NTNU – Trondheim
Norwegian University of
Science and Technology

# Non-linear time domain analysis of vortex induced vibrations

Jan Vidar Ulveseter

Marine Technology
Submission date: July 2015
Supervisor: Carl Martin Larsen, IMT

Norwegian University of Science and Technology
Department of Marine Technology

## M.Sc. thesis 2015

**for**

# Jan Vidar Ulveseter

## NON-LINEAR TIME DOMAIN ANALYSIS OF VORTEX INDUCED VIBRATIONS

Vortex induced vibrations (VIV) have to be accounted for in design of a large variety of slender marine structures. Research on VIV during the last decades has been substantial, but we still have to accept that uncertainties in prediction of response amplitudes are significant. One important contributor to this uncertainty is damping – both quantification of damping and the influence from damping on VIV amplitudes. Another uncertainty is the non-linear effects for a structure with seafloor contact like a catenary riser and free spanning pipelines.

The only way of accounting for non-linear interaction between the oscillating pipeline and the sea bottom is by non-linear time domain analysis. So far almost all VIV analyses are carried out in frequency domain. However, a new time domain approach has been proposed by Mads Thorsen, and by combining this method with a non-linear finite element model it becomes possible to investigate the importance of damping and non-linear contact forces at the shoulders of free spanning pipelines.

The purpose of the present project is to study non-linear contact and damping mechanisms for free spanning pipelines by using a time domain model for VIV.

The work may be carried out in steps as follows:

1. Literature study that should concentrate on basic theory for time domain VIV, non-linear FEM analysis as applied for free spanning pipelines. Part of this study has already been carried out during the master pre project.
2. Describe the interaction mechanisms for pipes with bottom contact and how contact forces and damping can be accounted for in a VIV analysis.
3. Make a simple MATLAB code for analysis of a free spanning pipeline with non-linear springs and dampers at the shoulders.
4. Combine this model with Thorsen's time domain model for VIV.
5. Use the new method on selected cases, and compare results to results from traditional frequency domain analyses and previous calculations from a combination of frequency and time domain analyses.

The work may show to be more extensive than anticipated. Some topics may therefore be left out after discussion with the supervisor without any negative influence on the grading.

The candidate should in her/his report give a personal contribution to the solution of the problem formulated in this text. All assumptions and conclusions must be supported by mathematical models and/or references to physical effects in a logical manner.

The candidate should apply all available sources to find relevant literature and information on the actual problem.

The report should be well organised and give a clear presentation of the work and all conclusions. It is important that the text is well written and that tables and figures are used to support the verbal presentation. The report should be complete, but still as short as possible.

The final report must contain this text, an acknowledgement, summary, main body, conclusions and suggestions for further work, symbol list, references and appendices. All figures, tables and equations must be identified by numbers. References should be given by author name and year in the text, and presented alphabetically by name in the reference list. The report must be submitted in two copies unless otherwise has been agreed with the supervisor.

The supervisor may require that the candidate should give a written plan that describes the progress of the work after having received this text. The plan may contain a table of content for the report and also assumed use of computer resources.

From the report it should be possible to identify the work carried out by the candidate and what has been found in the available literature. It is important to give references to the original source for theories and experimental results.

The report must be signed by the candidate, include this text, appear as a paperback, and - if needed - have a separate enclosure (memory stick or DVD/ CD) with additional material.

Trondheim, February 2015

Cm. Larsen

Carl M. Larsen

Deadline: August 1st, 2015

# Preface

This is a Master Thesis carried out at the Department of Marine Technology, at NTNU, in the spring of 2015. The topic is vortex induced vibrations (VIV) of slender marine structures, focusing on nonlinear time domain analysis of free spanning pipelines. The topic was chosen based on what was done in the Projects Thesis (Ulveseter, 2014), and can be considered an extension of that work.

The PhD student Mats Thorsen has been working on a linear time domain model for calculations of vortex induced vibrations since 2012. Professor Carl Martin Larsen, my supervisor, thought it would be interesting to apply Thorsen's work on a structural model of a pipeline, taking into account nonlinear soil-pipe interaction. I wanted to to code my own program instead of using commercial software. This Master Thesis made it possible to develop a new calculating program, based on Thorsen's work, and thus we agreed on this project.

The readers of this report should be familiar with basic theory of dynamic analysis. Frequency domain, time domain, eigenfrequency and dynamic equilibrium are some of the terms assumed known in advance. Concerning VIV, basic concepts are presented. However, the language is technical. It is thus a great advantage to have knowledge about hydrodynamics and marine structures, to understand the technical expressions used throughout the text.

Trondheim, 2015-07-30

Jan Vidar Ulveseter

# Acknowledgment

There are two persons who have made a significant contribution to the present work. Professor Carl Martin Larsen has given weekly advices and recommendations, both concerning the literature study and the development of the nonlinear program. The PhD student, Mats Thorsen, has made it possible to perform the analysis part of this thesis. He has given me access to his MATLAB code, explaining his model and also given general advices during the project period. I am very grateful for their support.

I would like to thank the members of office A 1.019, who has contributed to a friendly and good working environment throughout the project period. This has made the year a joyful experience. I will miss them all, and wish them the best for the future to come. Lastly, I would like to thank my family for their support.

O.N.

(JVU)

# Abstract

In this thesis a model for nonlinear analysis of vortex induced vibrations (VIV), applied to free spanning pipelines, is proposed. The developed model takes into account the soil-pipe interaction. Through case studies, conclusions can be drawn toward the influence of nonlinear soil-pipe interaction, compared to linear prediction tools. Soil-pipe interaction is important to consider because of VIV induced bending stresses at the pipe shoulders. The fatigue damage at the span shoulders can be a limiting factor for lifetime estimations of pipelines. Hence a reliable model, including nonlinear soil dampers and soil springs, is necessary for realistic results.

The nonlinear analysis program is created in MATLAB. It is based on Mats Thorsen's linear time domain VIV model for analysis of simple supported beams. To make it applicable for free spanning pipelines, a seabed profile and the pipeline static configuration, must be found. RIFLEX is used to find the static configuration. When the seabed and static configuration is known, soil dampers and soil springs can be added to the vertical translation degrees of freedom at positions of soil penetration.

The proposed program has the option of a linear or a nonlinear analysis. They are referred to as Ulveseter's linear model and Ulveseter's nonlinear model, respectively. If the linear analysis is chosen, the stiffness and damping matrices are based on the static configuration, and are constant throughout the time integration. By updating the stiffness and damping matrices for every time step of the time integration, it is possible to switch the soil springs and soil dampers off, when the pipeline lifts up from the seabed. They can be turn on again in case of seabed penetration. This is how the nonlinear soil-pipe interaction is established.

Ulveseter's nonlinear model is compared to Ulveseter's linear model and VIVANA. Through six case studies predictions of response amplitudes and stress amplitudes are found for a variation of pipeline data, seabed profiles, soil stiffness values, soil damping values and current velocities. The results show that the stress amplitudes at the pipe shoulders are reduced for the nonlinear model. The soil damping has a small influence on the response predictions for most cases. The seabed curvature around the pipe shoulders influences the touch down point position for Ulveseter's nonlinear model. Less seabed curvature makes the touch down point vary more than in case of a large curvature. Increasing soil stiffness is observed to increase the stress amplitudes. It can also trigger a mode shift, as seen for Ulveseter's linear model. Increased current velocity results in a mode shift of the response shape. Ulveseter's model compares well to VIVANA, but predicts in most cases somehow smaller stress and response values than VIVANA.

Concerning the influence of soil damping, it may depend on the pipeline properties, seabed profile and current conditions. It is only the first case study where the soil damping influence is of significance for all VIV models. This case is based on a shorter pipe, with less bending stiffness, with less end tension facing a stronger current, than the other cases. Also, the seabed profile has a smaller curvature at the pipe shoulders than the rest of the cases. However, Case 1

is based on an extreme current velocity and the pipe tension is low compared to bending stiffness and and dimensions of the pipe. Hence, for the more realistic cases, the soil damping is of less importance for the response. In Case 3 and 4, the soil damping effect is larger for Ulveseter's nonlinear model, than the linear prediction tools. However, the result might be influenced by transient effects.

The proposed nonlinear program is limited to look at cross-flow motion only. Axial degrees of freedom are neglected because they are considered to be of secondary importance. The only nonlinearity accounted for is the soil-pipe interaction. Tension variations, large displacements and nonlinear material properties are not considered. A fully nonlinear analysis should include both static and dynamic forces in the dynamic analysis. Because the proposed model uses the static configuration found from RIFLEX, this is not possible to do in a consistent way. Therefore an approximation is performed to account for the effect of gravity. To improve the present work, Ulveseter's model should be modified to account for these limitations.

# Sammendrag

I denne avhandlingen er en modell for ikke-lineær analyse av virvelinduserte vibrasjoner (VIV) foreslått. Den er begrenset til analyse av frittspente rør. Modellen tar hensyn til havbunn-rør interaksjon. Ved å utføre case-studier kan vi trekke slutninger om hvordan ikke-lineær havbunn-rør interaksjon påvirker responsen sammenlignet med lineære modeller. Havbunn-rør interaksjon er viktig å ta hensyn til på grunn av VIV induserte bøyespenninger på rørets skuldre. Tretthetskader på skuldrene kan være en begrensende faktor for levetiden til rørledninger. Derfor må en pålitelig modell inkluder ikke-lineære bunndempere og bunnfjærer, for realistiske resultater.

Det ikke-lineære analyseprogrammet er laget i MATLAB. Det er basert på Mats Thorsens lineære tidsdomene VIV modell for analyse av fritt opplagte bjelker. For at Thorsens modell skal kunne gjøre beregninger på rørledninger i frie spenn må det etableres en havbunnsprofil og en tilhørende statisk konfigurasjon. Denne delen av analysen er gjort i RIFLEX. Når statisk konfigurasjon er kjent kan vi legge til bunnfjærer og bunndempere ved nodene som penetrerer havbunnen.

Programmet har mulighet til både en lineær og en ikke-lineær analyse. De er referert til som Ulveseters lineære og ikke-lineære modell. I den lineære analysen er stivhets og dempingsmatrisene konstante gjennom tidsintegrasjonen av den dynamiske likevektsligningen. Dermed må havbunnsstivhet og dempning inkluderes i modellen før den dynamiske analysen begynner. Dette gjøres basert på den statiske konfigurasjonen. For den ikke-lineære modellen blir stivhets og dempningsmatrisen oppdatert for hvert tidskritt. Hver node sjekkes for havbunnspenetrasjon. Hvis vi har havbunnskontakt blir en bunnfjær og en bunndemper lagt til stivhetsmatrisen og dempningsmatrisen. Hvis vi ikke har bunnkontakt vil programmet sjekke neste node uten å legge til en bunnfjær og en bunndemper.

Ulveseters ikke-lineære modell er sammenlignet med Ulveseters lineære modell og VIVANA. Gjennom seks case-studier finnes responsamplituder og spenningsamplituder for varierende rørledningskarakteristikk, bunnprofiler, bunnstivhetsverdier, bunndempningsverdier og strømhastigheter. Resultatene viser at spenningsamplitudene i skuldrene er redusert for den ikke-lineær modell. Bunndemping har en liten innvirkning på responsen i de fleste tilfeller. Havbunnens krumning rundt skuldrene påvirker berøringspunktet for Ulveseters ikke-lineære modell. Mindre krumning gir større variasjon av berøringspunktet. Økende bunnstivhet gir større spenningsamplituder. Det kan også utløse et modeskifte, som er observert for Ulveseters lineære modell. Øket strømhastighet resulterer i et modeskifte. Ulveseter modell sammenligner godt med VIVANA, men spår i de fleste tilfeller mindre spenning og responsverdier.

Effekten av bunndempning avhenger av rørledningskarakteristikk, bunnprofil og strømforhold. Det er bare i Case 1 at bunndempingen er av signifikant betydning for responsen till alle VIV modellene. Denne casen er basert på et kortere rør, med mindre bøyestivhet, med mindre strekk som møter en sterkere strøm enn de andre tilfellene. Dessuten har bunnprofilen en

mindre krumning ved rørskuldrene enn i resten av tilfellene. Case 1 er basert på en ekstrem strømhastighet og rørets strekk er lav sammenlignet med bøyestivhet og dimensjoner til røret. Derfor er konklusjonen at bunndemping har liten betydning for responsen i de mer realistiske studiene. I Case 3 og 4 er den ikke-lineære modellen mer påvirket av bunndempning, enn de lineære modellene. Det er konkludert med at dette antakeligvis skyldes transiente effekter.

Det foreslåtte ikke-lineære programmet er begrenset til å se på bevegelser i vertikal retning. Aksielle frihetsgrader er neglisjert fordi de anses å være av underordnet betydning. Den eneste ikke-lineariteten i programmet er havbunn-rør interaksjonen. Strekkvariasjoner, store forskyvninger og ikke-lineære materialegenskaper er ikke inkludert i modellen. En full ikke-lineær analyse bør inneholde både statiske og dynamiske krefter i den dynamisk analysen. Fordi den foreslåtte modellen bruker statisk konfigurasjon funnet fra RIFLEX, er ikke dette mulig å gjøre på en tilfredsstillende måte. Derfor er en tilnærmelse utført for å ta hensyn til virkningen av tyngdekraften. For å forbedre dette arbeidet bør Ulveseters modell endres for å ta hensyn til disse begrensningene.

# Notation

## Symbols

### General

| | | |
|---|---|---|
| $U$ | [m/s] | Undisturbed flow velocity |
| $D$ | [m] | Diameter of cross section |
| $L$ | [m] | Length of pipe |
| $T$ | [N] | End tension |
| $EI$ | $[Nm^2]$ | Bending stiffness |
| $\nu$ | $[m^2/s]$ | Kinematic viscosity of water |
| $\rho$ | $[kg/m^3]$ | Water density |
| $f_v$ | $[s^{-1}]$ | Vortex shedding frequency |
| $f_n$ | $[s^{-1}]$ | Eigenfrequency in still water for mode number n |
| $f_{osc}$ | $[s^{-1}]$ | Response frequency |
| $\omega$ | [rad/s] | Oscillation frequency |
| m | [kg/m] | Dry mass of cylinder per unit length |
| $m_{a0}$ | [kg/m] | Added mass of cylinder in still water |
| $m_a$ | [kg/m] | Added mass of cylinder in actual flow conditions |
| $w_s$ | [kg/m] | Submerged weight of pipe per unit length |
| $k_s$ | $[N/m^2]$ | vertical soil stiffness per unit length |
| $c_s$ | $[Ns/m^2]$ | vertical soil damping per unit length |

### Thorsen's VIV model

| | | |
|---|---|---|
| $y_0$ | [m] | Cross-flow amplitude |
| $\phi_{exc,y}$ | [-] | Instantaneous phase of cross-flow excitation force |
| $\phi_{\dot{y}}$ | [-] | Instantaneous phase of cross-flow velocity |
| $C_v$ | [-] | Empirical coefficient for amplitude dependent force |

**VIVANA**

| | | |
|---|---|---|
| $C_{e,CF}$ | [-] | Excitation force coefficient in cross-flow direction |
| $C_{e,IL}$ | [-] | Excitation force coefficient in in-line direction |
| $C_e$ | [-] | Excitation force coefficient in cross-flow or in-line direction |

## Non-dimensional parameters

| | |
|---|---|
| $\hat{f} = \frac{Df_{osc}}{U}$ | Non-dimensional frequency |
| $U_r = \frac{U}{Df_n}$ | Reduced velocity |
| $St = \frac{Df_v}{U}$ | Strouhal number |
| $Re = \frac{UD}{\nu}$ | Reynolds number |
| $\frac{A}{D}$ | Amplitude ratio |
| $\lambda = \frac{c_s}{c_{cr}}$ | Estimated vertical soil damping ratio |

## Abbreviations

| | |
|---|---|
| VIV | Vortex Induced Vibrations |
| CF | Cross-flow, normal to inflow |
| IL | In-line, parallel to inflow |
| CFD | Computational Fluid Dynamics |
| FEM | Finite Element Method |
| dof | degree of freedom |

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

## 1.1 Background

Slender marine structures exposed to a current may experience vortex induced vibrations (VIV). It is a phenomenon important to include in dynamic analyses of risers and pipelines because it causes fatigue damage. It is a complex phenomenon were research has been substantial for several decades. Still, there are large uncertainties related to response predictions. The uncertainty makes it necessary to do conservative estimates of VIV. Vortex shedding suppression devices, such as helical strakes on risers, might be over-used as a consequence of this. Suppression devices increase cost and makes the installation more demanding. Better VIV models can potentially save the industry for a lot of money.

