model. Contrastingly, the following graph (see Fig. 4.38) shows the response of the particular system for different frequencies only at the cap. It can already be observed that the responses of both models are greatly comparable. It is worth mentioning that the response of the FE model possesses slight peaks. However, it cannot be confirmed if they are physical or if it is a minor numerical inaccuracy that creates this trend instead of a smoother one. An extensive sensitivity study for each FE model would be needed, and perhaps more advanced solution methods should be compared against FEM. That is out of the scope of this thesis.

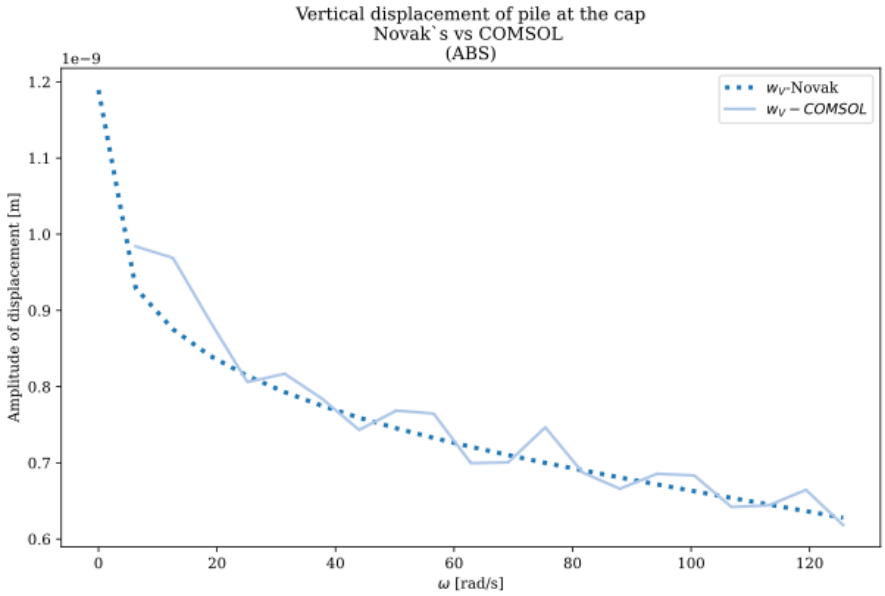


Figure 4.38: Comparison between Novak's response and COMSOL for vertical vibrations at the cap.

The comparison between the analytical and FE models retroactively indicates two things. Firstly, the model proposed by Novak for this particular SSI case captures the behaviour of the pile accurately. Secondly, the FE model was built correctly regarding the load and boundary definitions, the energy is being dissipated and the sizes of the domain and mesh are adequate. Consequently, the run for the whole parameter set could be run in order to generate the dataset needed for the NN.

4.3.3. Generation of data

The same procedure was carried out for the "pile in soil" type of foundation. In this case, there is an extra parameter, which is the length of the pile. The radius range was adjusted to adequate values for piles. The combinations of radius and length were chosen by setting L/D ratios, as conventionally done in practice. The length and radius ranges are shown in Table 4.8, as well as the corresponding L/D ratio. The values for the characteristics of the soil were the same as for the previous problem.

Parameters		Amount
$\nu_{soil} =$	$[0.2, 0.3, 0.4, 0.495]$ [-]	4
$\rho_{soil} =$	$[1400, 1600, 1800, 2000]$ [kg/m^3]	4
$G_{soil} =$	[2.0e7, 6.0e7, 1.0e8, 1.4e8, 1.8e8, 2.2e8, 2.6e8] [Pa]	7
	Combinations of soil	112

Table 4.7: Ranges of values for soil parameters for each type of motion for the dataset generation of Pile in soil.

r_{pile} [m]	L_{pile} [m]	L/D [-]
0.25	4	8
0.25	10	20
0.5	10	10
0.25	15	30
0.75	15	10
0.5	25	25
1	25	12.5
0.5	35	35
0.75	35	23.3
1	35	17.5
0.5	50	50
0.75	50	33.3
1	50	25
2	50	12.5
Combinations of piles		14

Table 4.8: Combinations of r_{pile} and L_{pile} governed by L/D.

Total number of combinations	1568
Total number of analyses per motion (50 frequencies)	78400

Table 4.9: Total number of combinations and analyses.

These combinations of parameters were run in the FE model (using the commercial software COMSOL) in order to obtain the system's response at the defined 50 frequencies. This allows to understand the tendencies of the behaviour of such a system and get an initial insight into the case. Figures 4.39 and 4.40 show the responses of a pile with $L/D = 8$ for different soil parameters. Figure 4.39 depicts that the vertical displacement at the pile's cap is greater for lower G, corresponding to softer soil and marked in blue. Congruently, a larger G of a stiffer soil (in pink) yields a smaller displacement. Additionally, an increasing Poisson's ratio results in a slightly greater displacement, as indicated in the graph by the thicker lines within the same colour. However, the effect of ν is not as relevant as the effect of G.

With the purpose of clarity, Figure 4.40 shows the response for different soil shear moduli (G) without varying the Poisson's ratio (fixed to $\nu = 0.2$) or density (fixed to $\rho = 1400$) of the soil. Significant variations in the displacement are given by the various values of G, remarking once again the great influence of this parameter in the response.

It is important to point out the peaks present in the graphs and clearly depicted in Figure 4.40. These unexpected high-energy frequencies might be due to the shortness of the pile or

some inaccuracies in the FE model. A more detailed analysis would be needed to confirm the reason for these peaks.

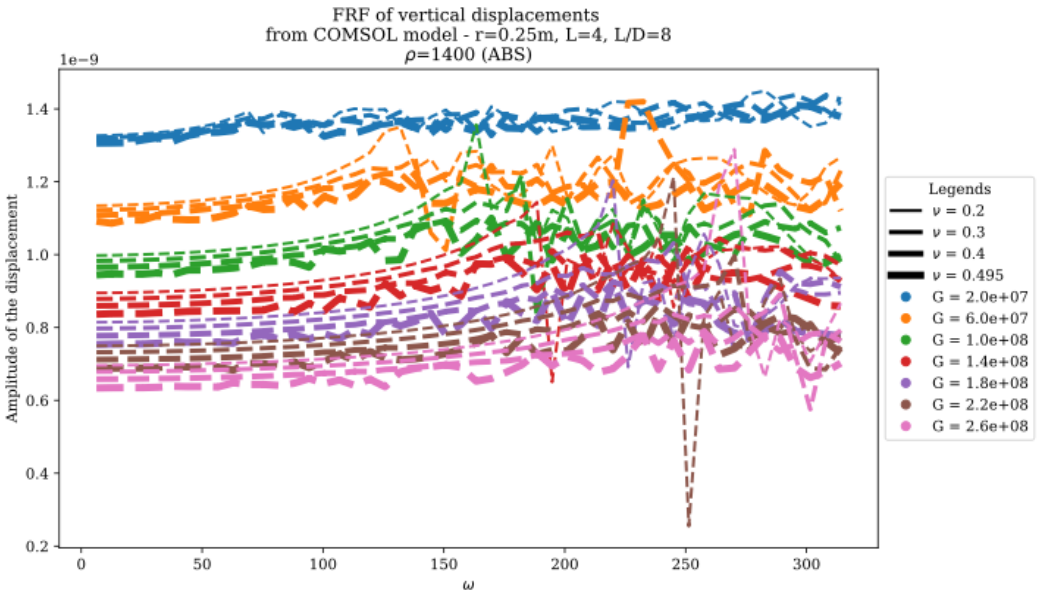


Figure 4.39: Comparison of the effect of ν and G.

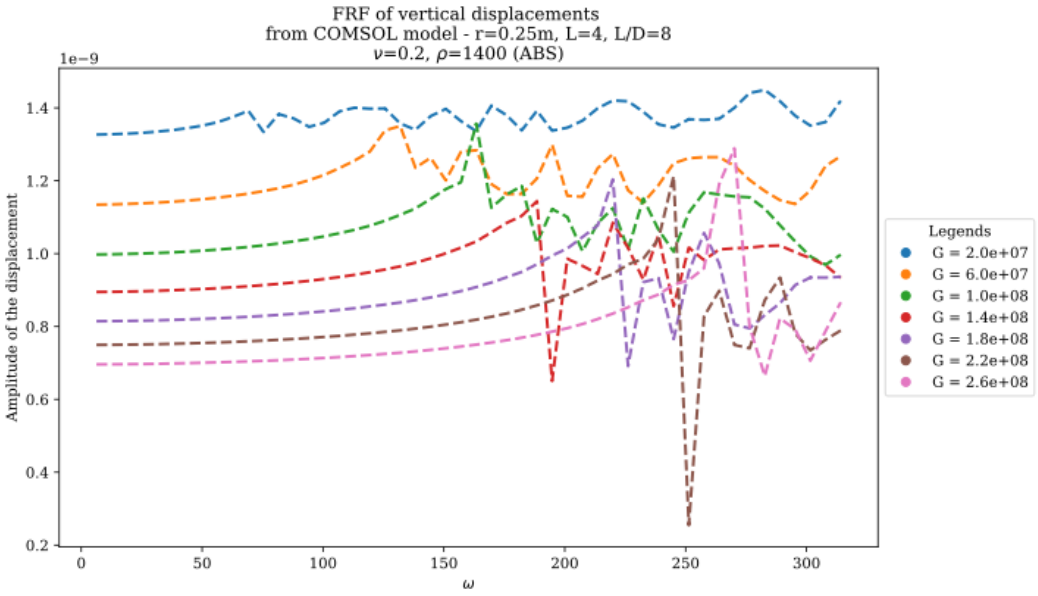


Figure 4.40: Comparison of the effect of G.

In order to compare the response of the pile, after plotting the lowest L/D ratio in the range of analysis, the highest L/D is also plotted. Figures in 4.41 and 4.42 show the response of a pile with $L/D = 50$. Once more, it can be observed from both graphs that the greatest impact in the response is given by the shear modulus of the soil, rather than the density or Poisson's ratio. Furthermore, the variation in ν or ρ does not modify the system's response substantially.

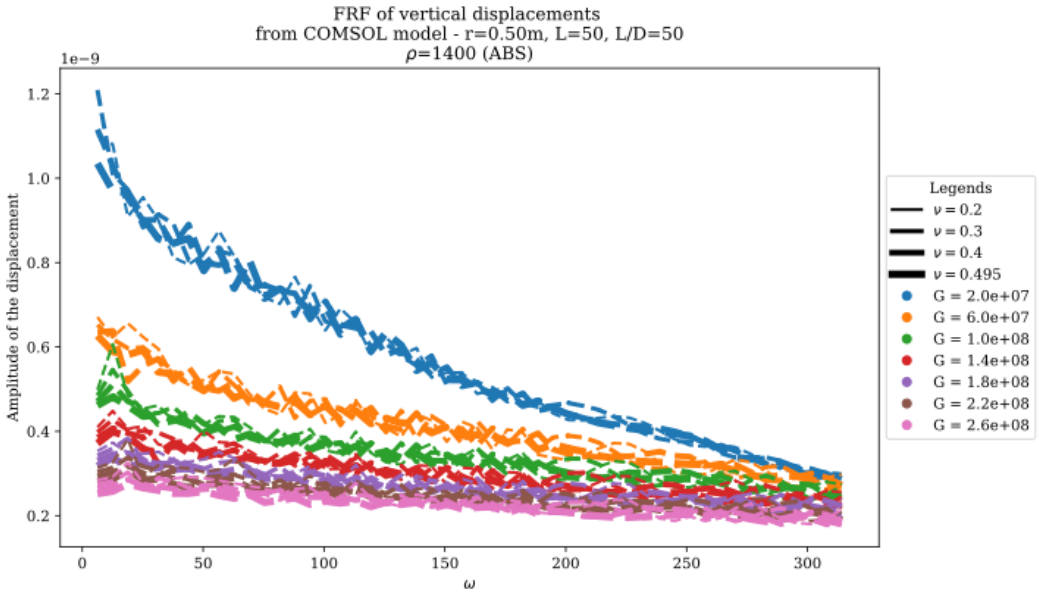


Figure 4.41: Comparison of the effect of ν and G.