SHEAR7 is a VIV prediction program widely used by the industry. It is limited to treat a linearized problem because it solves the dynamic equilibrium equation in frequency domain. Risers and pipelines are highly nonlinear structures. Tension variations and soil-pipe interaction are important nonlinear effects that must be included in an analysis of vortex induced vibrations, for reliable results. To include these effects, a time domain model for the VIV forces must be coupled to a nonlinear structural model. Several attempts have been made to find a good time domain force model, but with varying results. However, Mats Thorsen, a PhD student at the Department of Marine Technology at NTNU, is developing a time domain VIV model, which looks promising. If his force model is coupled to a nonlinear structural model, nonlinearities can be included in the VIV analysis.

One of the uncertainties of VIV for pipelines is the pipe-soil interaction, and the corresponding soil damping. The soil-pipe interaction depends on the properties of the soil, which are not easily obtained. If realistic values of the soil stiffness and soil damping are used in a nonlinear soil-pipe interaction model, more knowledge of VIV for free spanning pipelines can be gained. In this Master Thesis, the nonlinear soil-pipe interaction for free spanning pipelines is to be investigated. It is only cross-flow motion that is considered, when the pipeline is subjected to a uniform current flow. We are interested in the pipeline response for different pipeline properties, seabed profiles, current velocities, soil stiffness and soil damping values. Comparing the nonlinear VIV model with linear models, it is interesting to see and to discuss the differences.

### Literature Survey

Describing basic concepts of VIV, (Larsen, 2011) gives a good overview of the state of the art. (Sumer and Fredsøe, 2006) gives information about experimental results with cylinders oscillating in air and water. This is directly related to empirical VIV models, using experimental data to describe the loads from the water. One of these empirical VIV programs is VIVANA. It solves the dynamic equilibrium equation in frequency domain (Passano et al., 2014).

Several time domain VIV models have been developed since the 1990s. (Lie, 1995), (Finn et al., 1999) and (Mainçon and Larsen, 2011), are the most important ones. Also, the concept of wake oscillators (Zarantonello and Brikhoff, 1957) is the basis for some time domain VIV analyses. Lastly, Mats Thorsen has published several papers based on his time domain VIV model. (Thorsen et al., 2014), (Thorsen et al., 2015a) and (Thorsen et al., 2015b) describe the theoretical foundation of the model, and results from case studies.

Concerning VIV on pipelines in particular, (Larsen et al., 2004) and (Larsen and Koushan, 2005) suggests a nonlinear time domain method. The effect of nonlinear soil-pipe interaction is investigated with respect to stress amplitudes at the pipeline shoulders. The nonlinear soil-pipe interaction is complex, and (Veritas, 2006) is the Recommended Practice, for how to determine values of the soil stiffness and soil damping. (Lie et al., 2001) and (Larsen and Passano, 2006) gives information about how to include soil-pipe interaction in a dynamic analysis. Even though Computational Fluid Dynamics (CFD) is considered immature for VIV predictions due to the computational cost, attempts have been made to analyze pipelines using 2D planes. This was done by (Halse, 1997).

To create a nonlinear time domain analysis program we need to know the theory of dynamic analysis. (Langen and Sigbjörnsson, 1979) is a book describing in detail general methods of how to perform dynamic analyses. (Larsen, 2014) does not contain the same level of details on nonlinear dynamic analyses. However, it discusses the differences between linear and nonlinear analyses of flexible pipes in particular, which is relevant for analyses of free spanning pipelines.

## 1.2 Objectives

The purpose of the present work is to study the effect of nonlinear soil-pipe interaction for free spanning pipelines experiencing VIV. Both nonlinear soil springs and soil dampers are of interest. This is to be done using the time domain VIV model proposed by Mats Thorsen combined with a self-made nonlinear structural FEM model.

The objectives are as follows:

- Present basic theory for VIV, with focus on time domain VIV combined with nonlinear FEM as applied for free spanning pipelines
- Describe interaction mechanisms for pipes with bottom contact and how contact forces and damping can be accounted for in a VIV analysis
- Make a simple MATLAB program for analysis of free spanning pipelines with nonlinear springs and dampers at the shoulders, and combine it with Thorsen's time domain VIV model
- Use the developed method on selected cases and compare with frequency domain solutions, and other time domain solutions

## 1.3 Limitations

The study is limited to 2D only considering cross flow VIV. The pipeline is facing a uniform current, so we do not look at the effect of the the boundary layer induced by the fluid-soil interaction. The nonlinearity in the model is limited to nonlinear soil-pipe interaction. Other nonlinearities, like varying tension and large displacements are not considered. Also, axial translation degrees of freedom are not included in the structural model because they are of secondary importance.

The developed nonlinear time domain VIV program is compared to results from VIVANA and time domain VIVANA/RIFLEX analyses. We do not look at how the models compare to experimental results, or to other VIV prediction tools. The study treat varying soil damping, soil stiffness, seabed profile, current velocity and pipeline characteristics in six separate case studies. For more reliable trends and results, more cases should be investigated.

## 1.4 Approach

To fulfill the objectives, a good theoretical foundation of VIV, dynamic analysis, pipelines and soil-pipe interaction is needed. Hence, an extensive literature survey is performed. From this, the two first objectives focusing on presenting theory, can be met. Parts of this was already covered in (Ulveseter, 2014).

Mats Thorsen's linear model is the starting point for the third objective. With this program, a linear time domain VIV analysis of a simple supported beam can be analyzed. To make the program applicable for pipelines, a seabed and a static configuration must be found. The static configuration can be found in RIFLEX. Based on the pipeline position relative to the seabed, springs and dampers can be applied to the vertical translation degree of freedom (dofs) where the pipe initially is penetrating the seafloor. Applying the Newmark-$\beta$ time integration scheme that was already in Thorsen's code, the linear pipeline response can be found.

A nonlinear time domain analysis where the nonlinearity is related to pipe-soil interaction, has time varying stiffness and damping matrices. The soil damping and soil stiffness must be turned off when the pipeline rises from the soil, and switched on again if the pipeline penetrates the seafloor. This can be taken into account by updating the stiffness and damping matrices for every time step of the time integration. Applying a nonlinear scheme for time integration, as outlined in (Langen and Sigbjörnsson, 1979), the nonlinear pipeline response is the output. Thus, the third objective can be fulfilled.

The third objective is, by far, the most time consuming objective. Time is spent on:

- Understanding Thorsen's original program
- Learning how to use RIFLEX/VIVANA
- Debugging the code, and make sure the different functions work as they should
- Convergence testing
- Making an understandable code

The forth objective is related to testing the new program, referred to as Ulveseter's model, against other VIV prediction tools. VIVANA and RIFLEX are MARINTEK programs developed in cooperation with the Marine Technology Department at NTNU. It thus natural that VIVANA is used as the other VIV prediction tool. In (Larsen et al., 2004) nonlinear time domain analysis of VIV on free spanning pipelines is presented. Pipeline dimensions, and case studies can be based on what was done in this paper.

## 1.5 Structure of the Report

Chapter 2 introduces VIV in general. What causes VIV and what parameters influence the VIV response are questions to be answered. First, we look at the fixed cylinder case and explain the hydrodynamics when the cylinder faces a current. Then, VIV modeling based on different solution methods is presented, with a main focus on empirical VIV models. The empirical models are based on experiments, and two traditional VIV experiments are discussed. The experiment with a rigid pipe with flexible support shows lock-in, and how the vortex shedding frequency, still water eigenfrequency and oscillation frequency are all important concepts in VIV. Then, the experiment with a rigid pipe given forced harmonic motion is explained. Chapter 2 is to a large extent based on (Ulveseter, 2014).

Chapter 3 presents the theory behind VIVANA, the MARINTEK developed VIV model. We look at the different steps of the analysis, from finding the static configuration using RIFLEX, to calculate the VIV response solving the dynamic equilibrium equation in frequency domain. Added mass model, excitation force model, excitation zones and damping model are discussed and how they relate to experimental results is presented. This chapter is also, based on (Ulveseter, 2014).

Chapter 4 describes different time domain VIV models. What theories they are based upon, and their limitations. Thorsen's time domain VIV model is presented in Chapter 5. The force model, both in-line and cross-flow is described in detail. The synchronization of the oscillation frequency and the vortex shedding frequency is explained, which is the main difference between this model and other VIV models. The linear structural model is also presented.

Chapter 6 goes more into details about free spanning pipelines. It is explained why a linear VIV model is not suited for analyses of free spanning pipelines, and important nonlinearities are discussed. We see examples on how it is possible to perform VIV analyses on pipelines, with focus on the soil-pipe interaction. This interaction is described in form of soil stiffness and soil damping, and it is suggested how these parameters can be quantified and implemented in an analysis. Parts of this is taken from (Ulveseter, 2014).

Chapter 7 gives information about the developed nonlinear VIV program for analysis of free spanning pipelines. We are introduced to the theoretical foundation of the nonlinear dynamic analysis. Many pages are used to discuss the limitations of the program, and thus how reliable the predictions are. Then, the MATLAB code is presented, showing the different functions created and the build-up of the program. To make it easier to understand, flowcharts are used to visualize the program build-up, and the logic behind it.

Chapter 8 is where the case studies are presented. The first case is an extension of the case in (Ulveseter, 2014), while the other cases are based on pipeline properties and seabed configurations as applied in (Larsen et al., 2004). During the six cases, we see how pipeline response is affected by pipeline properties, seabed profile, soil stiffness, soil damping and current velocity. All cases include a comparison between the developed nonlinear program, the developed linear program and VIVANA.

Chapter 9 discusses the findings in Chapter 8. The results are discussed with respect to what is physically expected. Also, trends regrading the soil damping influence is presented. Lastly, current velocities and pipeline properties as applied in the different case studies are discussed with respect to how realistic they are. Chapter 10 is the conclusion and recommendations for further work.

# Chapter 2

# Introduction to Vortex Induced Vibrations

Vortex induced vibrations (VIV) is an important phenomenon to account for in design of long slender marine structures, such as risers and pipelines. When subjected to a current, vortices will develop and be shed in the wake of the structure. These vortices give rise to forces, causing vibrations. VIV can be a limiting factor considering the lifetime of the slender marine structure. This is a result of the fatigue fatigue damage caused by the oscillatory VIV motion, inducing oscillatory stresses.

## 2.1 Hydrodynamics for a fixed circular cylinder

In describing the hydrodynamics around a fixed circular cross section, it is common to refer to hydrodynamic numbers. The value of Reynolds number (Re), Strouhal number (St) and the roughness parameter (Δ ) will influence the flow situation, and are defined as follows:

$$Re = \frac{UD}{\nu} \tag{2.1}$$

$$St = \frac{Df_v}{U} \tag{2.2}$$

$$\Delta = \frac{k}{D} \tag{2.3}$$

Reynolds number gives the relationship between inertia forces and frictional forces (Steen and Minsaas, 2013). It will influence the flow characteristics in form of laminar-turbulent transition, flow separation and the vortex street. This is shown in Figure 2.1.

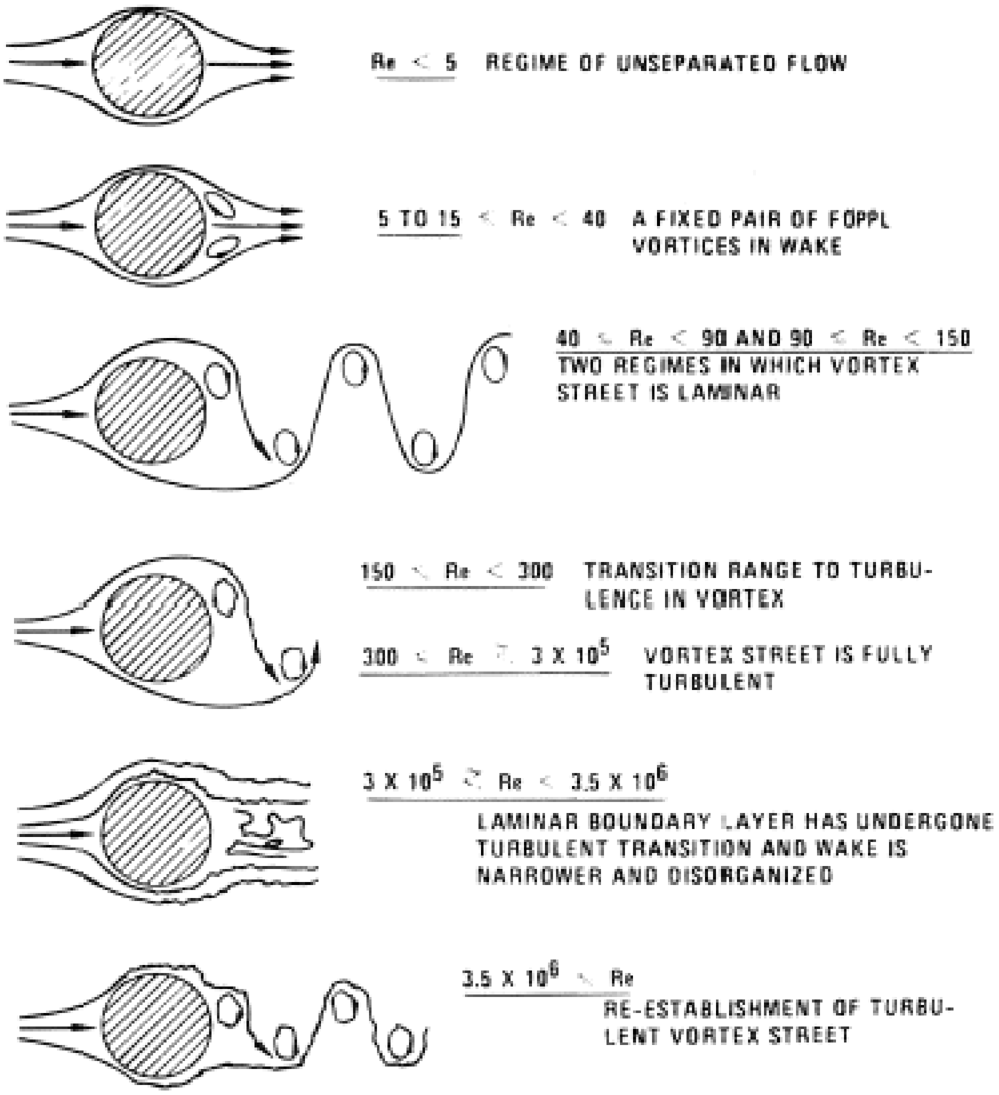

Re < 5 REGIME OF UNSEPARATED FLOW
5 TO 15 < Re < 40 A FIXED PAIR OF FÖPPL VORTICES IN WAKE
40 < Re < 90 AND 90 < Re < 150 TWO REGIMES IN WHICH VORTEX STREET IS LAMINAR
150 < Re < 300 TRANSITION RANGE TO TURBULENCE IN VORTEX
300 < Re < 3 X 10^5 VORTEX STREET IS FULLY TURBULENT
3 X 10^5 < Re < 3.5 X 10^6 LAMINAR BOUNDARY LAYER HAS UNDERGONE TURBULENT TRANSITION AND WAKE IS NARROWER AND DISORGANIZED
3.5 X 10^6 < Re RE-ESTABLISHMENT OF TURBULENT VORTEX STREET

Figure 2.1: Flow around a circular cross section for different Re (Sunden)

The Strouhal number gives the relationship between the diameter of the cross section, the inflow velocity and the vortex shedding frequency. Strouhal number is a function of Reynolds number and the roughness parameter as indicated in Figure 2.2. In the subcritical flow regime St $\approx 0,2$ and the value will not significantly change for other flow regimes if the pipe has a rough surface. This is often the case in the marine environment.