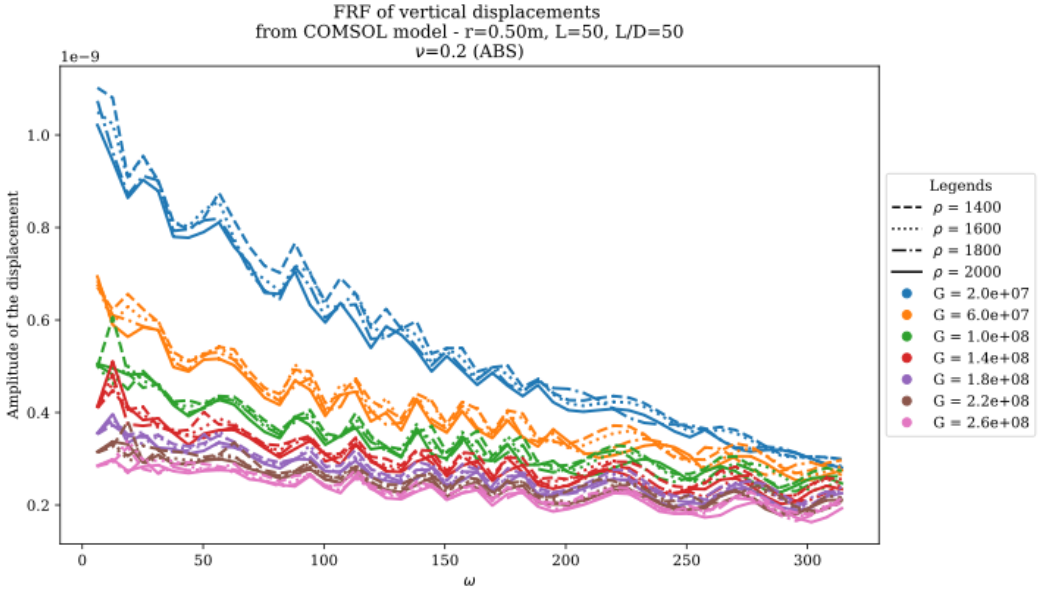


Figure 4.42: Comparison of the effect of L and G.

It should be noted that, compared to the response of the disc over a homogeneous halfspace (see Fig. 4.15), the effect of the Poisson's ratio is not as significant in the pile embedded in soil. This SSI system does not present large variations with the modification of ν (see Fig. 4.39 and 4.41). This indicates that the relevance of the parameters might change for each SSI problem and should not be overlooked.

4.3.4. Sensitivity analysis for the Neural Network and optimisation approach

The same steps were taken for this SSI scenario. Firstly, the optimal NN architecture was defined by comparing the validation losses of several options. The activation function was not varied, as the previous case indicated that ReLU performs adequately. Only the effect of the

number of neurons and of layers was looked into. Table 4.10 shows the hyperparameters and choices for the initial analysis.

Model hyperparameters	
Number of hidden layers	$[3, 4]$
Number of neurons	$[16, 32, 64, 128]$

Training choices	
Solver	*adam*
Activation function	ReLU
Weight-decay (λ)	[1e-9, 1e-8, 1e-7, 1e-6, 1e-5, 1e-4, 1e-3, 1e-2, 1e-1, 1, 10]

Table 4.10: Model hyperparameters and training choiced for finding the optimal NN for Pile in soil.

Figure 4.43 depicts the validation losses for the evaluated model hyperparameters. It can be seen that a higher number of neurons yields a faster decrease in the validation loss. Similarly to the previous case, more layers do not significantly impact the behaviour. For such reason, a 3-layer NN was chosen. The number of neurons will be chosen based on the lowest validation loss shown in Table 4.11.

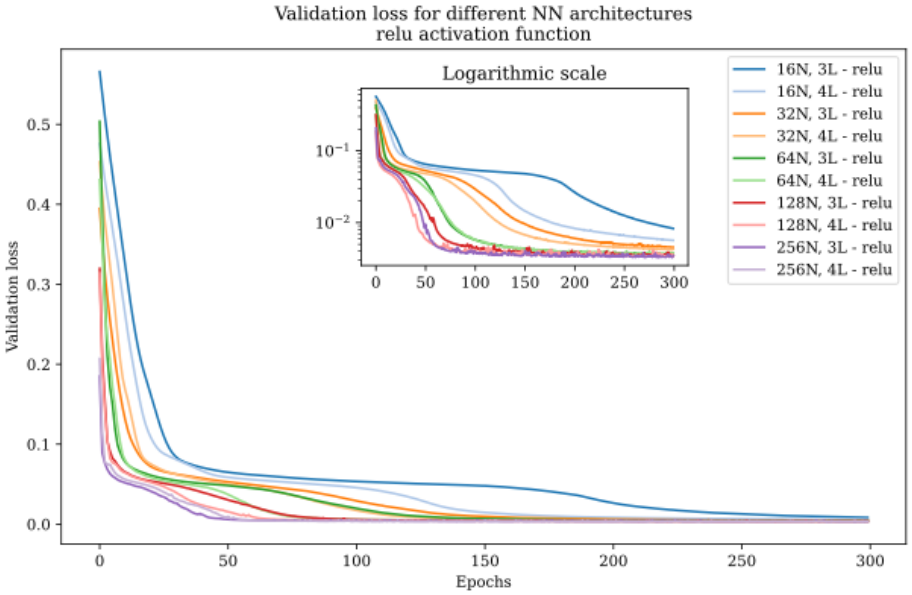


Figure 4.43: Validation loss of NN with ReLU activation functions and a varying number of layers and neurons per layer.

Number of neurons	Minimum validation loss
16	8.2787e-3
32	4.5428e-3
64	4.2290e-3
128	3.6357e-3
256	3.3915e-3

Table 4.11: Minimum validation losses for different numbers of neurons in a 3-layer NN for the pile model.

From the lowest validation losses given by each option with 300 epochs, it can be observed that the lowest validation loss is obtained with 256 neurons. However, this is not significantly lower than the validation loss of 128 neurons. For such reason, and to be consistent with the previous case, 128 neurons were chosen and only this architecture will be considered for further steps. Furthermore, a smaller NN, with fewer neurons and layers, has fewer parameters to be tuned during training, and this is convenient, as explained in Chapter 3. Figure 4.44 shows both the training and validation losses for the selected architecture.

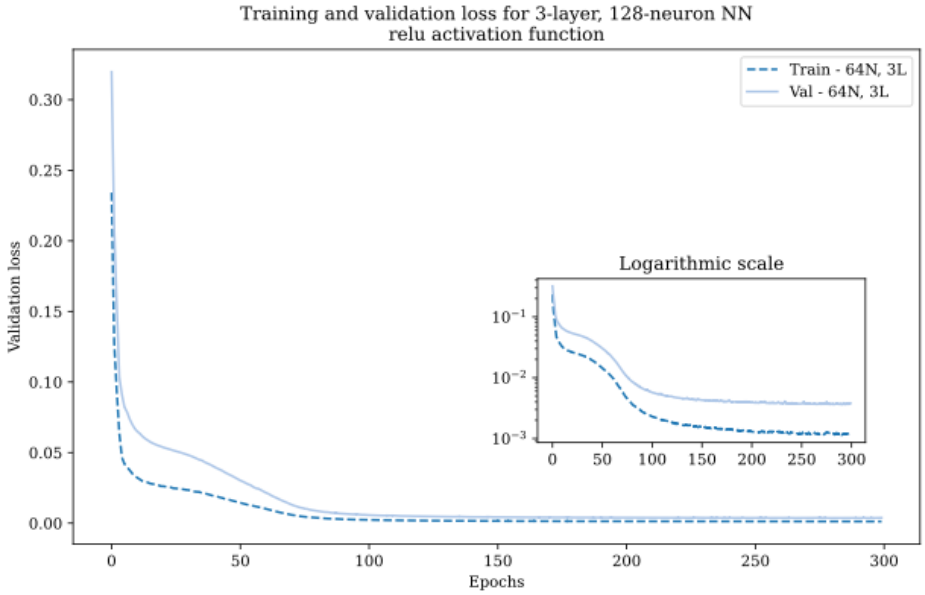


Figure 4.44: Training and validation loss of 3-layer, 128-neurons NN with ReLU activation functions for Pile in soil.

After this, the λ-values were applied to the optimiser of the model in order to avoid overfitting. The results of the validation losses for each λ are shown in Figure 4.45. The last value for each of these options is plotted in Figure 4.46. It can be observed that the minimum validation loss after 1200 epochs is given by an NN with $\lambda = 1e - 6$. Hence, the optimal NN for this problem will consider such a weight-decay.

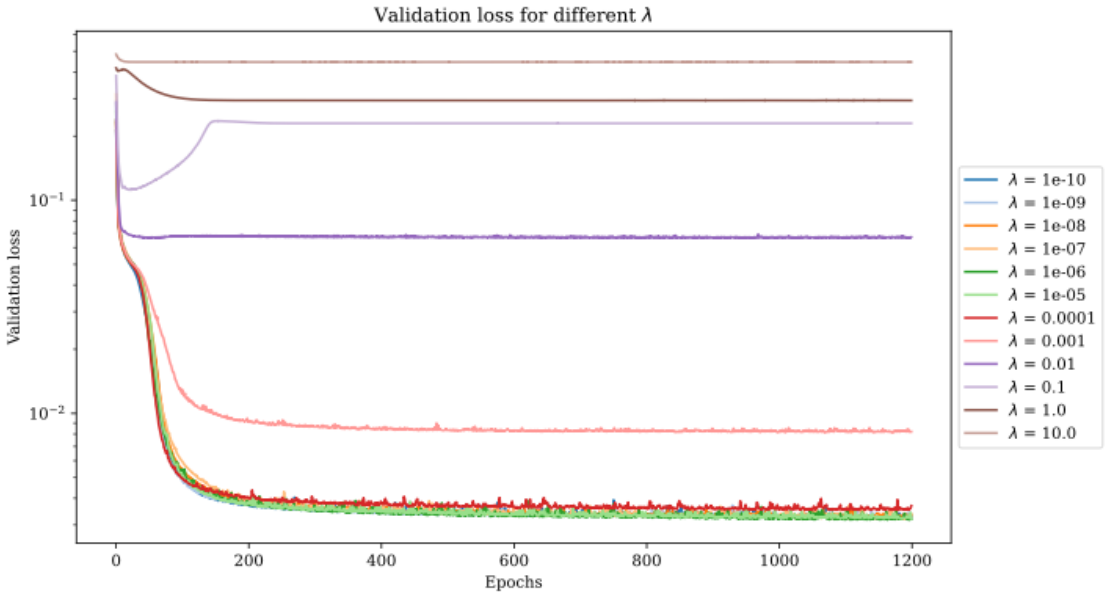


Figure 4.45: Validation losses for different λ-values for the pile model.

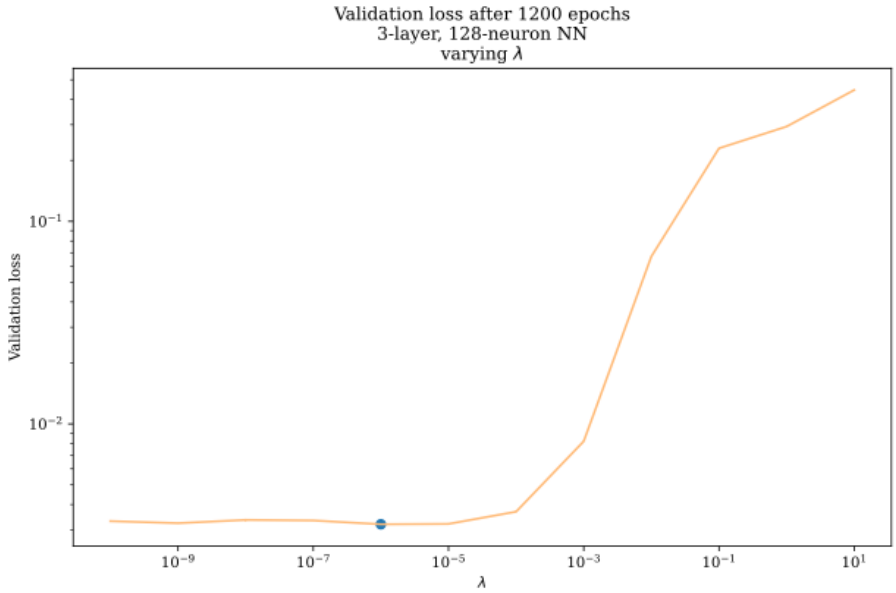


Figure 4.46: Validation loss for 3-layer, 128-neuron NN with ReLU activation function for varying λ-value after 1200 epochs.

Optimal NN

According to the results obtained from the previous hyperparameter study, the optimal Neural Network was determined. The hyperparameters that yielded an optimal performance are shown in Table 4.12.

Number of hidden layers	3
Number of neurons	128
Activation function	ReLU
Weight-decay (λ)	1e-6

Table 4.12: Optimal hyperparameters for Pile in soil.

Figure 4.47 below shows the response predicted by the trained NN for the testing set. These predictions are compared to the targets (the information generated through the FE model). As can be seen, the results align almost perfectly, indicating that the NN works adequately. Only part of the results are plotted in the additional Figure 4.48 to be able to observe the details better. It can be seen that also for this analysis, the results between the target and the predicted displacement amplitudes align well.

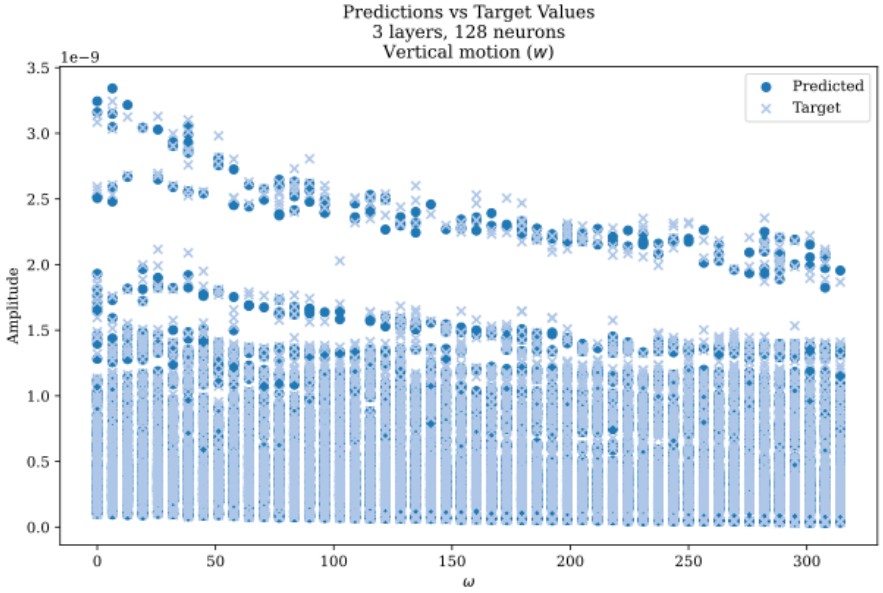


Figure 4.47: Target and predicted displacements from a trained NN - Vertical motion

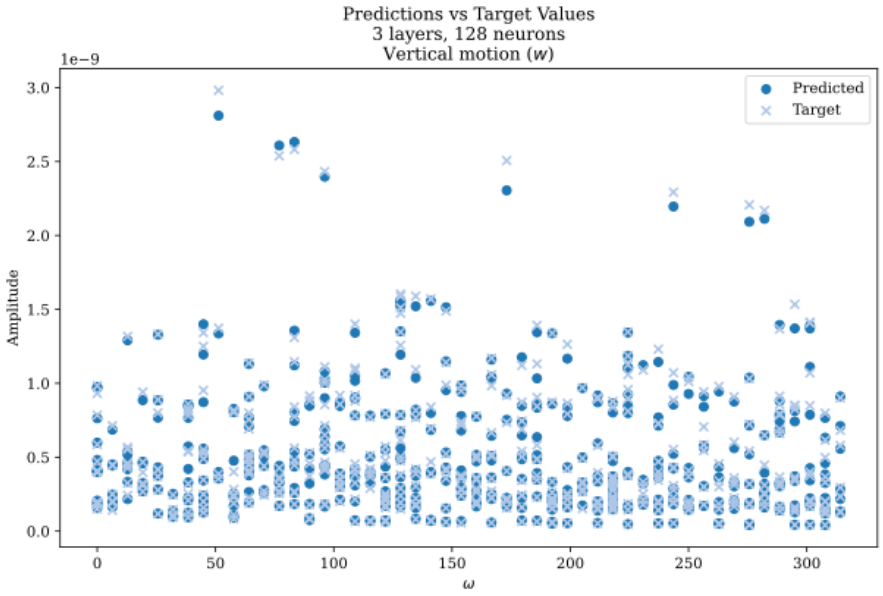


Figure 4.48: Target and predicted displacements from a trained NN (detail) - Vertical motion

Optimisation

Once again, the system's dynamic stiffness matrix was obtained using an optimisation approach. A complete array with the characteristics of the soil, the pile and the frequencies similar to the one shown in Equation 4.13 was introduced in the NN to predict the FRF. Then this one was approximated with the optimisation algorithm.

$$\begin{bmatrix} r_0 & L & \nu & \rho & G & \omega \\ 0.7 & 43 & 0.22 & 1954 & 8e7 & 0 \\ 0.7 & 43 & 0.22 & 1954 & 8e7 & 3.1416 \\ 0.7 & 43 & 0.22 & 1954 & 8e7 & 6.2832 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0.7 & 43 & 0.22 & 1954 & 8e7 & \omega_n \end{bmatrix} \tag{4.13}$$

1DoF optimisation Firstly, 1DoF optimisation was evaluated. The FRF predicted by the NN and the optimisation process are shown in Figure 4.49. It can be observed that NN predictions follow the targets but not so closely. Some of the information on the peaks given by the targets is missed in the prediction. The plot also depicts that the optimisation can approximate the predicted response and provide the system's mass, stiffness and damping matrices fairly accurately. However, 2DoF optimisation is also studied to evaluate if it gives a better response.

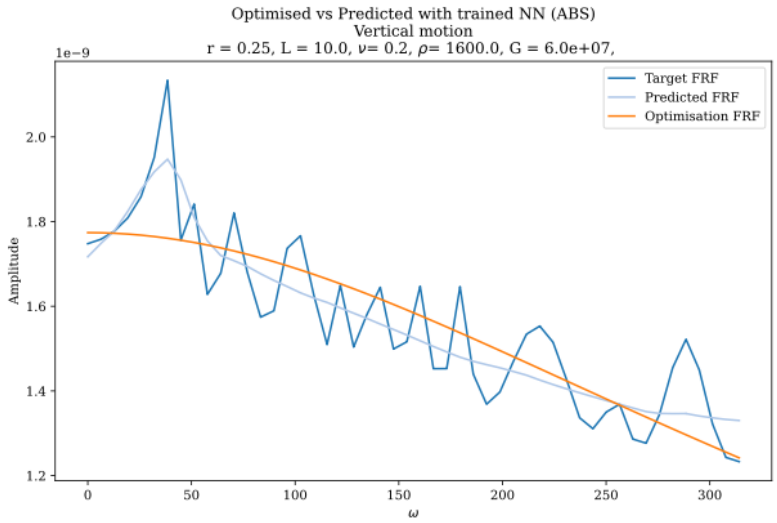


Figure 4.49: Comparison of absolute FRF of vertical motion with 1DoF optimisation.

2DoF optimisation A 2DoF optimisation was evaluated. Figure 4.50 shows the target, predicted and optimised FRF for the same system as before. It can be observed that the optimisation with 2 DoFs also gives an accurate response. The advantage is that the SSI would be included in the overall analysis with a higher DoF system and therefore more information.

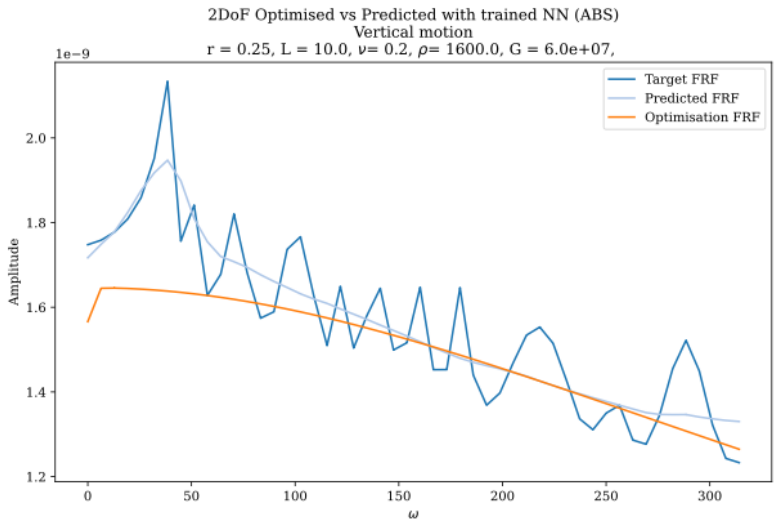


Figure 4.50: Comparison of absolute FRF of vertical motion with 1DoF optimisation.

4.4. Conclusions

A hybrid computational method was required to solve soil-structure interaction problems. Neural Networks provided the possibility of generating FRFs for new soil and foundation characteristics without having to run a complete FE model. The correct tuning of this model ensured the accuracy of the predicted FRF. Concurrently, the optimisation process allowed to obtain the M, C and K values needed for the final SSI consideration in structural analysis with 1 and 2 DoFs. However, finer tuning of the NN and optimisation process could provide better predictions and optimised FRF. Further refining of the parameters especially in the 2DoF optimisation is still required.

5

Conclusions and recommendations

5.1. Conclusions

The significance of Soil-Structure Interaction (SSI) has been well-established by numerous researchers, and it remains a topic of ongoing study. Recognising its critical importance, various models and approaches have been developed to address SSI. However, existing mechanical analogues are often tailored to specific scenarios, and a comprehensive, generic approach is still lacking. Moreover, there is a notable absence of straightforward analogues for more complex foundation arrangements, highlighting an area that requires further investigation and development.

This raises the research questions addressed in this thesis. The first aim is to understand the essential parameters and tools to generate the dataset for different SSI cases. This includes determining the soil and foundation properties relevant to the analysis. The second objective is to define an efficient data-driven process for solving SSI in the frequency domain. This involves generating a reduced model and its formulation.

To accomplish the objectives and answer the research questions, several steps were taken. Existing mechanical analogues for simple SSI problems like the disc over homogeneous halfspace and pile embedded in a homogeneous soil stratum were studied in this thesis and were found to be effective. These mechanical analogues solve the SSI problems with *impedance functions*. However, more complex foundation arrangements lack accurate analogues, highlighting the need for further research in this area.

This thesis proposed to approach the solution of SSI problems using Neural Networks and optimisation algorithms. To do so, data to train the Neural Network was generated. Afterwards, different Neural Networks' architectures were evaluated. Once properly tuned, they provided accurate predictions of the frequency response function of a foundation-soil system based on its characteristics parameters. Subsequently, an optimisation process was used to derive the necessary system's reduced mass (M), damping (C) and stiffness (K) matrices for these models. In conclusion, the research questions could be addressed.

First research question

What are the essential SSI parameters and necessary tools required to establish a consistent modelling framework for generating data suitable for a computational approach for SSI cases?.

The generation of data requires efficient tools that can run analyses for different soil and foundation characteristics. This can be done using FEM software that allows for a systematic loop over a set of parameters, although it is not limited to commercial FEM programs. If an

alternative model to FE models was available for a specific SSI problem, it could also be used to generate data. This approach may even be computationally preferable. Once the dataset was generated for the SSI problems analysed in this work, it was possible to determine the properties of the soil and foundation that have a greater impact on the response.

On one hand, the Poisson's ratio, mass density, and shear modulus of the soil are enough parameters to define the soil in this application. However, they do not affect the SSI equally. Considering the two analysed cases, the response was mostly determined by the effect of the shear modulus. Softer soils generated a significantly greater response than stiffer soils. Poisson's ratio had the second most important role. Lower ν elicited a smaller displacement than a higher ν. Nevertheless, it must be noted that the effect of this soil characteristic was more prominent in the case of the disc over homogeneous halfspace than the pile embedded in soil. This is attributed to the consequence of the embedment. The reaction of the soil on the pile is located on the sides of the element, while for the disc it is located underneath and over the whole area. Therefore the difference in compressibility of the soil is much more relevant in the disc than in the pile. Lastly, the density of the soil had very little effect on the response of the systems. The differences in the response of systems with the lowest density and the highest in the frequency range were negligible. In conclusion, the most important parameters in the behaviour of the soil for SSI problems are the shear modulus G and the Poisson's ratio ν, It would be interesting to evaluate the effectiveness of the proposed approach considering only these two and disregarding the density of the soil.