The forces induced by the vortex shedding are shown in Figure 2.3. The vortex shedding will cause the velocity profile around the cross section to change. Vortex shedding from one side of the cylinder results in an opposite circulation. The change of velocity around the cylinder results in a pressure change. This gives a force resultant contributing to a force component in-line

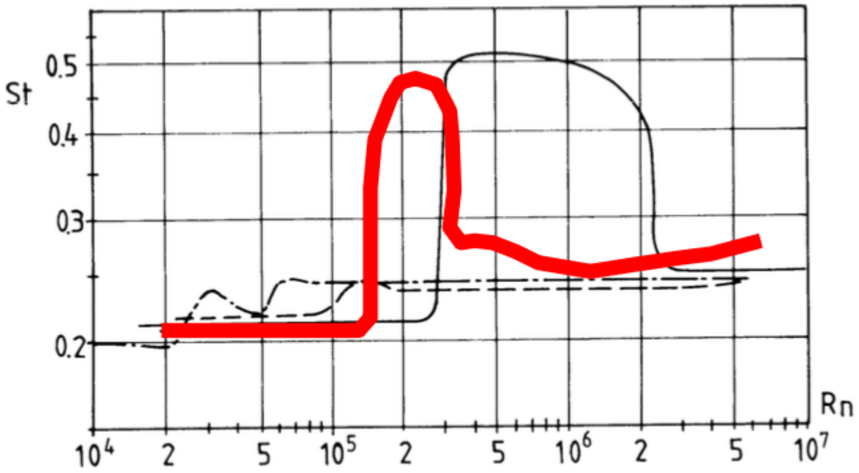


Fig. 6.26. Strouhal number St of rough circular cylinders in steady incident flow for different surface roughness values $k/D$ ($k$ = average height of surface roughness, $D$ = cylinder diameter, $\mathrm{St} = f_v D/U_\infty$, $f_v$ = vortex shedding frequency, $U_\infty$ incident flow velocity, $\mathrm{Rn} = U_\infty D/\nu$.) ——, smooth —···—, $k/D = 7.5 \cdot 10^{-4}$ — —, $k/D = 3 \cdot 10^{-3}$; —·—, $k/D = 9 \cdot 10^{-3}$; ----, $k/D = 3 \cdot 10^{-2}$. (Achenbach & Heinecke, 1981.)

Figure 2.2: Strouhal number for rough and smooth cylinder surface as function of Re (Larsen, 2011). Originally in (Faltinsen, 1990)

with the inflow (IL) and a component perpendicular to the inflow (CF). After half a vortex shedding period, a vortex will shed from the other side of the cylinder. This causes the lift force to change direction. The direction of the drag force will be independent on vortex shedding being from the upper or lower side of the cylinder. Thus the drag force will oscillate with two times the vortex shedding frequency.

## 2.2 Modeling of VIV

To perform a VIV analysis the challenge is related to modeling of the hydrodynamic forces acting on the slender structure. There are two different approaches for this:

- Computational Fluid Dynamics (CFD)
- Empirical methods

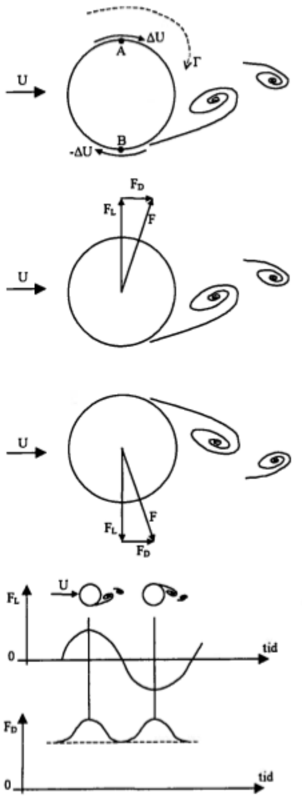


Figure 2.3: How drag forces and lift forces vary in time for a circular cross section (Pettersen, 2007)

The first method (CFD) is to solve the Navier-Stokes equations in a numerical way to get the pressure and velocity distribution in the fluid area around the structure. It is common to use Finite Element Method (FEM), Finite Difference Method or Finite Volume Method as the numerical tool. Today such an analysis is very time consuming and hence the use is limited.

The second method is to model the hydrodynamic forces based on empirical coefficients. The reliability of the VIV prediction of this method will depend on added mass coefficient, dynamic force coefficient, average drag coefficient and average lift coefficient. The hydrodynamic coefficients must be determined through experiments.

There exists a lot of empirical data from experiments with cylinders. This data is typically from the subcritical flow regime, Re $\epsilon(300, 300000)$. For a real size riser facing a high stream velocity, we might exceed the subcritical flow regime. If we in this case choose to use empirical coefficients from the subcritical flow regime, it is considered conservative on cases where Re is higher (Larsen, 2011). It represents, however, an uncertainty in the empirical model, and even though it is expensive an time consuming, large scale experiments might make the empirical VIV models more reliable.

Today, most tools for calculation of VIV are based on empirical hydrodynamic coefficients, since CFD-based programs demand a lot of computational power and time. Some of the existing programs are:

- SHEAR7
- VIVA
- VIVANA
- ABRAVIV
- ViCoMO
- Norske Hydro

The models apply different methods to solve the dynamic equilibrium equation. There are two main solution procedures. As can be seen in Figure 2.4, we distinguish between solution in frequency domain and time domain.

A frequency domain solution is limited to treat a linearized problem. Pipelines and risers behave nonlinearly due to soil-pipe interaction. This effect is impossible to investigate for solution in frequency domain. An upside applying a frequency domain solution is the use of empirical data. From experiments, empirical coefficients are found as function of frequency. This makes the empirical data easy to fit a frequency domain VIV model.

A time domain solution gives rise to more possibilities because linearization is not needed. It is possible to potentially implement the VIV model in advanced structural FEM models, like RIFLEX or SIMLA. Then, all relevant nonlinearities related to pipelines and risers in operation can be investigated. However, a time domain solution is complex, due to the complexity of VIV as a phenomenon. The complexity and the time integration process can make the code slower than a frequency domain method, and thus not as attractive for the industry as an engineering tool. However, the largest problem has been to create a reliable force model in time domain. Several attempts have been made, with varying success. In Chapter 4 and 5, different time domain models are presented and discussed.

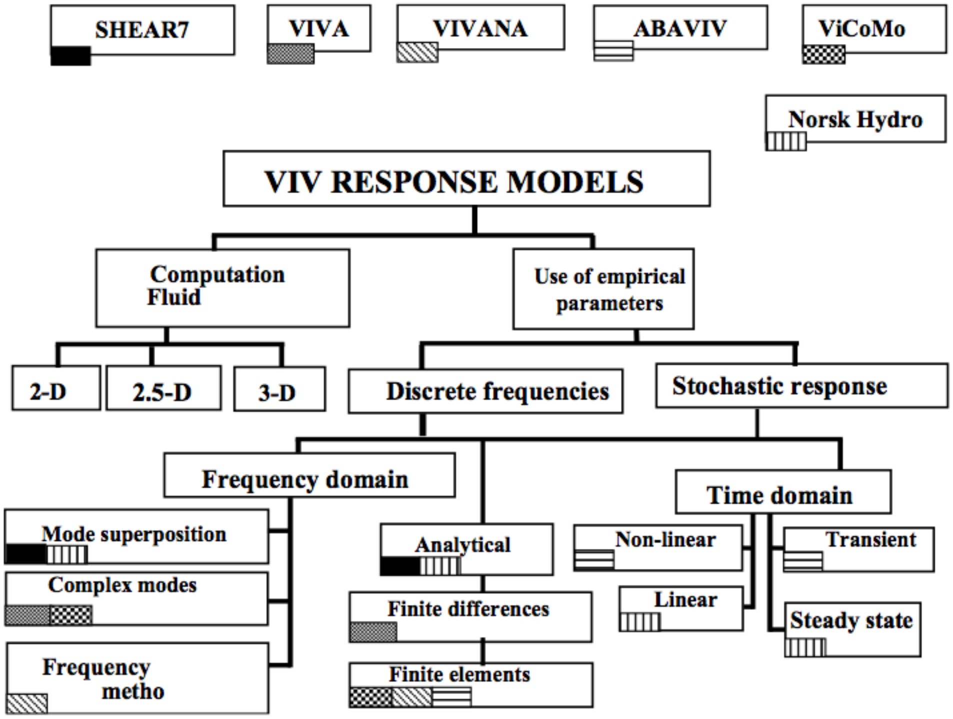


Figure 2.4: Solution method for empirical VIV models (Larsen, 2011)

## 2.3 Experiments and empirical coefficients

Hydrodynamic coefficients used in empirical VIV models are found through experiments. There are several types of experiments that can be performed. In the following we will look at two cases:

- Rigid pipe with flexible supports in constant flow
- Rigid pipe with forced harmonic motions

### Rigid pipe with flexible supports in constant flow

The experimental setup is schematically shown in Figure 2.5, for IL- and CF-direction. An experiment with a rigid pipe with flexible supports will give us information about amplitudes of motion and phase angle between excitation force and response. It is also possible to find added mass coefficients when the oscillation frequency is measured, applying Equation 2.7. However, it is more common to find added mass from forced harmonic motions as described in the next section. Running the experiment for different current velocities, the amplitudes, phase angels,

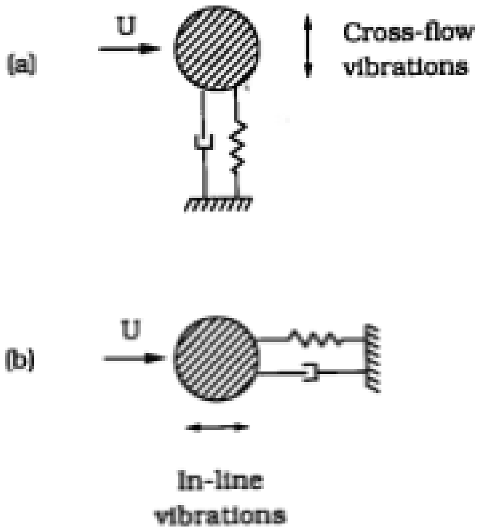


Figure 2.5: Schematic illustration of experimental setup for rigid pipe with flexible supports (Sumer and Fredsøe, 2006)

vortex shedding frequency and vibration frequency can be found as function of reduced velocity, defined as:

$$U_r = \frac{U}{Df_n} \tag{2.4}$$

The three important frequencies in this case are:

Still water frequency:

$$f_n = \frac{1}{2\pi}\sqrt{\frac{k}{m+m_{a0}}} \tag{2.5}$$

Vortex shedding frequency:

$$f_v = \frac{StU}{D} \tag{2.6}$$

Response/vibration frequency:

$$f_{osc} = \frac{1}{2\pi}\sqrt{\frac{k}{m+m_a}} \tag{2.7}$$

Here, k is the spring stiffness, m is the mass of the cylinder, $m_{a0}$ is the added mass in still water and $m_a$ is the added mass for the actual flow and oscillation situation.

Results from an experiment of a rigid pipe in water (Sumer and Fredsøe, 2006) are given in Figure 2.6. What is shown is the response frequency, vortex shedding frequency and corresponding motion amplitudes as function of reduced velocity, in the cross flow direction. The plot illustrate the phenomenon called lock-in. We see that for a wide range of reduced velocities the vortex shedding frequency and the vibration frequency attaches to one another. It is seen that, at a reduced velocity ≈ 3, the vibration frequency attaches to the vortex shedding frequency and the motion amplitudes start to increase. At reduced velocity ≈ 6 the vortex shedding frequency no longer follows Equation 2.6, but instead it attaches to the eigenfrequency of the structure. As the reduced velocity is further increased the eigenfrequency of the structure changes. This is due to added mass, which is dependent on the actual flow and oscillation process. In the lock-in range the vibration frequency is the eigenfrequency corrected for changing added mass, and lock-in is thus a true resonant phenomenon. The vortex shedding frequency still follows the eigenfrequency of the structure until the reduced velocity has a value ≈ 8. Then the vortex shedding again follows Equation 2.6, and the motion amplitude decreases.

The vibration frequency is a compromise between the eigenfrequency in still water and the vortex shedding frequency. Where, in this range the vibration frequency is, depends on the dry mass of the cylinder. For a heavy cylinder, the added mass contribution is relatively less important than for a light cylinder. Thus, a heavy cylinder will have a vibration frequency closer to the eigenfrequency in still water. The lock-in range is also affected by the dry mass. A light cylinder will have a wider lock-in range due to the changing added mass.

VIV is a self-limiting process. For two different initial conditions, we can see how the VIV motions change with time. What is observed is that for larger initial amplitude than in steady state, the response decreases in time until it reaches the steady state value. For lower initial amplitude than in steady state, the response increases until steady state is reached. This can be seen from Figure 2.7. We get back to how VIV is self-limiting in Section 3.1.2, looking at how the excitation force coefficients in VIVANA depend on response amplitude.

## Rigid pipe with forced harmonic motions

From full scale measurements it is possible to find the trajectory of a 2D cross section of slender marine structures. It is possible to do an analysis where, instead of measuring responses from given forces, we measure forces from given responses. Thus we can find the force contributions in phase with acceleration and velocity when we know the response of the structure. Then added mass coefficient, in phase with acceleration, and excitation force coefficients, in phase with velocity, can be found.