On the other hand, only the most determining geometric characteristics of the foundation were taken into account. For the disc, only the effect of the radius was considered. This is justified because the thickness in discs, like in plates, is significantly small compared to the wavelengths excited in the disc and can be disregarded. In this case, it is evident that the responses vary significantly with different radii. Therefore it definitely cannot be disregarded from the analysis. On the contrary, it would be interesting to investigate the effect of other aspects, like the thickness, although it is known to not play an important role in a rigid disc. In the case of the pile, only radius and length were considered and both were proven to affect the response. Therefore, it would not be possible to remove either of them.

Both models have the ν, ρ and G of the soil as common parameters. These should be considered in every SSI problem for the proposed approach. The removal of the density could be evaluated to reduce the number of analyses in the data generation process and the number of input neurons in the NN, which would simplify the process. This is because the range of this parameter, both in the study and in reality, is smaller than the range of the shear modulus. Therefore the impact of its variation is not as determining. The modification in the geometric parameters is more limited and should probably kept unaltered.

Second research question

What constitutes an efficient data-driven process for translating large soil-foundation datasets into simplified mechanical analogues that provide a reduced and accurate representation of the system in the frequency domain?

The main objective of this thesis was to present a Neural Network-based approach that can provide the system's mass, damping and stiffness matrices that can effectively reproduce the dynamic behaviour of the soil-foundation system. For such purpose, a two-step hybrid method was developed combining Neural Networks and an optimisation process.

Several other alternatives were also evaluated. For example, to retrieve the system's reduced MCK matrices directly from the Neural Network instead of having a two-step approach. A second alternative was having 3 different NNs, one for each motion in the disc case. Lastly, not having the frequencies as an input parameter in the NN. However, the results were not

satisfactory. On the contrary, the two-step proposed method was able to provide the desired result.

The NN-based two-step approach is applicable to scenarios with different characteristics, like disc and pile. This indicates that is possible to be applied to other more complicated foundations. To do so, understanding the most important parameter in such a foundation system would be crucial. For example, analysing pile-soil-pile foundations may demand considering parameters apart from the geometry of the pile. Additionally, it offers a practical and efficient alternative to traditional FEM models for SSI problems, with potential applications in engineering practice.

To conclude, it was possible to develop a generic approach to obtain mechanical analogues for different SSI problems. The selected hybrid approach proved to be the best option for the current work. The NN successfully predicted the FRF for new input parameters. The optimisation process showed that an SDoF provided appropriate results for both cases and that more degrees of freedom have the potential to provide interesting results.

5.2. Recommendations for future research

The computational tools that were explored and studied showed their particular implications. Further development and refinement of both the Neural Networks and the optimisation process could lead to broader adoption in the field, improving the efficiency and accuracy of SSI analysis. For the former, additional hyperparameters could be considered for the tuning process. Additionally, a larger range of the analysed hyperparameters could also give a greater insight into the behaviour of the NN. Finally, other architectures could be studied to perhaps improve the results. For the optimisation, other tools and algorithms could be tested, as they could show better results. Additionally, the process for a higher number of degree of freedom should be finely revised to assure its applicability to every case.

By advancing the computational methods used for SSI problems, this thesis contributes to the field of civil engineering, offering innovative solutions that bridge the gap between traditional modelling techniques and modern AI-driven approaches.

References

[1] Vishwajit Anand and S.R. Satish Kumar. "Seismic Soil-structure Interaction: A State-of-the-Art Review". In: *Engineering Advance - Strucutres* 16 (2018), pp. 317–326.

[2] Y. O. Beredugo and M. Novak. "Coupled Horizontal and Rocking Vibrations of Embedded Footings". In: *Canadian Geotechnical Journal* 9 (1972), pp. 477–497.

[3] By The DSAIE teaching team. *Data Science and Artificial Intelligence for Engineering*. 1st ed. 2024. URL: https://interactivetextbooks.citg.tudelft.nl/dsaie/intro.html.

[4] Shi-Shuenn Chen and Jun-Yang Shi. "Simplified Model for Vertical Vibrations of Surface Foundations". In: *Journal of Geotechnical and Geoenvironmental Engineering* 132(5) (2006), pp. 651–655.

[5] Kenny F. Conto et al. "A pile–soil interaction model for ground-borne vibration problems based on the singular boundary method". In: *Journal of Sound and Vibration* 568 (2024).

[6] P.S. Dineva, G.D. Manolis, and F. Wuttke. "Fundamental solutions in 3D elastodynamics for the BEM: A review". In: *Engineering Analysis with Boundary Elements* 105 (2019), pp. 47–69. DOI: https://doi.org/10.1016/j.enganabound.2019.04.003. URL: https://www.sciencedirect.com/science/article/pii/S0955799718305290.

[7] MIT E. Kausel. *Advanced Structural Dynamics*. 1st ed. One Liberty Plaza, 20th Floor, New York, NY 10006, USA: Cambridge University Press, 2017.

[8] François Fleuret. *The Little Book of Deep Learning*. Université de Genève, 2023.

[9] George Gazetas and George Mylonakis. "Seismic soil-structure interaction: new evidence and emerging issues". In: *Geotechnical Earthquake Engineering and Soil Dynamics III* 2 (1998), pp. 1119–1174.

[10] Geeks for Geeks. *Why ReLU is better than other Activation Functions?* 2024. URL: https://www.geeksforgeeks.org/why-relu-is-better-than-the-other-activation-functions/ (visited on 07/05/2024).

[11] Lu Lu George Em Karniadakis Khemraj Shukla. *Deep Learning for Science and Engineering Teaching Kit - Lecture 1.1- Introduction to Deep Learning*. NVIDIA and Brown University.

[12] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. http://www.deeplearningbook.org. MIT Press, 2016.

[13] Bipin K. Gupta. "A hybrid formulation for torsional dynamic response of pile foundations in a layered soil deposits". In: *Computers and Geotechnics* 168 (2024), p. 106115. ISSN: 0266-352X. DOI: https://doi.org/10.1016/j.compgeo.2024.106115. URL: https://www.sciencedirect.com/science/article/pii/S0266352X2400051X.

[14] George W. Housner. "Interaction of building and ground during an earthquake". In: *Bulletin of the Seismological Society of America* 47.3 (July 1957), pp. 179–186. ISSN: 0037-1106. DOI: 10.1785/BSSA0470030179. eprint: https://pubs.geoscienceworld.org/ssa/bssa/article-pdf/47/3/179/5303064/bssa0470030179.pdf. URL: https://doi.org/10.1785/BSSA0470030179.

[15] Lars Bo Ibsen and Morten Liingaard. “Lumped-parameter models”. In: *Deparment of Civil Engineering, Aalborg University* 11 (2006).

[16] A.K. Jain, Jianchang Mao, and K.M. Mohiuddin. “Artificial neural networks: a tutorial”. In: *Computer* 29.3 (1996), pp. 31–44. DOI: 10.1109/2.485891.

[17] Takaaki Kagawa. “Seismic Soil-Pile-Structure Interaction, Pile Groups”. In: *National Science Foundation* (1981).

[18] Eduardo Kausel. In: *Fundamental Solutions in Elastodynamics: A Compendium*. Cambridge University Press, 2006.

[19] Eduardo Kausel. “Early history of soil–structure interaction”. In: *Soil Dynamics and Earthquake Engineering* 30.9 (Sept. 2010), pp. 822–832.

[20] Amir M. Kaynia. “Seismic considerations in design of offshore wind turbines”. In: *Soil Dynamics and Earthquake Engineering* 124 (2019), pp. 399–407. ISSN: 0267-7261. DOI: https://doi.org/10.1016/j.soildyn.2018.04.038. URL: https://www.sciencedirect.com/science/article/pii/S0267726117309363.

[21] Diederik P. Kingma and Jimmy Lei Ba. “Adam: A Method for Stochastic Optimization”. In: *International Conference on Learning Representations 2015* (2017). DOI: https://doi.org/10.48550/arXiv.1412.6980.

[22] Horace Lamb. “On the propagation of tremors over the surface of an elastic solid”. In: *Royal Society* (1904). DOI: https://doi.org/10.1098/rsta.1904.0013.

[23] Jianbo Li et al. “A novel hybrid framework based on modified continued-fraction for pile-soil dynamic interaction of large-scale nuclear power structures”. In: *Computers and Geotechnics* 173 (2024), p. 106571. DOI: https://doi.org/10.1016/j.compgeo.2024.106571. URL: https://www.sciencedirect.com/science/article/pii/S0266352X2400510X.

[24] J. Lysmer. “Vertical Motion of Rigid Footings”. PhD thesis. University of Michigan, 1965.

[25] S.M. Mamoon, A.M. Kaynia, and P.K. Banerjee. “Frequency Domain Dynamic Analysis of Piles and Pile Groups”. In: *Journal of Engineering Mechanics* 116.10 (1990), pp. 2237–2257.

[26] D. Margalit and J. Rabinoff. *Interactive Linear Algebra*. Georgia Institute of Technology, 2017.

[27] Harsh K. Mistry and Domenico Lombardi. “Role of SSI on seismic performance of nuclear reactors: A case study for a UK nuclear site”. In: *Nuclear Engineering and Design* 364 (2020), p. 110691. ISSN: 0029-5493. DOI: https://doi.org/10.1016/j.nucengdes.2020.110691. URL: https://www.sciencedirect.com/science/article/pii/S0029549320301850.

[28] *MPh - Pythonic scripting interface for Comsol Multiphysics*. 2024. URL: https://mph.readthedocs.io/en/stable/index.html (visited on 07/06/2024).

[29] NIST and NEHRP Consultants Joint Venture. *Soil-Structure Interaction for Building Structures*. US Department of Commerce, 2012. URL: https://www.nzsee.org.nz/db/2003/View/Paper050s.pdf.

[30] T. Nogami and M. Novak. “Resistance of soil to a horizontally vibrating pile”. In: *Earthquake Engineering and Structural Dynamics* 5 (1977), pp. 249–261.

[31] T. Nogami and M. Novak. “Soil-pile Interaction in Vertical Vibration”. In: *Earthquake engineering and structural dynamics* 4 (1976), pp. 277–293.

[32] Toyoaki Nogami, M. ASCE, and Kazuo Konagai. "Time domain axial response of dynamically loaded single piles". In: *Journal of Engineering Mechanics* 11.112 (1986), pp. 1241–1252.

[33] Toyoaki Nogami, M. ASCE, and Kazuo Konagai. "Time domain flexural response of dynamically loaded single piles". In: *Journal of Engineering Mechanics* 9.114 (1988), pp. 1512–1525.

[34] M. Novak. "Effect of soil on structural response to wind and earthquake". In: *Earthquake engineering and structural dynamics* 3.1 (1974), pp. 79–96. DOI: https://doi.org/10.1002/eqe.4290030107.

[35] M. Novak, M. ASCE, and B.E. Sharnouby. "Stiffness constants of single piles". In: *Journal of Geotechnical Engineering* 7 (1983), pp. 961–974.

[36] M. Novak and K. sachs. "Torsional and coupled vibrations of embedded footings". In: *Earthquake engineering and structural dynamics* 2.1 (1972), pp. 11–33. DOI: https://doi.org/10.1002/eqe.4290020103.

[37] Milos Novak. "Dynamic Stiffness and Damping of Piles". In: *Canadian Geotech Journal* 11 (1974), pp. 574–598.

[38] A. Pais and E. Kausel. "Approximate formulas for dynamic stiffnesses of rigid foundations". In: *Soil Dynamics and Earthquake Engineering - MIT, Cambridge* 7.4 (1988), pp. 213–227.

[39] Li Pang, Chong Jiang, and Chaoyang Zhang. "An analytical method for predicting the soil-structure interaction of axially loaded piles in sands incorporating the two-surface plasticity model". In: *Engineering Structures* 295 (2023).

[40] *PyTorch*. 2024. URL: https://pytorch.org/ (visited on 07/06/2024).

[41] PyTorch Contributors. *Adam*. 2023. URL: https://pytorch.org/docs/stable/generated/torch.optim.Adam.html (visited on 07/05/2024).

[42] Ellen M. Rathje, Norman A. Abrahamson, and Jonathan D. Bray. "Simplified Frequency Content Estimates of Earthquake Ground Motions". In: *Journal of Geotechnical and Geoenvironmental Engineering* 124.2 (1998), pp. 150–159. DOI: 10.1061/(ASCE)1090-0241(1998)124:2(150). eprint: https://ascelibrary.org/doi/pdf/10.1061/%28ASCE%291090-0241%281998%29124%3A2%28150%29. URL: https://ascelibrary.org/doi/abs/10.1061/%28ASCE%291090-0241%281998%29124%3A2%28150%29.