It is normal to do such experiments with pure harmonic IL- or CF-response, and plot the re-

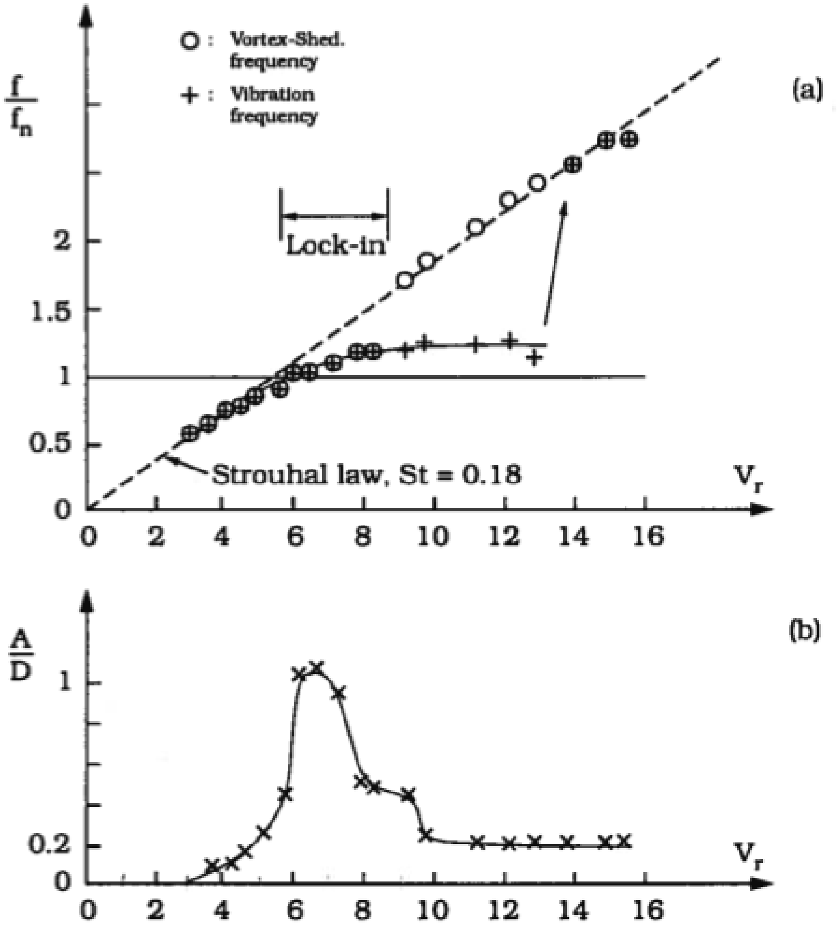


Figure 2.6: Experimental results for cross flow vortex induced vibrations (Sumer and Fredsøe, 2006)

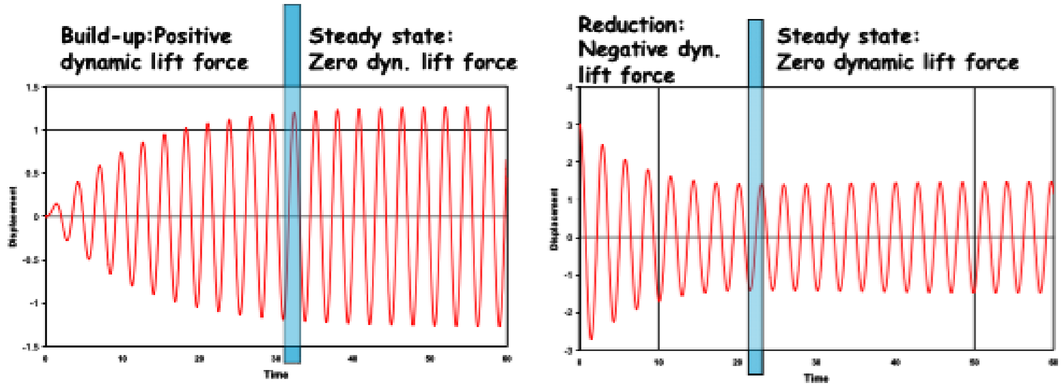


Figure 2.7: Time history of response oscillation for a cylinder with two different initial conditions (Larsen, 2011)

sulting coefficients as contour-plots dependent on the amplitude ratio $\frac{A}{D}$ and non dimensional frequency f̂.

$$\hat{f} = \frac{f_{osc}D}{U} \tag{2.8}$$

We can write the known harmonic response, velocity and acceleration as:

$$r = x_0 sin(\omega t), \dot{r} = \omega x_0 cos(\omega t), \ddot{r} = -\omega^2 x_0 sin(\omega t) \tag{2.9}$$

The inertia force and the hydrodynamic force are the only forces acting on the cylinder with forced harmonic motions. We have no restoring force and no damping force. The sum of the inertia force and the hydrodynamic force must thus equal the driving force, which can be measured.

$$M(-\omega^2 x_0 sin(\omega t) + F_H sin(\omega t + \epsilon) = \text{Driving force} \tag{2.10}$$

Now, splitting the hydrodynamic force in a component in phase with velocity and acceleration, we get:

$$(M + M_A)(-\omega^2 x_0 sin(\omega t)) + F_e(\omega x_0 cos(\omega t)) = \text{Driving force} \tag{2.11}$$

Thus we can find added mass $M_A$ and excitation force $F_e$.

This experimental procedure, with pure harmonic IL- and CF-response, is the basis for added mass and excitation force coefficients in VIVANA (Section 3.1.1 and Section 3.1.2 ). In the newer versions of VIVANA, empirical data for combined CF and IL VIV is given. This data is based on observed trajectories for cross sections of a flexible beam, and is also found through experiments with forced motions.

# Chapter 3

# Frequency domain VIV model

By the industry, semi empirical VIV tools in frequency domain are considered state of the art. Two typical models used are SHEAR7 and VIVANA. Even though the main focus of the present work is related to time domain models, case studies are performed in Chapter 8 comparing frequency domain and time domain solutions. VIVANA is used as the frequency domain program. It is thus an advantage to know the theoretical basis of the program. This chapter gives an overview of the solution procedure implemented in VIVANA, and the build-up of the program.

## 3.1 VIVANA

MARINTEK has developed VIVANA for analyses of VIV for slender marine structures. It is based on solution of the dynamic equilibrium in frequency domain. An iteration scheme is applied to generate consistency between excitation forces and the response. IL and CF analyses are typically performed separately, but from VIVANA version 3.7, we also have access to hydrodynamic coefficients from combined IL and CF experiments. Thus it is possible to perform a combined CF and IL analysis. The structure itself is a finite element model with beam or bar elements. The analysis is performed as follows (Passano et al., 2014):

1. **Static analysis**:
   VIVANA is dependent on the MARINTEK program RIFLEX. In RIFLEX we must first perform a static analysis to get the static configuration of the slender marine structure.

2. **Eigenvalue analysis in still water**:
   Modeshapes and corresponding eigenfrequencies are found for still water, solving the eigenvalue problem. The modeshapes are split into CF and IL contributions.

3. **Identification of possible excitation frequencies**:
   A subset of the eigenfrequencies are used as potential active response frequencies. Added mass depends on the motion of the structure, and eigenfrequencies depends on added

mass. Hence iteration must be performed to match the added mass with the frequencies of oscillation under VIV conditions. However, in the combined IL and CF case, it is assumed that the IL frequency is two times the CF frequency, and the iteration is not need.

4. **Finding excitation zones**:
   The response frequencies are given an excitation zone based on an interval of non-dimensional frequency. In this zone the active response frequency can excite the structure. In practical problems, excitation zones for different response frequencies may overlap. To avoid this VIVANA has the option between space sharing and time sharing.

5. **Calculate response**:
   Using the response frequencies found in step 3 and the excitation zones found in step 4, the response of the slender marine structure can be found using the frequency domain method. The solution has converged when we have consistency between nonlinear excitation forces, damping forces and response. The iteration scheme also gives consistent local response amplitude and the phase between local load and response.

6. **Post-processing**:
   In this step fatigue analysis and amplification of drag coefficients are performed. The results are stored on VIVANA result files.

### 3.1.1 Added mass model

The added mass coefficients, in pure CF and IL, are based on experiments with rigid cylinders given a harmonic motion. The response ratio and the frequency of oscillation are varied systematically. The result can be presented as a contour plot. It shows that added mass, in general, is dependent on both response amplitude and frequency of oscillation. However, in VIVANA, it is assumed that the response amplitude is less important than the frequency. In CF-analyses, a value of $\frac{A_{CF}}{D} = 0.5$ is chosen, as indicated in Figure 3.1. This makes added mass only dependent on frequency, as can be seen from Figure 3.2. A similar procedure is applied for pure IL-analysis.

For the case with combined CF and IL, the added mass coefficients used in VIVANA are based on experiments with forced motions of a rigid cylinder. However, the trajectory of the cylinder is based on observed trajectories for cross sections of a flexible beam.

### 3.1.2 Excitation force model

The excitation forces are in phase with the response velocity and will oscillate accordingly. The excitation force amplitude over the length $\Delta L$ is given as:

$$F_{e,CF} = \frac{1}{2}\rho C_{e,CF} D U^2 \Delta L \tag{3.1}$$

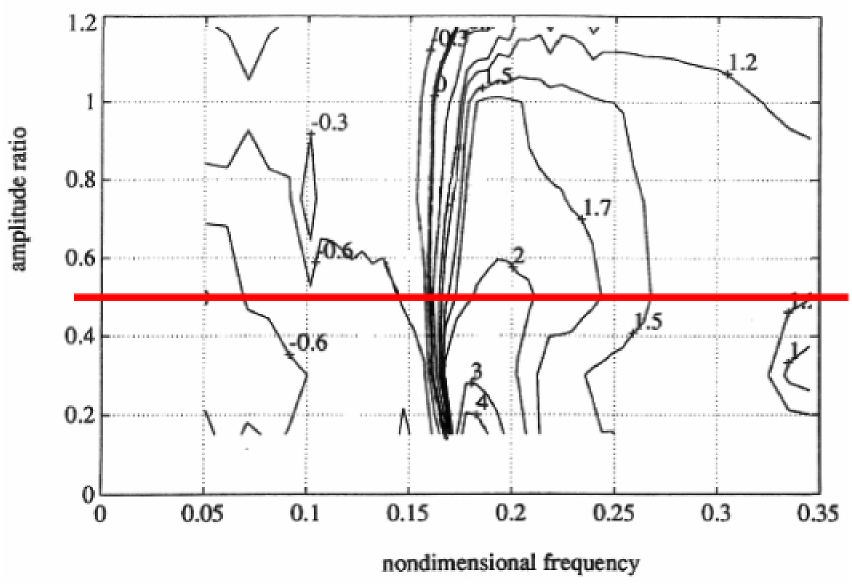


Figure 3.1: Contour plot of added mass coefficient as function of non-dimensional response amplitude and frequency of oscillation (Lie et al., 2008)

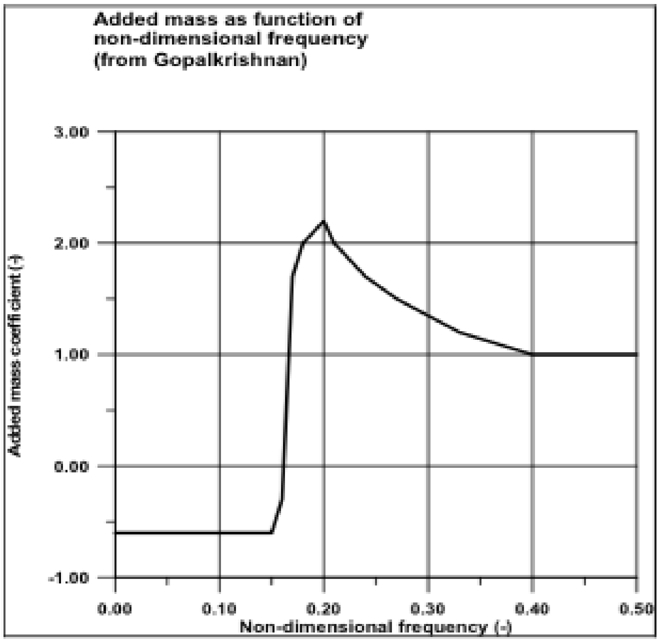


Figure 3.2: Simplified added mass coefficient only as function of non dimensional frequency used in VIVANA (Lie et al., 2008)

The same setup applies for the IL-direction.

As for added mass coefficients, the excitation force coefficients are based on experiments with rigid cylinders given a harmonic motion. The result is a contour plot of the excitation force coefficient shown in Figure 3.3. We know from item 3 in the list above (Section 3.1), that we only look at a subset of the eigenfrequencies when finding the active response frequencies. To find out which frequencies that are most important we look at the $C_{e,CF}$-values of largest magnitude. From Figure 3.3, we find this to be for $\hat{f} \in (0.125, 0.3)$.

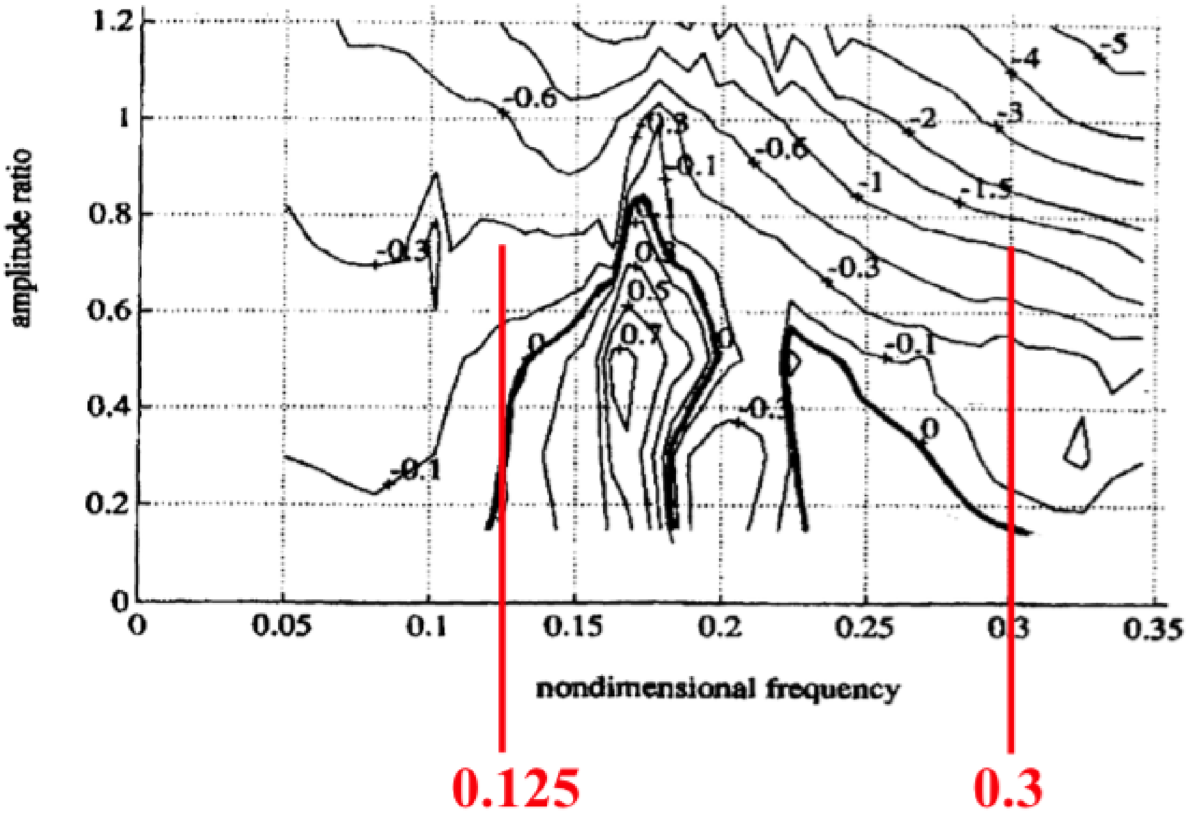


Figure 3.3: Contour plot of excitation force coefficient in CF-direction, with red lines pointing out frequency interval of importance, in VIVANA (Passano et al., 2014)

In VIVANA the excitation force coefficients are treated as function of the amplitude ratio. But the points A, B and C in Figure 3.4 and Figure 3.5 depend on non-dimensional frequency. The curves from A to B and B to C are given as second order polynomials. The points are defined as follows:

- Point A: Value of $C_e$ at $\frac{A}{D} = 0$
- Point B: The maximum value of $C_e$ at corresponding amplitude ratio
- Point C: Amplitude ratio where $C_e = 0$

From these curves (Figure 3.4 and Figure 3.5) we see that at $\frac{A}{D} > (\frac{A}{D})_{C_{e=0}}$ the excitation force coefficients become negative. This implies that we have damping instead of excitation. Physically, instead of the water giving energy to the structure, the structure gives energy to the water. Thus VIV is a self-limiting phenomenon.