[43] Jayashree Kalpathy-Cramer Michael F. Chiang J. Peter Campbell Rene Y. Choi Aaron S. Coyner. *Introduction to Machine Learning, Neural Networks, and Deep Learning*. 2020. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7347027/ (visited on 05/21/2024).

[44] Iuri Rocha. *Modelling, Uncertainty and Data for Engineers - 2.6.1f Forward Neural Networks*. 2022.

[45] scikit-learn developers. *Scikit Learn - MLPRegressor*. 2024. URL: https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPRegressor.html#sklearn.neural_network.MLPRegressor.score (visited on 07/04/2024).

[46] Serokell. *Artificial Intelligence vs. Machine Learning vs. Deep Learning: What's the Difference*. 2020. URL: https://ai.plainenglish.io/artificial-intelligence-vs-machine-learning-vs-deep-learning-whats-the-difference-dccce18efe7f (visited on 05/21/2024).

[47] Sagar Sharma. *Apoch vs Batch size vs Iterations*. 2017. URL: https://towardsdatascience.com/epoch-vs-iterations-vs-batch-size-4dfb9c7ce9c9 (visited on 05/21/2024).

[48] European Standard. *Eurocode 2 – Design of concrete structures - Concrete bridges - Design and detailing rules*. 2005. URL: https://eurocodeapplied.com/design/en1992/concrete-design-properties.

[49] J.R. Styles, P.J. Moore, and R.K. Gupta. "Resonance Avoidance in Seismic Design". In: *6th Australia - New Zealand Conference on Geomechanics* (1992), pp. 464–467.

[50] Izuru Takewaki. "Remarkable response amplification of building frames due to resonance with the surface ground". In: *Soil Dynamics and Earthquake Engineering* 17 (1998), pp. 211–218. DOI: https://doi.org/10.1016/S0267-7261(98)00006-2.

[51] Riccardo Taormina. "Introduction to Machine Learning". In: *MUDE 2022-2023* (2022).

[52] The SciPy community. *Optimization and root finding - scipy.optimize - least$_s$quares*. 2008. URL: https://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.least_squares.html (visited on 07/04/2024).

[53] The SciPy community. *Optimization and root finding (scipy.optimize)*. 2008. URL: https://docs.scipy.org/doc/scipy/reference/optimize.html (visited on 12/06/2023).

[54] W. a.k.a. Lord Kelvin Thomson. "On the equations of equilibrium of an elastic solid". In: *The Cambridge and Dublin mathematical journal* 3 (1848), pp. 87–89.

[55] Apostolos Tsouvalas. *Applied Dynamics, Structural Response to Earthquakes - Lecture 15, Soil-Strcutre Interaction in Earthquake Engineering (Part I)*. 2023.

[56] Varun, D. Assimaki, and G. Gazetas. "A simplified model for lateral response of large diameter caisson foundation - Linear elastic formulaiton". In: *Soil Dynamics and Earthquake Engineering* 29 (2009), pp. 268–291.

[57] J. Wang et al. "Wave-passage effect on the seismic response of long bridges". In: *2003 Pacific Conference on Earthquake Engineering* 050 (2006). URL: https://www.nzsee.org.nz/db/2003/View/Paper050s.pdf.

[58] Lizhong Wang et al. "A unified lateral soil reaction model for monopiles in soft clay considering various length-to-diameter (L/D) ratios". In: *Ocean Engineering* 212 (2020).

[59] J.P. Wolf. "Spring-dashpot-mass models for foundations". In: *EARTHQUAKE ENGINEERING AND STRUCTURAL DYNAMICS* 26 (1997), pp. 931–947.

[60] John P. Wolf. *Foundation Vibration Analysis Using Simple Physical Moldels*. Upper Saddle River, NJ 07458: PTR Prentice Hall, 1994.

[61] Wen-Hwa Wu and Wen-How Lee. "Nested lumped-parameter models for foundation vibrations". In: *EARTHQUAKE ENGINEERING AND STRUCTURAL DYNAMICS* 33 (2004), pp. 1051–1058.

[62] Wen-Hwa Wu and Wen-How Lee. "Systematic lumped-parameter models for foundations based on polynomia-fraction approximation". In: *EARTHQUAKE ENGINEERING AND STRUCTURAL DYNAMICS* 31 (2002), pp. 1383–1412.

[63] Sophia Yang. *Deep learning basics — weight decay*. 2023. URL: https://medium.com/analytics-vidhya/deep-learning-basics-weight-decay-3c68eb4344e9 (visited on 07/05/2024).

A

Additional FRF of SSI cases

Results for r = 2.70 m - Disc over homogeneous halfspace

Figures A.1, A.2 and A.3 show the response of a disc with $r_0 = 2.7$ under vertical, horizontal and rocking motions respectively. Compared to smaller radii, it can be observed that the response at lower frequencies is much higher than at higher ranges. Apart from that difference, the effect of the soil parameters is similar to that of the case of $r_0 = 0.2$ (see Fig. 4.15, 4.16 and 4.17). Once again the shear modulus is the parameter that affects the most, followed by the Poisson's ratio in second place and lastly the density.

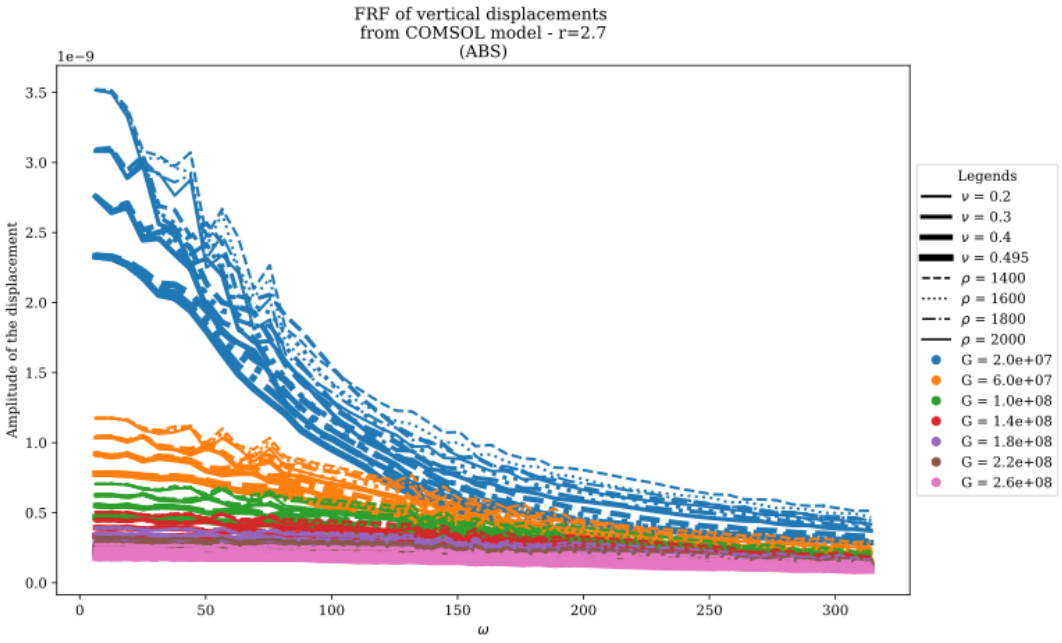


Figure A.1: Results of vertical motion for all soil combinations with $r_0 = 2.7$ m.

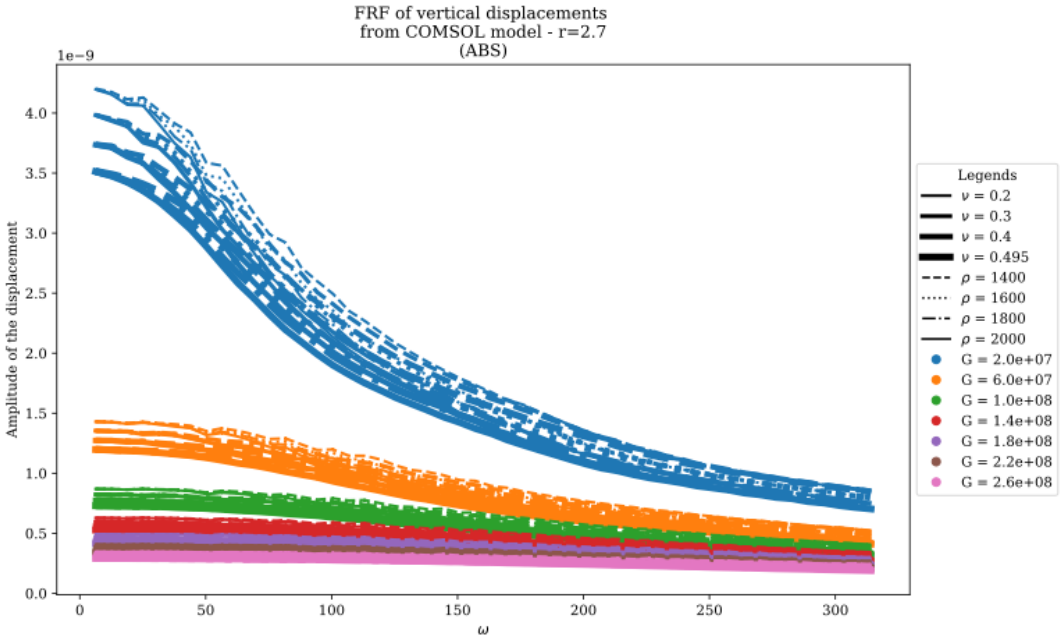


Figure A.2: Results of horizontal motion for all soil combinations with $r_0 = 2.7$ m.

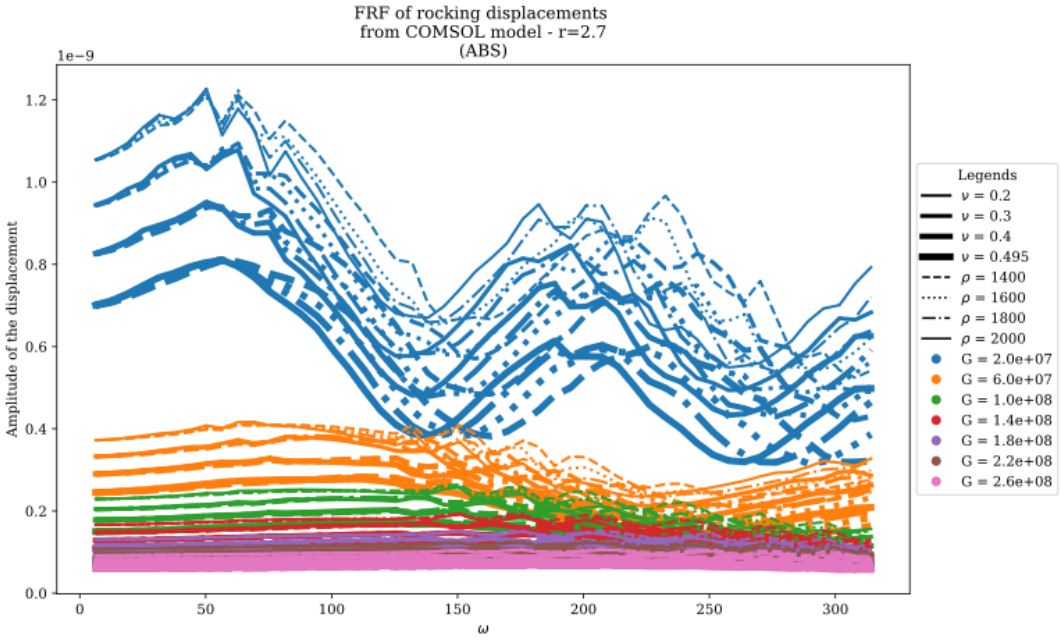


Figure A.3: Results of rocking motion for all soil combinations with $r_0 = 2.7$ m.

Results for r = 5.20 m - Disc over homogeneous halfspace

Figures A.4, A.5 and A.6 show the response of a disc with $r_0 = 5.2$ under vertical, horizontal and rocking motions respectively. Once again, it can be observed that the response at lower frequencies is much higher than at higher ranges and that the effect of the soil parameters is similar to that of the case of $r_0 = 0.2$ (see Fig. 4.15, 4.16 and 4.17).