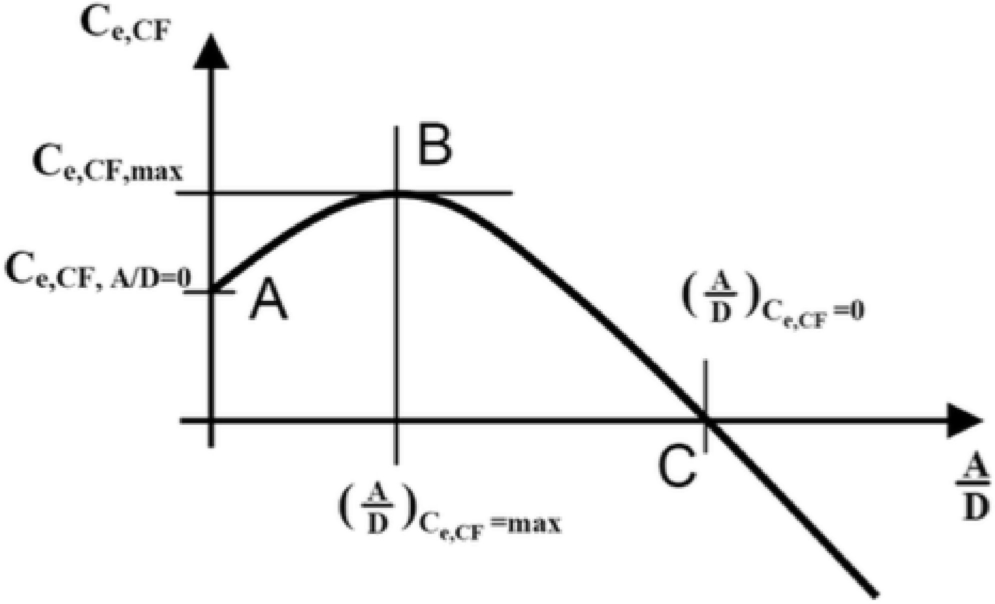


Figure 3.4: Excitation force coefficient in CF-direction used in VIVANA (Lie et al., 2008)

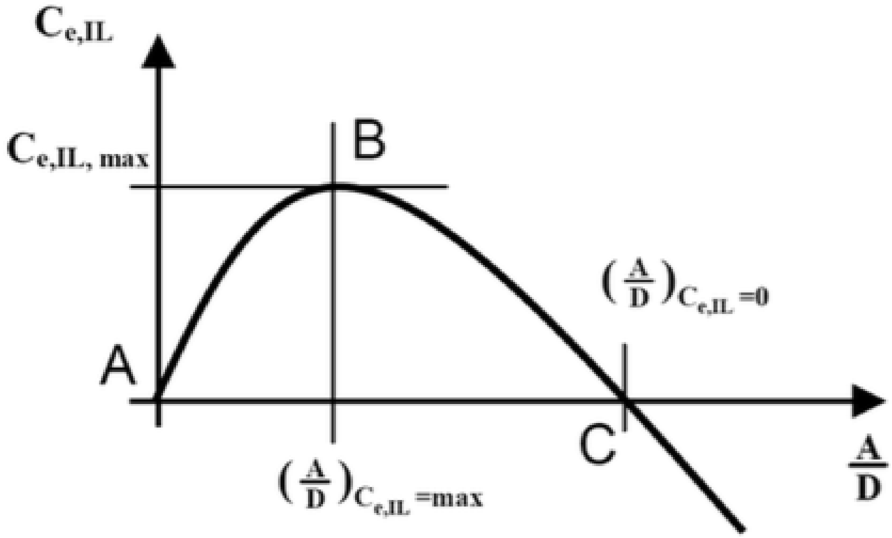


Figure 3.5: Excitation force coefficient in IL-direction used in VIVANA (Lie et al., 2008)

### 3.1.3 Excitation zones

In real life situations with a riser in sheared current, we will have a large number of different modeshapes and frequencies varying in time and space. To model this chaotic picture, VIVANA is based on the assumption that at a specific point on the riser, only one response frequency can be active, at a given time. This assumption is in good agreement with experiments. VIVANA provides the user with the choice between time sharing and space sharing, to make sure this criteria is satisfied. Overlapping excitation zones are shown in Figure 3.6

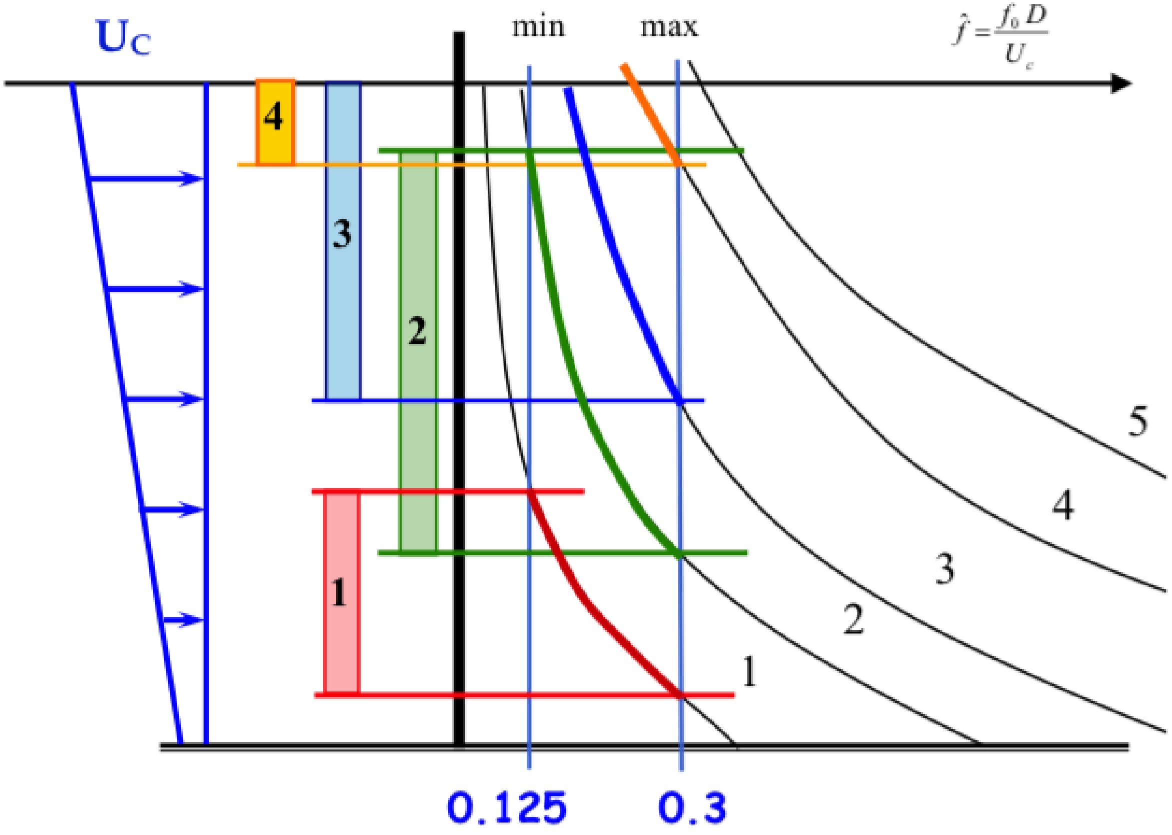


Figure 3.6: Overlapping excitation zones for a riser in sheared current (Passano et al., 2014)

The assumption of only one active frequency at a given point at a given time, is satisfied through an energy consideration. The active response frequencies are ranked based on the value of $E_i = \int_{L_{e,i}} U^3(s) D^2(s) (\frac{A}{D})_{C_e=0} ds$. Here, $L_{e,i}$ is the length of the excitation zone. The highest ranked response frequency, referred to as the dominating frequency, will have its whole excitation zone to excite the structure. For overlapping zones, the response frequency with the lowest ranking will loose the overlapping zone.

Space sharing means that the excitation zones, according to the ranking system, are constant in time. This is shown in Figure 3.7. Time sharing means that the same ranking system is used to define for how long time the different response frequencies are to occupy its entire excitation zone. In this case we have overlapping excitation zones, but they are not active simultaneously. This is shown in Figure 3.8

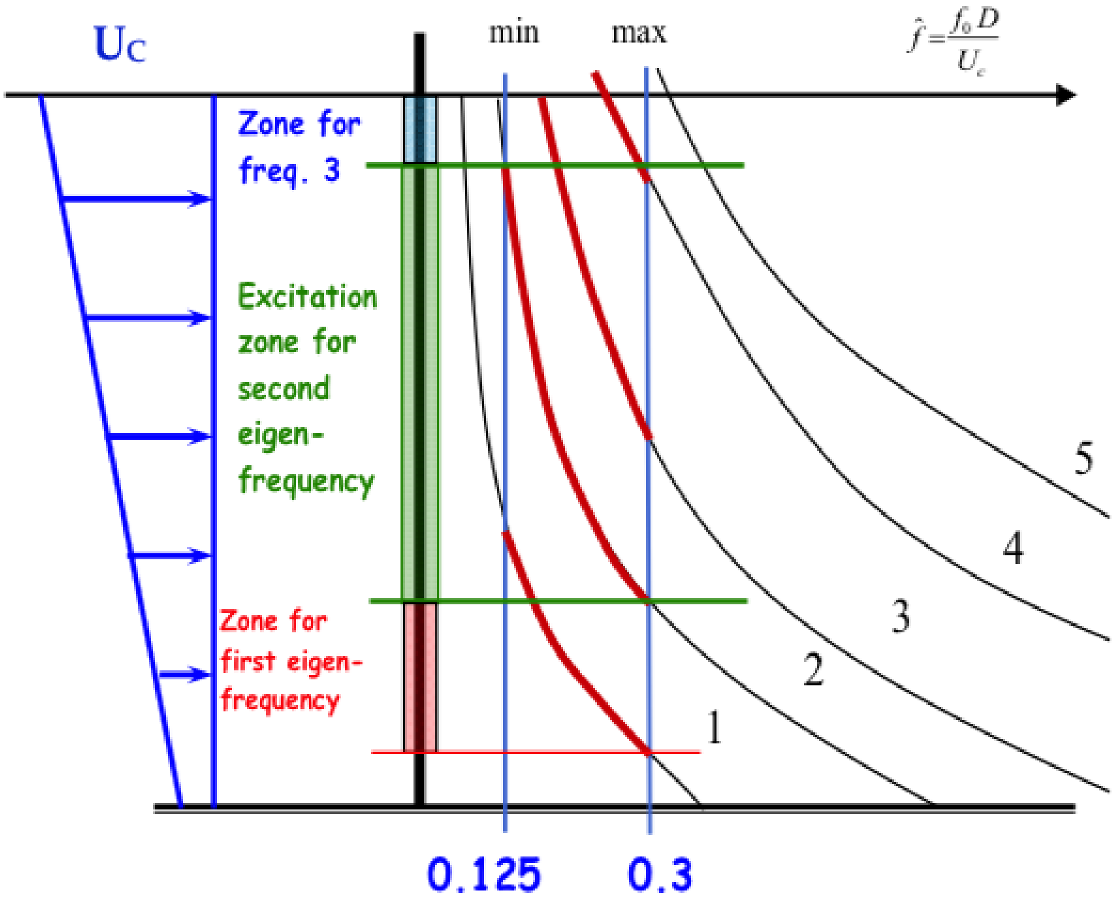

Uc
min
max
$\hat{f} = \frac{f_0 D}{U_c}$
Zone for freq. 3
Excitation zone for second eigen-frequency
Zone for first eigen-frequency
5
4
3
2
1
0.125
0.3

Figure 3.7: A riser in shear current with the second eigenfrequency as the dominating frequency (space sharing) (Larsen, 2011)

### 3.1.4 Damping model

In VIVANA, we look at structural damping and hydrodynamic damping separately.

The structural damping is limited to internal friction and local strains in the pipe, and is modeled using Rayleigh-damping. The damping matrix C is assumed to be proportional to the stiffness matrix K. The damping matrix is frequency dependent, and for response frequency $\omega_i$ the damping is written like:

$$C_i = \alpha_i K \tag{3.2}$$

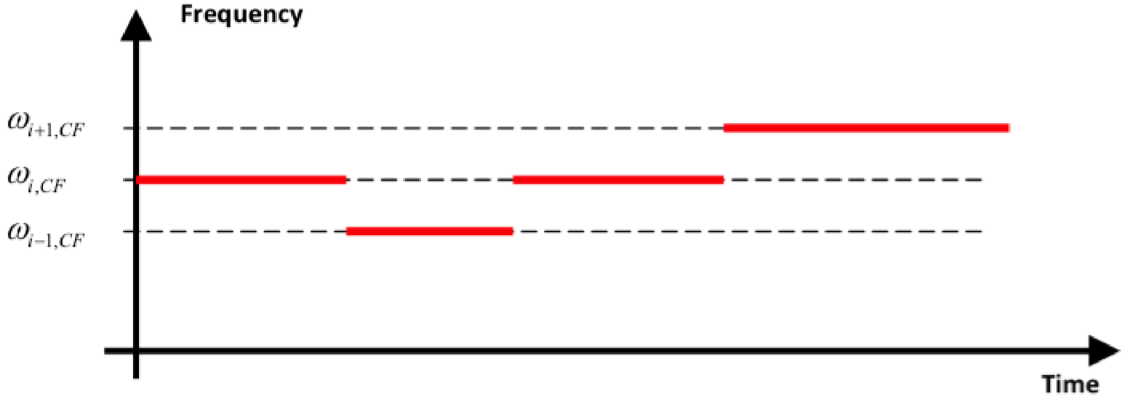


Figure 3.8: Illustration of time sharing (Passano et al., 2014)

where $\alpha_i = \frac{2\xi}{\omega_i}$. $\xi$ is the structural damping ratio.