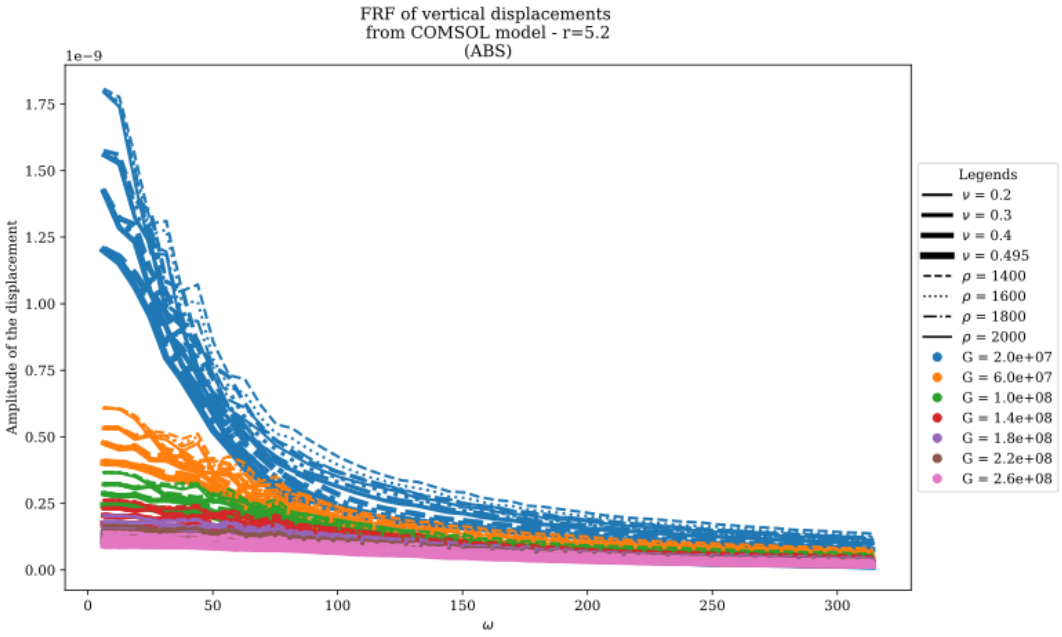


Figure A.4: Results of vertical motion for all soil combinations with $r_0 = 5.2$ m.

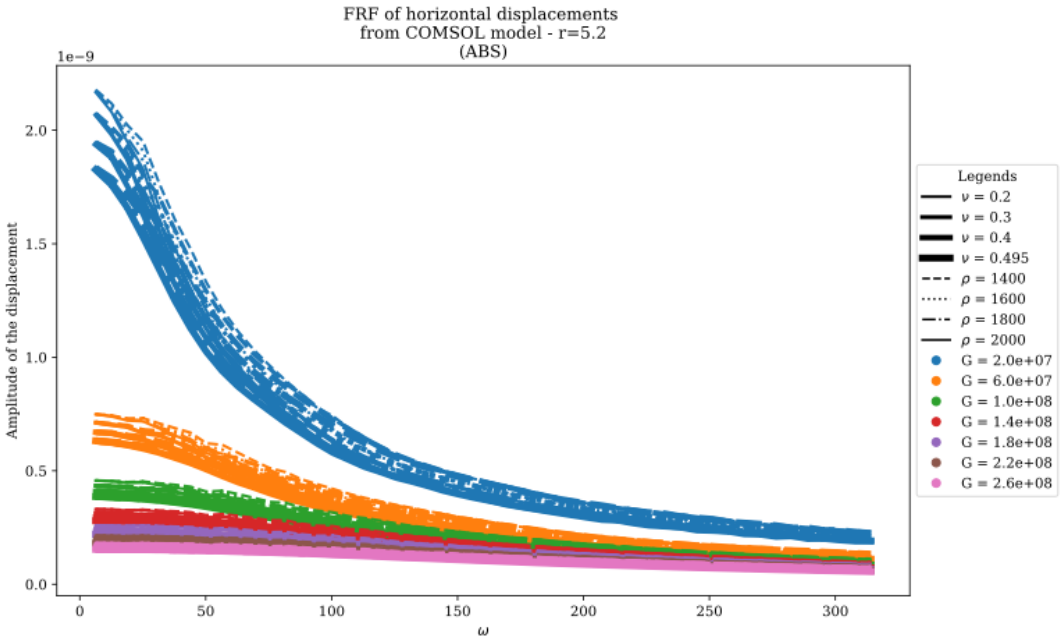


Figure A.5: Results of horizontal motion for all soil combinations with $r_0 = 5.2$ m.

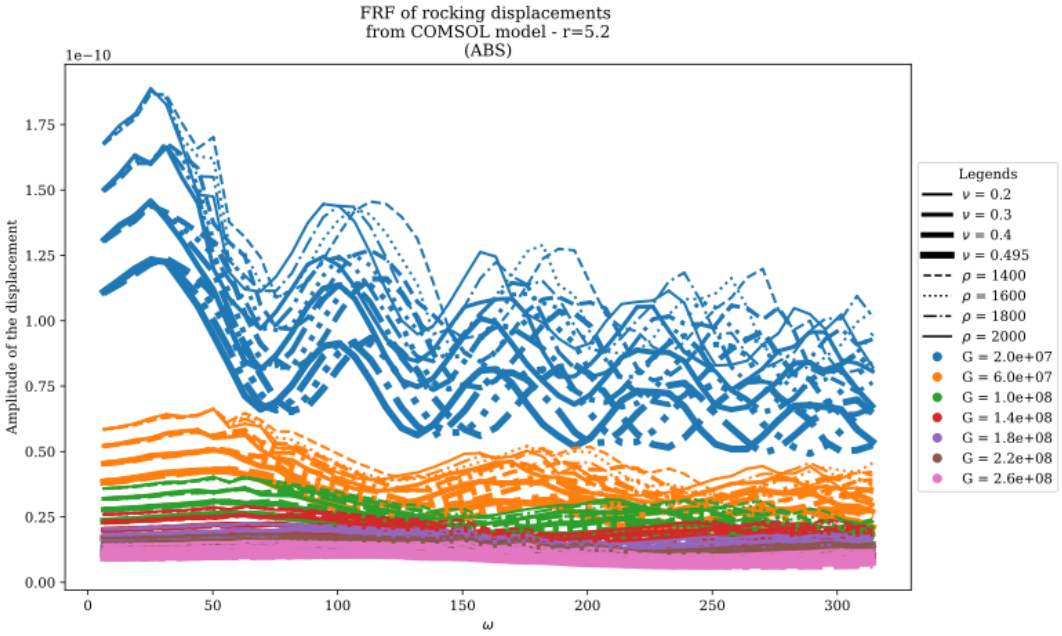


Figure A.6: Results of rocking motion for all soil combinations with $r_0 = 5.2$ m.

B

Target, predicted and optimisation graphs

Real and imaginary FRF of case in Section 4.2.4

These graphs show the comparison between the target, predicted and optimisation FRFs of the case presented in Section 4.2.4. They show that both the real and imaginary displacement are predicted and optimised successfully.

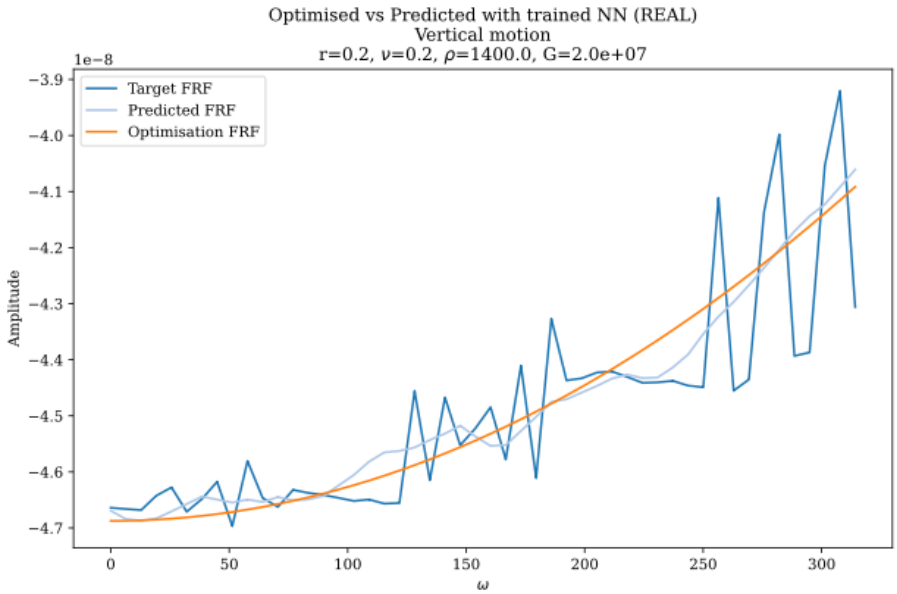


Figure B.1: Comparison of the real FRF of vertical motion with 1DoF optimisation.

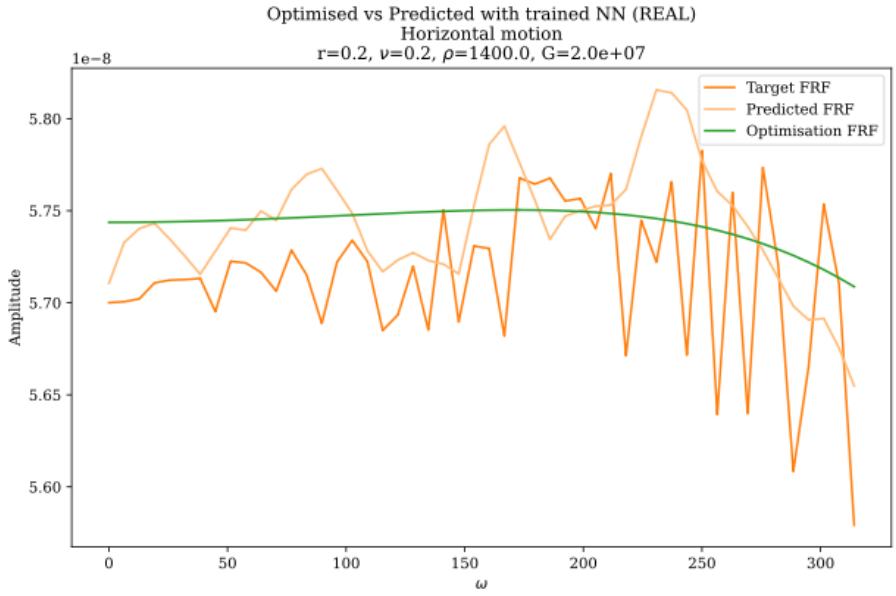


Figure B.2: Comparison of the real FRF of horizontal motion with 1DoF optimisation.

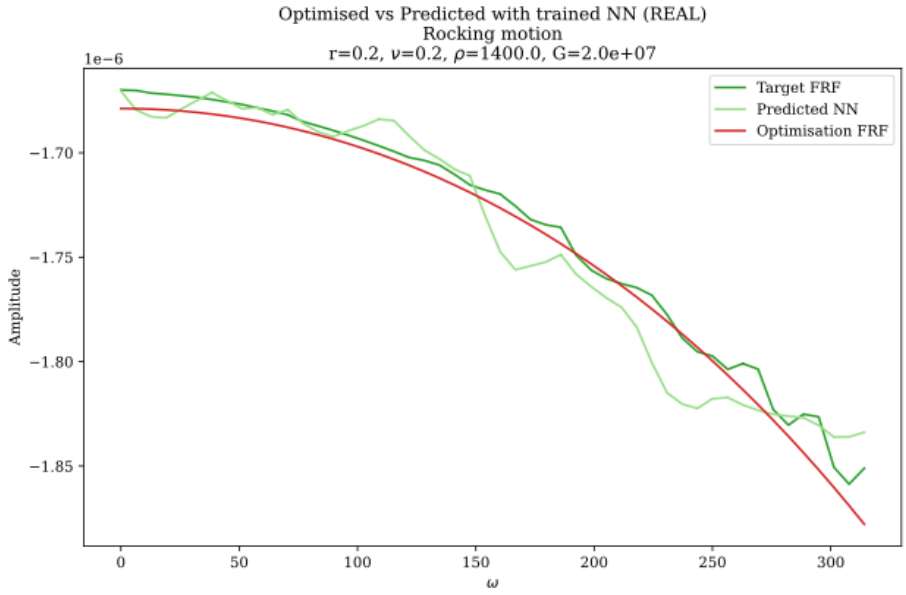


Figure B.3: Comparison of the real FRF of rocking motion with 1DoF optimisation.