The hydrodynamic damping is modeled at zones outside the excitation zones, and are due to drag forces. The damping force, over the length $\Delta L$, is defined as:

$$F^D = R\dot{u}\Delta L \tag{3.3}$$

Here $\dot{u}$ is the response velocity.

The damping coefficient distributed over the length of the damping zones is referred to as R. As a default value for this coefficient, VIVANA uses Venegopal's damping model propsed by Venuopal (1996). In this model, formulas for calculating R for low and high velocity zones and for still water are given in (Passano et al., 2014). The other possibility is to use the data for excitation force coefficients outside of the excitation zones, and combine this information with a general model for still water damping.

Since the excitation force coefficients turn negative for large values of the amplitude ratio, damping may also occur inside the excitation zones. VIVANA calculates the distributed damping coefficient according to the following formula:

$$R_{Ce} = -\frac{\rho D U^2 C_e}{2\omega A} \tag{3.4}$$

### 3.1.5 Solving the dynamic equilibrium equation

As already stated, VIVANA solves the dynamic equilibrium, Equation 3.5, in the frequency domain.

$$M\ddot{r} + C\dot{r} + Kr = R \tag{3.5}$$

the excitation force R is assumed to be harmonically oscillating. Introducing a complex notation we can write:

$$R = Xe^{i\omega t} \tag{3.6}$$

Here X, the load vector, contains a real part and an imaginary part. The phase angle of the harmonic load for the different sections of the structure is contained in this vector. Having harmonic loading, we assume harmonic response, hence the response is written as:

$$r = xe^{i\omega t} \tag{3.7}$$

We can split the mass matrix M and damping matrix C into a structural and a hydrodynamic component. We can also define an excitation force vector $X_L$ which is zero outside the excitation zones. The result is:

$$(M_s + M_H)\ddot{r} + (C_s + C_H)\dot{r} + Kr = X_L e^{i\omega t} \tag{3.8}$$

Substituting the expressions for r into the equilibrium equation we get:

$$-\omega^2(M_s + M_H)x + i\omega(C_s + C_H)x + Kx = X_L \tag{3.9}$$

Now, stating that the hydrodynamic damping coefficient and excitation force are functions of the response, we can write the response vector as:

$$x = [-\omega^2(M_s + M_H) + i\omega(C_s + C_H(x)) + K]^{-1} X_L(x) = H(\omega)X_L(x) \tag{3.10}$$

Here, $H(\omega)$ is referred to as the frequency response matrix.

For consistency in the solution, iteration must be applied to assure agreement between response, damping and excitation force.

# Chapter 4

# Time domain VIV models

Several time domain VIV models have been developed during the last decades. In this chapter, some of these methods are presented. Thorsen's time domain VIV model is the basis for the analysis part of this thesis (see Chapter 7 and 8). Because of this, Thorsen's VIV model is presented in great detail, in the next chapter.

## 4.1 Wake oscillators

Several attempts creating a time domain VIV model use the concept of a wake oscillator. A concept which was introduced by (Zarantonello and Brikhoff, 1957). The basic idea of the wake oscillator is that a a single variable is capable of describing the vortex shedding process. The variable satisfies a van der Pol equation describing a self-sustained, stable and nearly harmonic oscillator of finite amplitude (Facchinetti et al., 2004).

## 4.2 The MARINTEK model

(Lie, 1995) proposed a time domain VIV model which was described and compared to other VIV models by (Larsen and Halse, 1997). In the following it is refer to as the MARINTEK model. The model is applicable in cross-flow direction only, and does not take into account increased drag force due to the cross-flow vibrations. The structure is a finite element model and the dynamic equilibrium equation is solved using direct time integration.

The dynamic equilibrium is presented as:

$$(M + C_A)\ddot{x}(t) + (C + C_L)\dot{x}(t) + Kx(t) = F_v(t) \tag{4.1}$$

M, C and K are the structural mass, damping and stiffness matrix respectively. x(t) is the cross flow response vector, $\dot{x}(t)$ is the cross flow velocity vector and $\ddot{x}(t)$ is the cross flow acceleration vector. The hydrodynamic load is given by $F_v$, $C_L$ and $C_A$. $F_v$ is the Strouhal load vector, $C_L$ is the hydrodynamic damping matrix and $C_A$ is the added mass matrix.

The model is semi-empirical and based on experiments by (Sarpkaya, 1978). The experiments consist of forced harmonic motion of cylinders in steady flow. Describing the measured load as given in Equation 4.2, it is possible to extract a force coefficient in phase with acceleration ($C_{mh}$) and a coefficient in phase with velocity ($C_{dh}$). $\omega_c$ is the frequency of the forced harmonic motion.

$$F_L = \frac{1}{2}\rho DLU^2[C_{mh}sin(\omega_c t) - C_{dh}cos(\omega_c t)] \tag{4.2}$$

From Sarpkaya's data, the MARINTEK model defines the added mass coefficient and damping coefficient as follows:

$$C_A = C_{mh}\frac{\rho DLU^2}{2x_a{\omega_c}^2} \tag{4.3}$$

$$C_L = C_{dh}\frac{\rho DLU^2}{2x_a\omega_c} \tag{4.4}$$

where $x_a$ is the amplitude of the forced oscillation.

The forces caused by the vortex shedding process is found in the Strouhal load vector $F_v$ given as:

$$F_v = \frac{1}{2}\rho DLU^2 C_v sin(\omega_v t + \phi_v) \tag{4.5}$$

where $\omega_v$ is the Strouhal frequency, $\phi_v$ is the phase shift between the force and the structural response and $C_v$ is an empirical force coefficient.

To solve Equation 4.1 the response is allowed to build up with $F_v$ as the only excitation force. Then, the following procedure for time integration is performed (Larsen and Halse, 1997):

1. Based on the previous time history for each node of the structural model, frequency of oscillation an amplitude are found
2. Reduced velocity at each node is calculated from $U_R = \frac{2\pi U}{\omega D}$
3. $C_L$ and $C_A$ are found from Equation 4.3 and 4.4 for each node

4. The calculated hydrodynamic coefficients are introduced in Equation 4.1, and the dynamic equilibrium equation can be solved for this specific time step

What is positive with the MARITNEK model is that correlation length is not of concern. The response calculated from the procedure above will automatically build up correlation. Determination of correlation length has been a problem in earlier VIV models (Larsen and Halse, 1997). One example is the LICengineering model, where a model for vortex shedding correlation gives the correlation length. A more detailed description of the LICengineering model can be found in (Hansen, 1982).

A limitation regarding the MARINTEK model is due to the empirical coefficients. Since the experimental results are based on only one active frequency, the model will have trouble predicting VIV for structures with several active modes and frequencies. Also in a case with oscillating inflow, the time integration method applied will make it difficult to distinguish the convergency process and the transient dynamic effect.

## 4.3 Time Domain model by Lyle Finn, Kostas Lambrakos and Jim Maher

A time domain VIV model for prediction of riser VIV was developed and presented in (Finn et al., 1999). The goal with the method was to get a better VIV prediction tool than SHEAR7, for risers. Risers behave highly nonlinear due to moving contact point between riser and seabed, and also due to vessel motion on the sea surface. A linear frequency domain model is not capable of capturing these effects in a satisfactory way.

The time domain VIV model is three-dimensional. The force model is based on Morison's equation, including both a drag force term and an inertia term. The hydrodynamic force model is decomposed into an in-line model and a transverse (cross-flow) model.

The in-line force is given as:

$$F_{IT} = F_{DI} + F_{II} \tag{4.6}$$

The drag term $F_{DI}$ is:

$$F_{DI} = \frac{1}{2}\rho D C_D |V_{rel}| V_{IN} \tag{4.7}$$

Here, $V_{rel}$ is the resultant relative velocity vector between the inflow velocity and response velocity. $V_{IN}$ is the in-line component of the relative velocity.

The inertial term $F_{II}$ is given by:

$$F_{II} = \rho\frac{\pi D^2}{4}(C_a(a_{WI} - a_{RI}) + a_{WI}) \tag{4.8}$$

Here, $a_{WI}$ is the in-line acceleration of the water and $a_{RI}$ is the in-line acceleration of the riser.

The transverse force also consists of an inertia term and a drag term. They are given in the same way as in Equation 4.6 and 4.8, except for the in-line components of velocity and acceleration are replaced by the corresponding components in transverse direction. The difference between the in-line force and the transverse force is that the transverse force also includes the lift force. The lift force is given by:

$$F_L = \frac{1}{2}\rho C_L(t) D {V_{IN}}^2 \tag{4.9}$$

The lift force coefficient is:

$$C_L(t) = C_{L0} cos(2\pi f_L t + \phi_L) \tag{4.10}$$

Where $C_{L0}$ is the amplitude of the lift coefficient. It depends on Reynolds number and the amplitude of the VIV motions. $f_L$ is the frequency of the lift coefficient and $\phi_L$ is the phase of the lift coefficient. $f_L$ and $\phi_L$ are determined from experimental results presented by (Blevins, 1990).

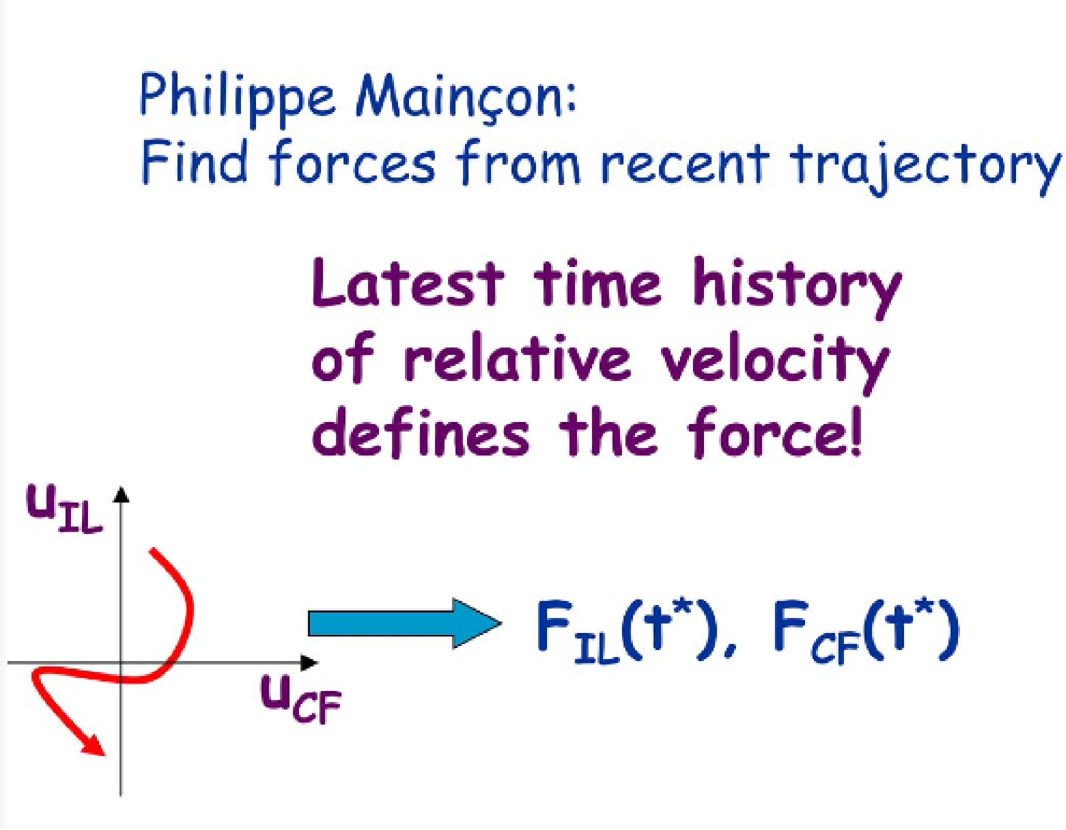
The presented force model is coupled to the FEM software ABAQUS. It makes it possible to account for geometric nonlinearities like large displacements and seafloor contact.

## 4.4 Time domain model by Philippe Mainçon

(Mainçon and Larsen, 2011) proposed a time domain VIV model. The work was based on the postulate:

> The fluid force on a section of the slender structure, is completely determined by the recent histories at that section of the velocities of the structure and the undisturbed fluid (Mainçon and Larsen, 2011)

By this formulation essential assumptions of the VIV model are included. Looking at the fluid forces on "a section" of the slender structure indicates that strip theory is assumed. This means that in the fluid, the fluid forces on one section are not influenced by the fluid force on another section. Furthermore, that the forces are "completely determined" means that the model is deterministic. The most important point in the postulate is related to the velocities of the fluid and the structure. In (Mainçon and Larsen, 2011) the word "tachogram" is used as the "recent history of x and y components of the velocity". This velocity history has to be determined in order to find the fluid forces on the section, and how this is done is a key point of this method. An illustration of the postulate is given in Figure 4.1.



Philippe Mainçon:
Find forces from recent trajectory

Latest time history
of relative velocity
defines the force!

Figure 4.1: Illustration showing that the velocity trajectory in in-line and cross-flow for a section gives the fluid forces in in-line and cross-flow direction (Larsen)

The VIV model can be split up into the following steps (Mainçon and Larsen, 2011):

- The velocity of the cylinder relative to the fluid is found
- the velocity is scaled making the cylinder diameter the unit of distance
- The tachogram is compressed into a small number of "Laguerre coefficients"
- The Laguerre coefficients are used to enter an interpolation function returning the x and y component of the hydrodynamic force
- The force is scaled back to the relevant diameter
- Forces related to acceleration of the fluid flow is added

A problem with this work is to determine what is recent time history, to be able to find the forces.

The model also needs data for a large number of trajectories. From case studies performed by (Mainçon and Larsen, 2011) the conclusion is that trajectory stability and related issues has to be understood in greater detail to make this model an engineering tool.

# Chapter 5

# Thorsen's time domain VIV model

Since 2012 Mats Jørgen Thorsen has worked on a description of the hydrodynamic forces caused by VIV. Finding a hydrodynamic force formulation applicable in time domain, Thorsen is developing a calculation program, making it possible to predict VIV response, for slender marine structures, by solving the dynamic equilibrium equation in time domain. The following discussion is based on (Thorsen et al., 2014), (Thorsen et al., 2015a) and (Thorsen et al., 2015b).

The structure is divided into a set of elements, and the program calculates the hydrodynamic forces at each element by assuming no hydrodynamic interaction between them. The structure is a FE model with structural stiffness, damping and mass. The program calculates CF VIV and IL VIV separately due to the assumption of small displacements and linear structural behavior. The coordinate system is defined as shown in Figure 5.1. The x-axis represents the IL-displacement away from the static equilibrium position, while the y-axis represents the cross-flow displacement. Furthermore x and y are used as subscripts indicating in-line and cross-flow respectively.