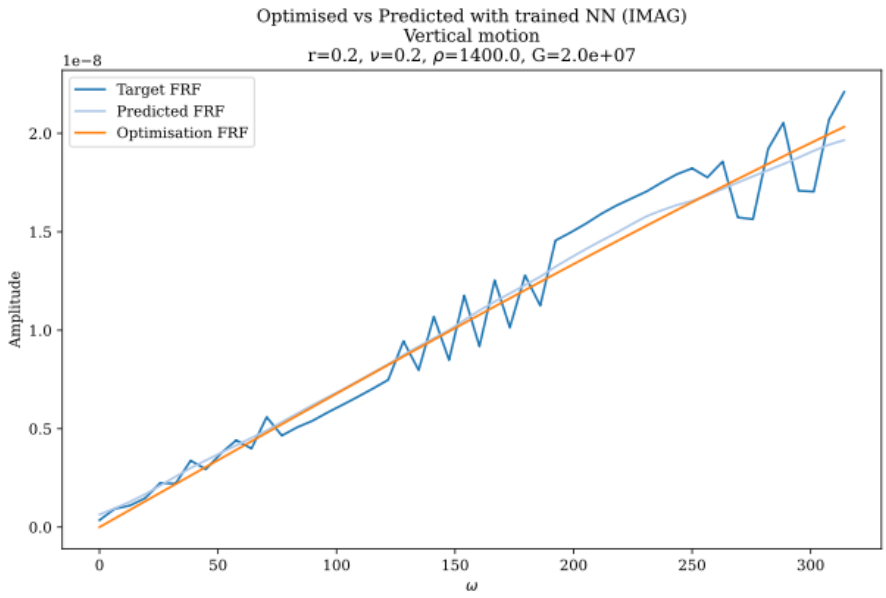


Figure B.4: Comparison of the imaginary FRF of vertical motion with 1DoF optimisation.

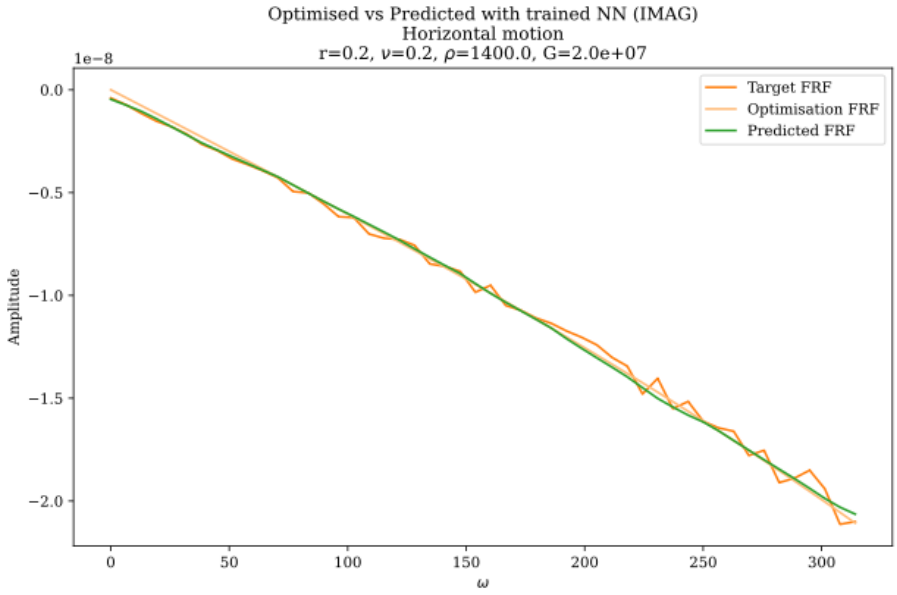


Figure B.5: Comparison of the imaginary FRF of horizontal motion with 1DoF optimisation.

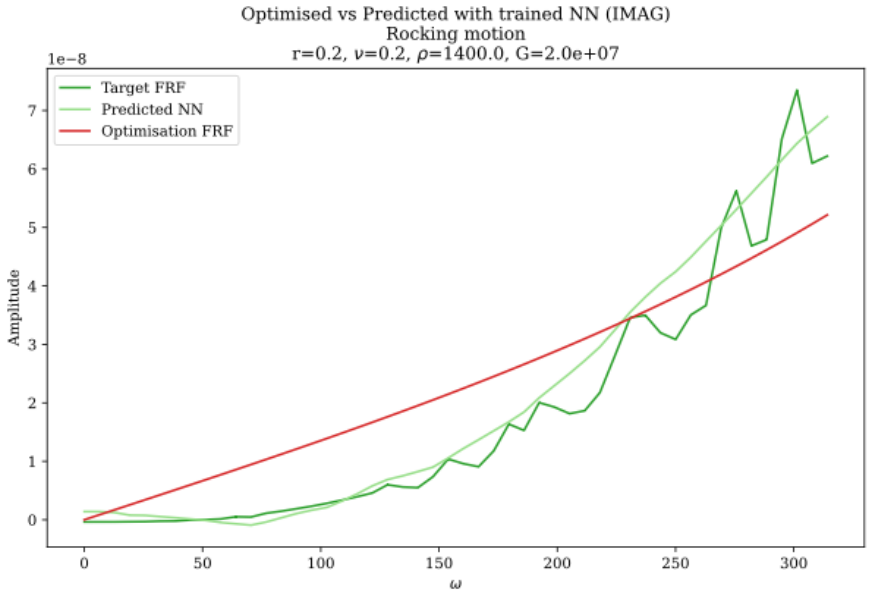


Figure B.6: Comparison of the imaginary FRF of rocking motion with 1DoF optimisation.

Absolute FRFs of additional cases

The following graphs show the target, predicted and optimised responses of cases with different soil and foundation characteristics. Each graph presents one of the three analysed motions.

Case 2 Figures B.7, B.8, B.9 show the results for a second case. It can be observed in the vertical and horizontal motions that predicted FRF followed the target very closely. Similarly, the optimised response is accurate. However, the rocking motion does not align very well. Once again, this indicates the need for further tuning the NN and optimisation process.

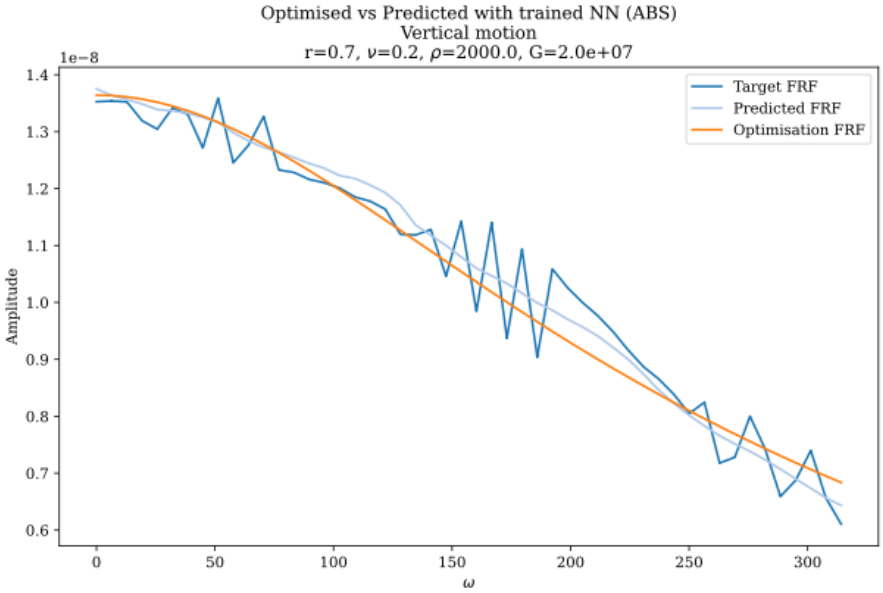


Figure B.7: Case 2 - Comparison of the absolute FRF of vertical motion with 1DoF optimisation.

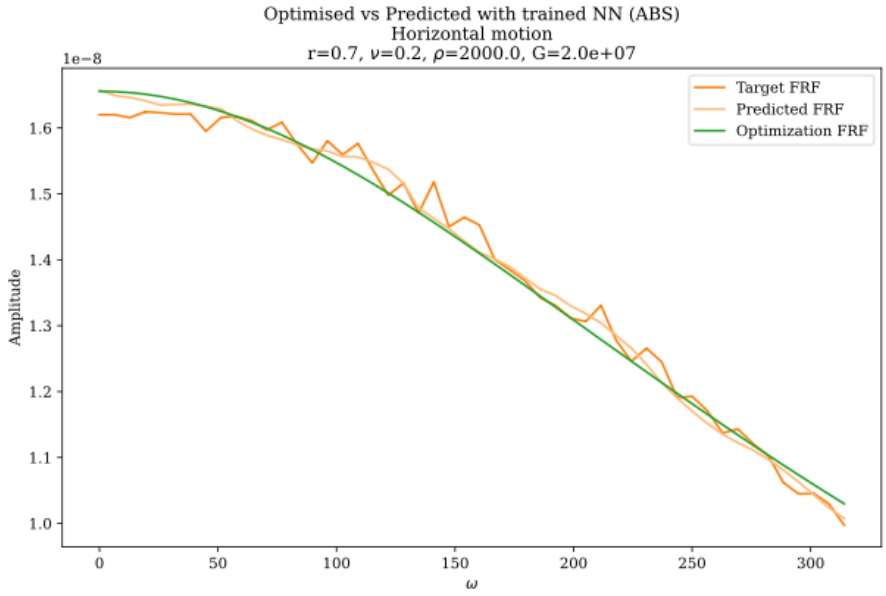


Figure B.8: Case 2 - Comparison of the absolute FRF of horizontal motion with 1DoF optimisation.

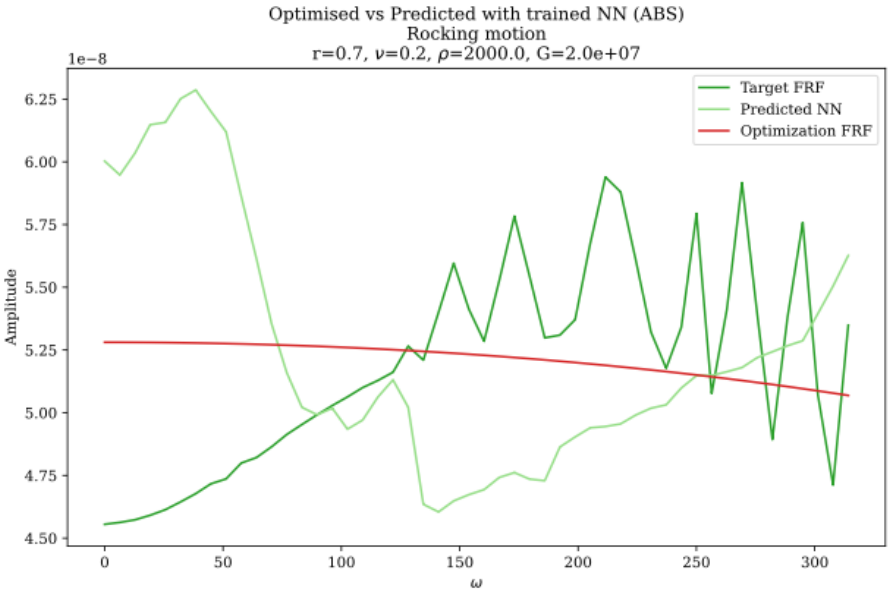


Figure B.9: Case 2 - Comparison of the absolute FRF of rocking motion with 1DoF optimisation.

Case 3 Figures B.10, B.11, B.12 show the results for a third case. It can be observed that the horizontal motion is predicted and optimised almost exactly. The vertical motion is predicted very well but not so well optimised. Finally, the rocking motion is predicted poorly. Once again, this indicates the need for further tuning the NN and optimisation process.