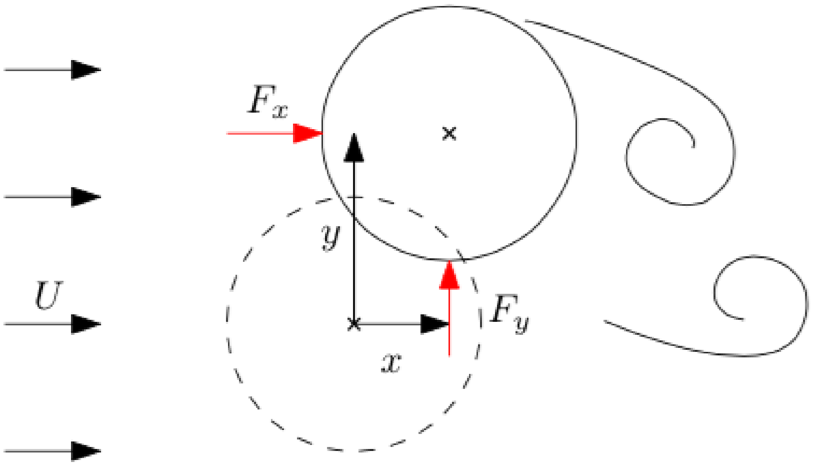


Figure 5.1: The coordinate system used in NTNU time domain (Thorsen et al., 2015b)

## 5.1 Hydrodynamic force model in cross-flow direction

The hydrodynamic force model consists of three terms:

- The excitation force $F_{exc,y}$
- The damping force $F_{d,y}$
- The added mass force $F_{a,y}$

The excitation force is due to vortex shedding changing the pressure field around the cylinder, causing a force resultant in the y-direction. It is given as:

$$F_{exc,y} = \frac{1}{2}\rho D U^2 C_v cos(\phi_{exc,y}) \tag{5.1}$$

A main difference between this program and other VIV models is the synchronization between the oscillation frequency of the cylinder and the vortex shedding frequency. Inspired by biological synchronization mechanisms (Thorsen et al., 2014), Thorsen was able to mathematically describe a relation between the frequency of oscillation and the vortex shedding frequency. This property is expressed in $\phi_{exc,y}$ referred to as the instantaneous phase of the force.

$$\frac{d\phi_{exc,y}}{dt} = H(\phi_{\dot{y}} - \phi_{exc,y}) \tag{5.2}$$

$\phi_{\dot{y}}$ is the instantaneous phase of the cross-flow velocity and H is a function of $\phi_{\dot{y}} - \phi_{exc,y}$, determined from modified data of the excitation coefficient. The meaning of Equation 5.2 can be explained by looking at two different signals oscillating from 0 to $2\pi$ with a phase shift between them. This is illustrated in Figure 5.2. Equivalently, we can look at the oscillations as time dependent angles around the unit circle. We now let one signal be the instantaneous phase of the force, and the other signal is the instantaneous phase of the cross-flow velocity. In order for the two signals to synchronize to each other, assuming the phase of the cross-flow velocity constant and larger than the phase of the force, $\phi_{exc,y}$ has to increase for the two phases to be equal. Mathematically speaking $\frac{d\phi_{exc,y}}{dt} > 0$ when $\phi_{\dot{y}}$ is constant. This is the situation in Figure 5.3.

The function H gives the synchronization properties between the excitation force and the oscillation frequency of the cylinder. It is made non-dimensional so that $\hat{f} = \frac{fD}{U} = \frac{HD}{2\pi U}$, where f is the oscillation frequency of the cylinder. As seen from Figure 5.4, $\hat{f} \in (0.1, 0.25)$. Thus $\hat{f}$ has to be in this range for lock-in to occur. If not, the excitation force will not synchronize to the cylinder velocity, and the fluid will be unable to transfer energy to the structure. Hence the vibration amplitudes will be severely reduced.

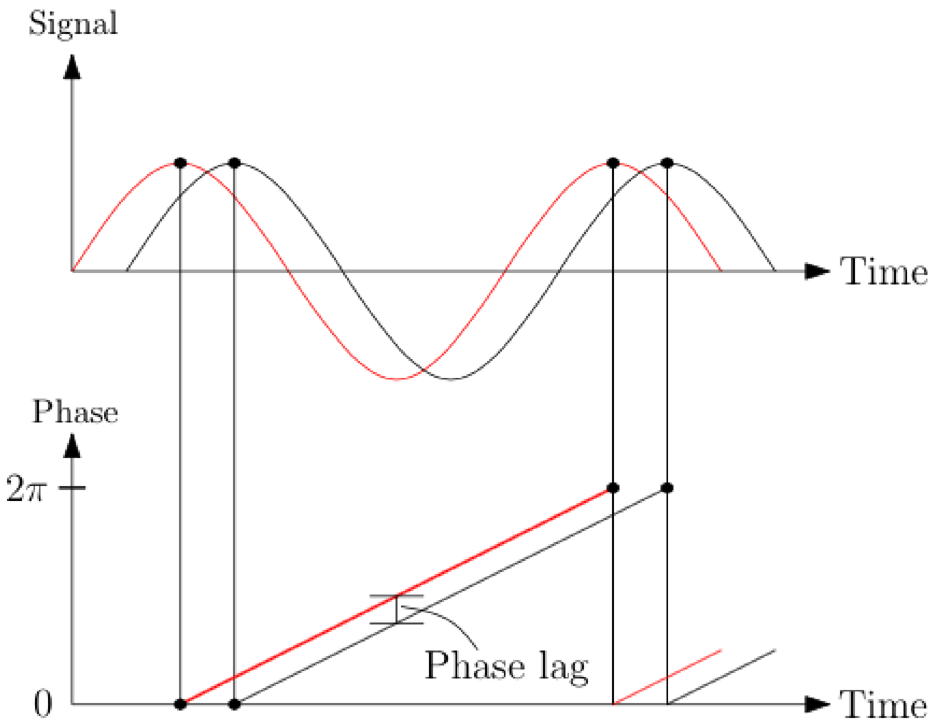


Figure 5.2: Two oscillating signals with a phase lag between them (Thorsen et al., 2015b)

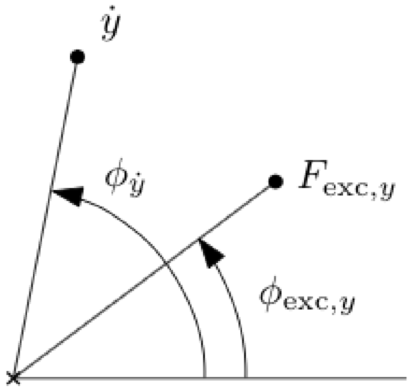


Figure 5.3: The instantaneous phase of cross-flow velocity and the instantaneous phase of the excitation force (Thorsen et al., 2015b)

The excitation force is a function of the cross-flow motion amplitude referred to as $y_0$. This dependency is expressed in the non-dimensional coefficient $C_v = C_v(\frac{y_0}{D})$, and is based on experimental results. The relation is shown in Figure 5.5

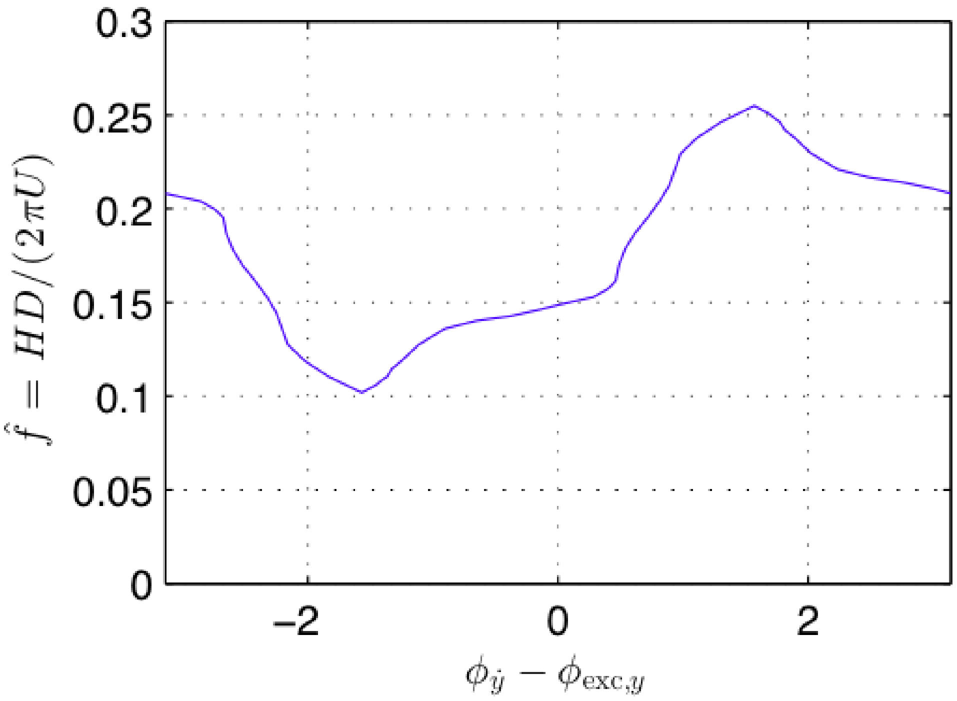


Figure 5.4: Non-dimensional frequency plotted against the instantaneous phase of the cross-flow velocity minus the instantaneous phase of the force (Thorsen et al., 2015b)

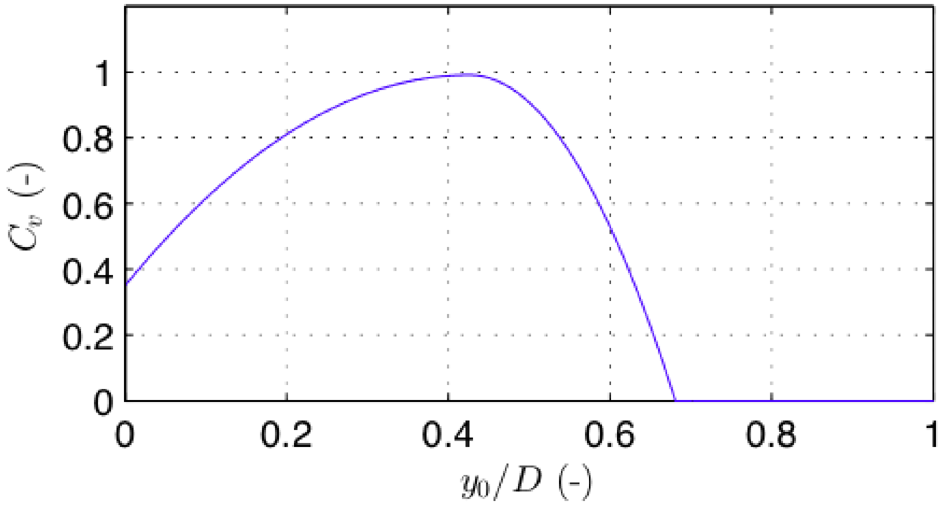


Figure 5.5: $C_v$ as a function of the amplitude ratio (Thorsen et al., 2015b)

The damping force is due to resistance created by cross-flow cylinder motion (Thorsen et al., 2015b). It is given by the following formula:

$$F_{d,y} = -\frac{1}{2}\rho D C_{d,y}|\dot{y}|\dot{y} \tag{5.3}$$

Where $C_{d,y} = 0.31 + 0.89\frac{y_0}{D}$. The damping force model was originally based on the damping model proposed by (Venugopal, 1996). However, for the damping model to work in time domain, modifications had to be done to make the damping not a function of frequency. In the latest version of the damping force model, as presented here, there is good agreement with experimental results from (Vikestad et al., 2000) and (Morse and Williamson, 2009).

The added mass force is the force caused by the changing pressure field due to the cylinder acceleration. Based on potential theory we get:

$$F_{a,y} = -\rho\frac{\pi D^2}{4}\ddot{y} \tag{5.4}$$

Thus the total hydrodynamic force in cross-flow direction is expressed as:

$$F_y = \frac{1}{2}\rho D U^2 C_v cos(\phi_{exc,y}) - \frac{1}{2}\rho D C_{d,y}|\dot{y}|\dot{y} - \rho\frac{\pi D^2}{4}\ddot{y} \tag{5.5}$$

## 5.2 Hydrodynamic force model in in-line direction

Concerning in-line VIV, the hydrodynamic force is based on the same principles as for the cross-flow case. The force is split up into an excitation term, a damping term and an added mass term. However, there are some differences. The motion amplitudes for in-line VIV are strongly dependent on cross-flow motion (Larsen, 2011). This is shown in Figure 5.6. It is observed that for $U_r > 2.5$ both IL and CF VIV will occur, and as a result of increasing cross-flow motion amplitudes, the in-line amplitudes will also increase. Thus, it is important that the present VIV model is able to capture this physical effect. Thorsen's VIV model is not capable of predicting IL VIV for $U_r$ less than 2.5. This is due to the excitation force model (Equation 5.6), which depends on the maximum cross-flow amplitude. At reduced velocity less than 2.5, CF VIV is not present.

The excitation force in in-line direction is derived assuming that the relative inflow velocity on the structure is normal to the excitation force vector. This is illustrated in Figure 5.7. Physically, we can argue that for a stationary cylinder, the in-line force component of the excitation force is much smaller than the cross-flow component. Thus the undisturbed inflow velocity is approximately normal to the excitation force vector. For an oscillating cylinder, the same argument is applied, but instead of the undisturbed inflow velocity, the relative velocity is assumed normal to the excitation force (Thorsen et al., 2015a).

Based on this simplification the following formula for the oscillating excitation force in in-line

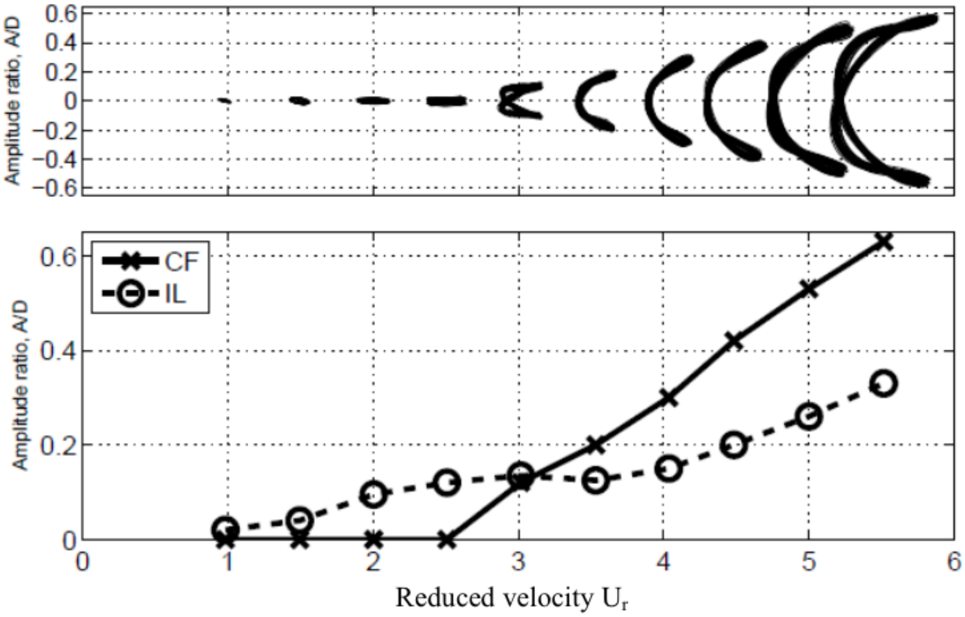


Figure 5.6: Trajectory and amplitude of vibrations for the midpoint of a flexible beam for increasing flow velocity (Larsen, 2011). Originally from (Aronsen, 2006)

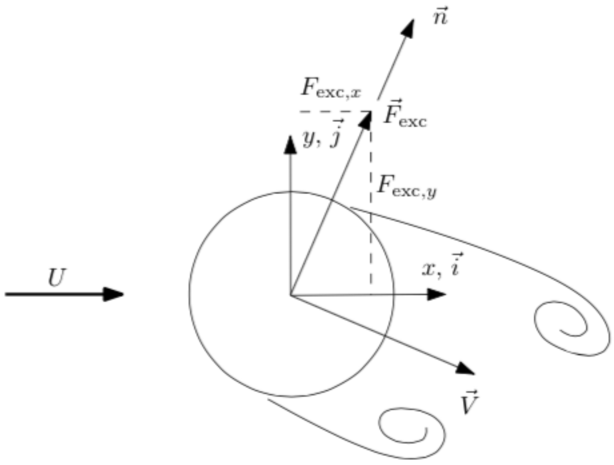


Figure 5.7: Cylinder with undisturbed inflow velocity U, relative velocity $\vec{V}$ and excitation force vector $\vec{F}_{exc}$ (Thorsen et al., 2015a)

direction can be derived:

$$F_{exc,x} = \frac{1}{4}\rho D U \dot{y}_{max} C_v cos(\phi_{exc,x}) \tag{5.6}$$

Where $\dot{y}_{max}$ is the maximum CF velocity, $C_v$ is the same amplitude dependent function as for the CF case, and $\phi_{exc,x}$ is the instantaneous phase of the IL velocity.