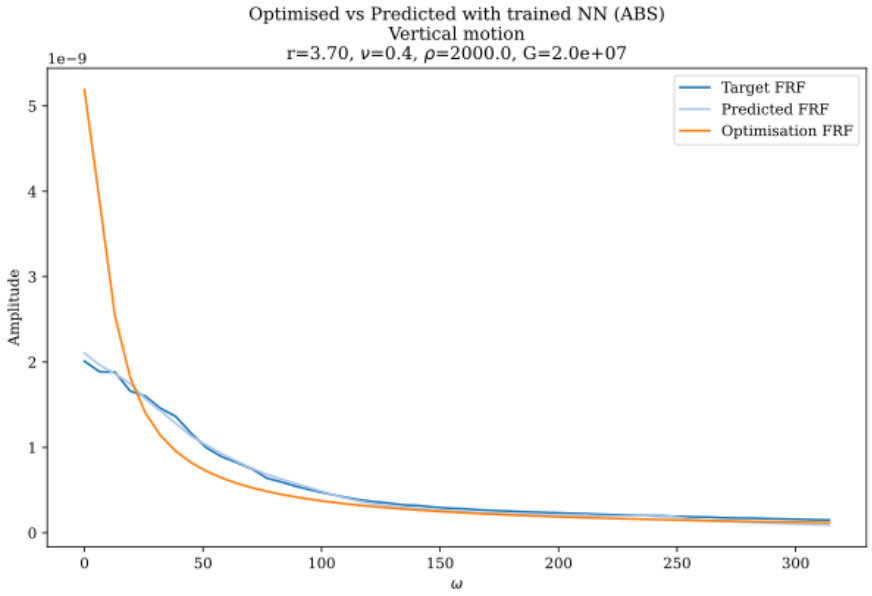


Figure B.10: Case 3 - Comparison of the absolute FRF of vertical motion with 1DoF optimisation.

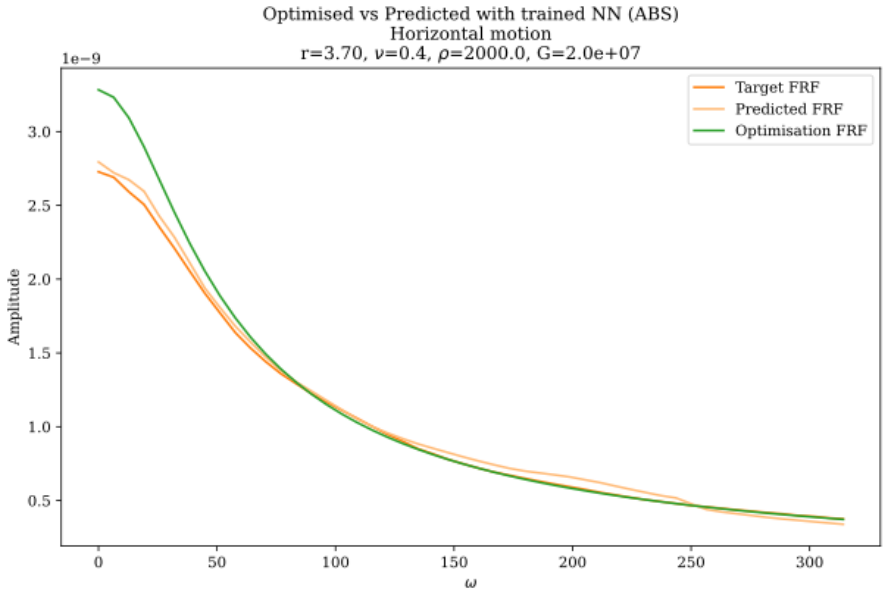


Figure B.11: Case 3 - Comparison of the absolute FRF of horizontal motion with 1DoF optimisation.

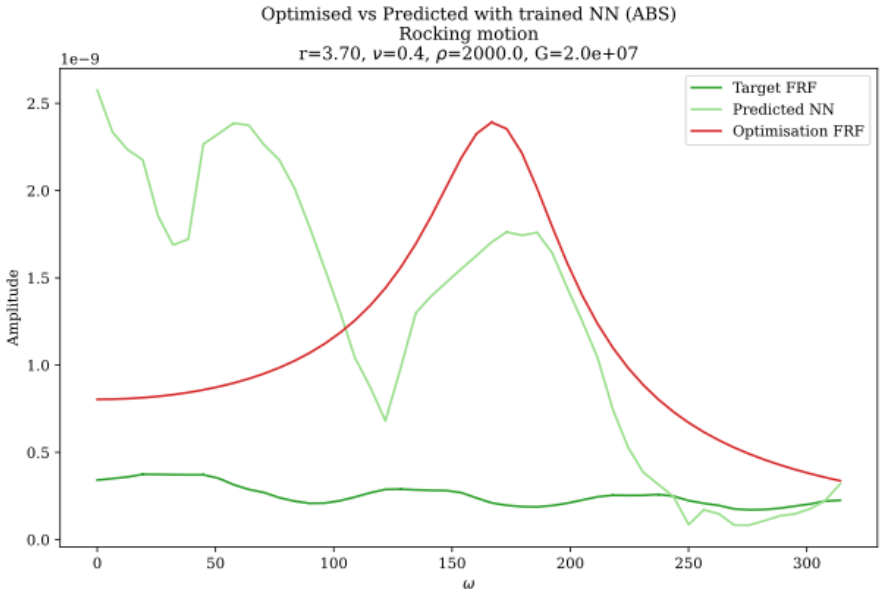


Figure B.12: Case 3 - Comparison of the absolute FRF of rocking motion with 1DoF optimisation.

C

Python code for NN

This Appendix shows some of the code snippets used for this thesis as a reference for future works.

C.1. General packages and libraries

The packages used in the work are shown below.

```
"""
Used Python packages and libraries
"""

import numpy as np
import torch
import torch.nn as nn
import torch.optim as optim
import sys
import time
import itertools
import scipy
import matplotlib.pyplot as plt
import copy
import mph
from torch.utils.data import DataLoader, TensorDataset, random_split
from scipy import optimize
from scipy.optimize import least_squares
from sklearn.preprocessing import StandardScaler
from matplotlib import rc
from cycler import cycler
from itertools import product
```

C.2. Data generation

To generate the data, a communication package between COMSOL and Python, called MPh was used. A sample snippet used for the process is shown below.

```
"""
Loop to generate and run each model combination in COMSOL. It also extracts the
    results from the FE software and saves them.
"""
# Load the COMSOL model from a directory
model = client.load('directory')
# Print the model parameters (geometric characteristics of the foundation, soil
    parameters, domain size...) to revise them.
params_dict = model.parameters()
print(f'\n\nModel␣for␣frequencies␣' + model.parameter('f_min') + '␣to␣' + model.
    parameter('f_max'))
for (name, value) in model.parameters().items():
    description = model.description(name)
    print(f'{name}␣=␣{value}␣-␣{description:20}')

# %%
'Loop␣through␣the␣radius␣(r0)␣of␣the␣foundation,␣Poisson's ratio (nu_soil),
    density (rho_soil) and shear modulus (G_soil) of the soil to run a COMSOL model
     for each combination. These arrays must be previously defined depending on the
     requirements of the analysis.'

for␣rr␣in␣r0:
␣␣␣␣for␣nn␣in␣nu_soil:
␣␣␣␣␣␣␣␣for␣rho␣in␣rho_soil:
␣␣␣␣␣␣␣␣␣␣␣␣for␣GG␣in␣G_soil:
␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣model.parameter('r0',␣rr)
␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣model.parameter('nu_soil',␣nn)
␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣model.parameter('rho_soil',␣rho)
␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣model.parameter('G_soil',␣GG)
␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣model.mesh()
␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣model.solve('Study 1')
␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣freq_min␣=␣model.parameter('f_min')
␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣freq_max␣=␣model.parameter('f_max')
␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣freq_range␣=␣f'f{freq_min}_{freq_max}'
␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣model.export('w2',␣f'w_disc_r{rr*100:.0f}_nu{nn}_rho{rho}_G{GG
    /100000:.0f}_{freq_range}.txt')
```

C.3. Functions for NN and optimisation

These are the functions used to lead the information for running the Neural Networks. Only the most relevant ones are shown in this Appendix.

Input array

```
"""
Functions to create the input array for the NN and optimisation.
"""
def import_soil_parameters():
    """
    This function imports the foundation and soil ranges of the Disc. It is
        particular for each problem, this is an example. The ranges were also
        defined for the particular problem and analysis.
    OUTPUT:
    r0, nu_soil, rho_soil, G_soil: arrays of every parameter
    """
```

```
    "[m], radius of foundation disc"
    r0_min, r0_max, r0_step = 0.2, 5, 0.5
    r0 = np.arange(r0_min, r0_max + r0_step, r0_step)

    "[-], Poisson's ratio"
    nu_soil = np.array([0.2, 0.3, 0.4, 0.495])

    "[kg/m^3], soil density"
    rho_min, rho_max, rho_step = 1400, 2000, 200
    rho_soil = np.arange(rho_min, rho_max + rho_step, rho_step)

    "[Pa], shear modulus"
    cs_min, cs_max = 100, 400
    G_min, G_max = np.max(rho_soil) * cs_min**2, np.min(rho_soil) * cs_max**2
    dG = 4e7
    G_soil = np.arange(G_min, G_max + dG, dG)

    return r0, nu_soil, rho_soil, G_soil
```

DataLoader

```
"""
Function to load the information to the PyTorch NN.
"""

def createDataLoaders(dataset, seed, training, validating, batch_size=100):
    '''
    This function first creates the training, validation and testing datasets and
        then creates the DataLoaders to be used in Pytorch NN. Splits the input
        dataset into training validation and testing data.
    Parameters
      ----------
      dataset : TensorDataset(inputs_tensor, w_tensor)
      seed: for reproducibility
      training: value. Between 0.7 and 0.8
      validating: value. Between 0.1 and 0.2 (rest for testing)
      batch_size: value.

    Returns
      train_loader: training data set for the model to train on
      val_loader: validation data set for the model to validate
      test_loader: testing data
      ----------
    '''
    # Defines the sizes of the training, validation and testing sets
    train_size = int(training * len(dataset))
    val_size = int(validating * len(dataset))
    test_size = len(dataset) - train_size - val_size

    # Splits the datasets according to the sizes defined above
    train_dataset, val_dataset, test_dataset = torch.utils.data.random_split(
        dataset, [train_size, val_size, test_size],
        generator=torch.Generator().manual_seed(seed))

    # DataLoader to use multiple workers in parallel, define batch, allow
        shuffling
    train_loader = DataLoader(train_dataset, batch_size=batch_size, shuffle=True)
    val_loader = DataLoader(val_dataset, batch_size=batch_size, shuffle=False)
    test_loader = DataLoader(test_dataset, batch_size=batch_size, shuffle=False)

    return train_loader, val_loader, test_loader
```

C.4. Neural Network

```
"""
Functions to create, train and evaluate the NN.
"""
class FRFNet(nn.Module):
    """
    This class defines the architecture of the NN
    input_neurons, size_layer and output_neurons must be defined previously.
    """
    def __init__(self, lambda_value):
        super(FRFNet, self).__init__()
        self.fc1 = nn.Linear(input_neurons, size_layer)
        self.fc2 = nn.Linear(size_layer, size_layer)
        self.fc3 = nn.Linear(size_layer, size_layer)
        self.fc4 = nn.Linear(size_layer, output_neurons)
        self.lambda_value = lambda_value

    def forward(self, x):
        # Using the ReLU activation function
        x = torch.relu(self.fc1(x))
        x = torch.relu(self.fc2(x))
        x = torch.relu(self.fc3(x))
        w = self.fc4(x)
        return w



'Create an input array for evaluating a particular case. It will have 5 columns: r0, nu, rho, G and omegas. It will have as many rows as omegas defined for the case'

r0_case = 0.7    # particular for the case
nu_case = 0.22   # particular for the case
rho_case = 1954  # particular for the case
G_case = 8e7     # particular for the case
samples_case = 1000
omegas_case = 2*np.pi*np.linspace(start_omega, stop_omega, samples_case)

# Create the input mesh for the case and normalize with the same scaler used for the training set.
input_case = create_input_mesh_omegas(r0_case, nu_case, rho_case, G_case, omegas_case)
inputs_case_norm = inputs_scaler.transform(input_case)
inputs_case_tensor = torch.tensor(inputs_case_norm, dtype=torch.float32)

# Calls the network and predicts with the case array
net.eval()  # set the network to evaluation mode
predictions_temp = []
w_test = net(inputs_case_tensor)
predictions_temp.append(w_test.detach().numpy())
predictions_case = np.concatenate(predictions_temp, axis=0)
```