(Pettersen, 2007) states that the IL excitation force oscillates with two times the vortex shedding frequency, while the CF excitation force oscillates with the vortex shedding frequency. Using the notation formulated by Thorsen this implies that $\frac{d\phi_{exc,x}}{dt} = 2\frac{d\phi_{exc,x}}{dt}$. However, for lock-in to occur, the in-line excitation force need to be able to synchronize with the in-line velocity of the cylinder. This is obtained assuming:

$$\frac{d\phi_{exc,x}}{dt} = 2\frac{d\phi_{exc,x}}{dt}[1 + \alpha sin(\phi_{\dot{x}} - \phi_{exc,x})] \tag{5.7}$$

Where $\alpha$ is a small number so that $\frac{d\phi_{exc,x}}{dt} \approx 2\frac{d\phi_{exc,x}}{dt}$.

The total hydrodynamic force in in-line direction is given as:

$$F_x = \frac{1}{4}\rho DU\dot{y}_{max}C_v cos\phi_{exc,x} - \frac{1}{2}\rho DC_{d,x}|\dot{x}|\dot{x} - \rho\frac{\pi D^2}{4}\ddot{y} \tag{5.8}$$

Where $C_{d,x} = 0.31 + 0.89\frac{x_0}{D}$

## 5.3 Phases and amplitudes

To implement an excitation force depending on response, we need a way of extracting this information in the dynamic analysis. Assuming a narrow-banded signal the response amplitude can be estimated from:

$$A_y = \frac{1}{2}\int_{t_1}^{t_2} |\dot{y}| dt \tag{5.9}$$

Here $t_1$ and $t_2$ are the time at the most recent zero up crossings of the velocity. The synchronization model needs the phase of the velocity. The concept of phase portraits is used to find the phase. More information about the phase portrait is provided in (Thorsen et al., 2015a).

## 5.4 Structural model

The structure itself consists of 2D beam elements in a finite element model. Axial degrees of freedom are not taken into account, so that each element has two rotation dofs and two translation dofs. The dynamic equilibrium equation is given as:

$$\mathbf{M}\ddot{\mathbf{r}}(t) + \mathbf{C}\dot{\mathbf{r}}(t) + \mathbf{K}\mathbf{r}(t) = \mathbf{F}(t) \tag{5.10}$$

The mass matrix **M** is based on a consistent mass formulation and the added mass term from the hydrodynamic force is included. The damping matrix **C** provides the structural damping, and a Rayleigh-damping formulation is applied so that $\mathbf{C} = \alpha_1\mathbf{M} + \alpha_2\mathbf{K}$. **K** is the stiffness matrix and is a sum of the bending stiffness and the geometric stiffness due to tensioning. $\mathbf{r}(t)$ contains the nodal displacements, and the hydrodynamic force vector $\mathbf{F}(t)$ contains either in-line or cross-flow hydrodynamic forces. Equation 5.10 is solved using time integration. Newmark-$\beta$ is applied with $\lambda = 0.5$ and $\beta = 0.25$ (see Section 7.1).

# Chapter 6

# Pipelines

Pipelines are frequently used by the offshore industry for transportation of oil and gas to land terminals (Larsen et al., 2004). When laid on an uneven seabed, these pipelines will have several free spans, as is the case for the Ormen Lange field in the North Sea. When facing a current, the free spans will be subjected to an oscillating excitation force, due to the vortex shedding process. If the structural characteristics of the free spanning pipe, the seabed profile and the current condition is such that the frequency of the excitation force is close to an eigenfrequency of free spanning pipe, we get VIV. The resulting stresses can cause fatigue damage. It is thus important to make fairly accurate predictions of VIV, to make sure the estimated lifetime of the pipeline is acceptable.

A good prediction tool for VIV on free spanning pipelines requires a model taking into account all important nonlinear effects. The pipe-seabed interaction requires nonlinear seabed springs. They should allow the pipe to lift off from the soil, but to some extent prevent pipe penetration of the seabed. Also nonlinear soil damping can be of importance for the pipeline response. Boundary conditions will be time varying because the touch down point position is a function of pipeline response, soil erosion and tension variations. This will again affect the length of the free span.

There are still many uncertainties related to the hydrodynamics for a pipe close to a seabed. Empirical VIV models are typically based on coefficients found from tests with beams in infinite fluid. The hydrodynamic situation will be different when the cylinder is close to a wall. More knowledge about soil pipe-interaction is a key point for better prediction tools of VIV for free spanning pipelines. Data on cylinder vibrations close to a wall exists, but has traditionally not been implemented in VIV codes.

## 6.1 Pipe-soil interaction

Pipe-soil interaction is complex. It depends strongly on the soil properties, which are not easily obtained. In addition, parameters like loading history, load rate and amplitude are important for the interaction (Veritas, 2006). In the following, general recommendations for how to determine soil stiffness and soil damping levels based on (Veritas, 2006), are briefly discussed. Also, a simplified method for determination of the soil parameters is presented. Then, methods on how to account for nonlinear soil-pipe interaction in a nonlinear FE model, are given. We limit ourselves to look at the vertical soil-pipe interaction properties. This means horizontal interaction, such as friction, is not discussed.

### 6.1.1 Stiffness due to soil-pipe interaction

The soil stiffness depends on the material of the soil. In (Veritas, 2006) the soil is classified into cohesive soil (clays) and cohesionless soil (sand). The different soil materials have different properties, and the Recommended Practice suggests what properties to use. Some of the properties are: Submerged weight, Poisson ratio, shear strength and plasticity index. Based on the soil properties, formulas for vertical reaction forces per unit length are proposed. $R_v$ is used as the symbol for reaction force per unit length, and $v$ is the symbol for penetration.

(Veritas, 2006) differ between the static soil stiffness and the dynamic soil stiffness. The static soil stiffness is determined as $K_{v,s} = R_v / v$ and the dynamic stiffness as $K_v = \Delta F_v / \Delta\delta_v$. Here, $\Delta F_v$ is the vertical dynamic force between pipe and soil per unit length, and $\Delta\delta_v$ is the associated vertical displacement of the pipe.

The vertical soil stiffness may be evaluated as:

$$K_V = \frac{0.88G}{1-\nu} \tag{6.1}$$

"This equation is based on elastic half space theory for a rectangular foundation under assumption of a pipe length that equals 10 times the contact width between pipe and soil" (Veritas, 2006). Here, G is the shear modulus and $\nu$ is the Poisson's ratio.

Another option, when the seabed shape is not complex, is to express $K_v$ $[kN/m^2]$ as:

$$K_v = \frac{C_v}{1-\nu}(\frac{2}{3}\frac{\rho_s}{\rho} + \frac{1}{3})\sqrt{D} \tag{6.2}$$

where $\frac{\rho_s}{\rho}$ is "specific mass ratio between the pipe mass (not including added mass) and the displaced water" (Veritas, 2006). The corresponding values of $C_v$ and the static soil stiffness $K_{v,s}$ is given in the tables below.

Table 6.1: Stiffness properties for pipe-soil interaction in sand (Veritas, 2006)

| **Sand type** | $C_v$ $[kN/m^{5/2}]$ | $K_{v,s}$ $[kN/m^2]$ |
|---|---|---|
| Loose | 10500 | 250 |
| Medium | 14500 | 530 |
| Dense | 21000 | 1350 |

Table 6.2: Stiffness properties for pipe-soil interaction in clay (Veritas, 2006)

| **Clay type** | $C_v$ $[kN/m^{5/2}]$ | $K_{v,s}$ $[kN/m^2]$ |
|---|---|---|
| Very soft | 600 | 50-100 |
| Soft | 1400 | 160-260 |
| Firm | 3000 | 500-800 |
| Stiff | 4500 | 1000-1600 |
| Very stiff | 11000 | 2000-3000 |
| Hard | 12000 | 2600-4200 |

### 6.1.2 Damping due to soil-pipe interaction

For a pipeline with bottom contact, we will have a damping contribution from the soil-pipe interaction. It can be split into two mechanisms:

- Material damping associated with the pipe-soil interaction
- Radiation damping due to elastic waves

The radiation damping is strongly dependent on oscillation frequency, and is more important for high frequency oscillations. It is mainly the material damping which is important for free spanning pipelines. This damping mechanism is associated with the material properties of the pipe an the sea bottom. The soil typically consists of sand or rocks and can have a large variety of hardnesses. In general, softer soil will give a larger soil damping. However, there are large uncertainties related to the material properties of the soil, which must be taken into consideration when choosing a damping level in the VIV analysis. In (Veritas, 2006) recommended values for the modal soil damping ratio is given for different soil types and hardnesses.

A modal analysis for determination of soil damping values is proposed. The modal soil damping ratio is defined as follows:

$$\xi_{soil} = \frac{1}{4\pi f_0} \frac{\int_L c(s)\phi^2(s)ds}{\int_L m(s)\phi^2(s)ds} \tag{6.3}$$

where c(s) is the distributed soil damping. It depends on elastic energy stored by the soil and the energy dissipated by the viscous damper.

In a FEM analysis, c(s) must be modeled with discrete dampers. In this case the viscous damping coefficient $c_i$ for support i is found from:

$$c_i = 2\xi_i \frac{k_i}{\omega} \tag{6.4}$$

$k_i$ is linearized spring stiffness at support i, $\xi_i$ is the damping ratio at support i and $\omega$ is the angular frequency of the mode. The damping ratio can be found from the following formula:

$$\xi_i = \frac{1}{4\pi} \frac{E_{Dissipated}}{E_{Elastic}} \tag{6.5}$$

$E_{Dissipated}$ is the energy loss per cycle at support i, and $E_{Elastic}$ is the equivalent elastic energy at support i. This is illustrated in Figure 6.1.

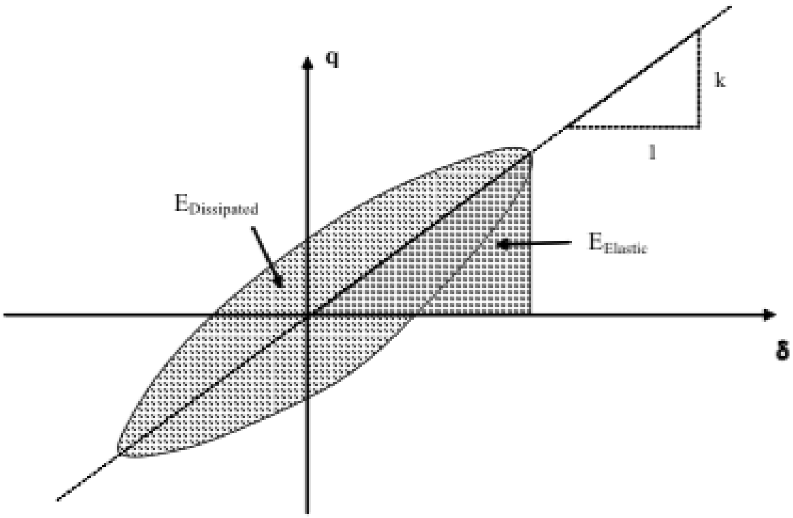


Figure 6.1: Energy dissipation at soil support (Veritas, 2006)

The soil damping depends on support displacements and is nonlinear. Iteration must be performed to assure agreement between $k_i$ and $\xi_i$, for determination of $c_i$ in Equation 6.4. In Figure 6.2, we see how the soil stiffness and the damping ratio depend on dynamic displacement and soil penetration.

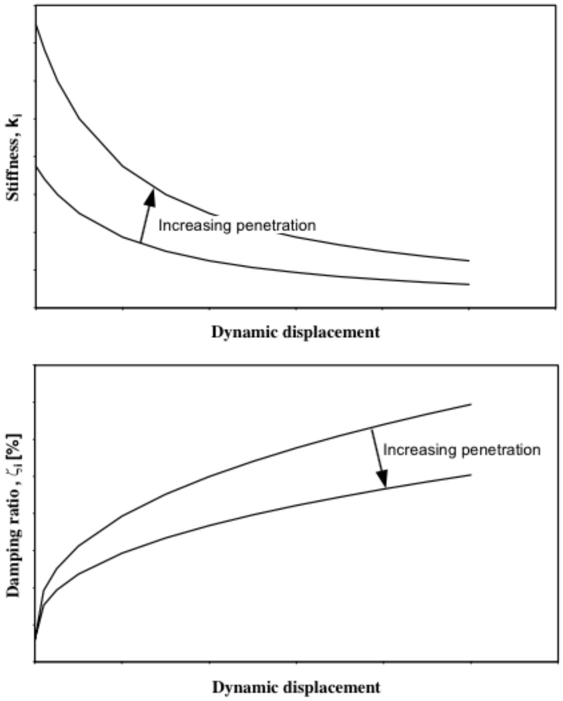


Figure 6.2: Characteristics of a nonlinear soil damper (Veritas, 2006)

In addition to the material damping and radiation damping, hydrodynamic damping close to the soil can be a contribution to the total soil-pipe interaction damping. When the pipe moves up and down relative to the soil, the induced fluid motion will generate damping.

### 6.1.3 Simplified method to determine soil stiffness and soil damping

As an estimate of the soil stiffness and the soil damping values, the simplified procedure applied in the Project Thesis (Ulveseter, 2014), can be used.

The critical damping per meter is given as:

$$c_{cr} = 2(m + m_a)\omega_0 \tag{6.6}$$

where m is the dry mass per meter and $m_a$ is the added mass per meter, $\omega_0$ is the eigenfrequency given as $\omega_0 = \sqrt{\frac{k}{m+m_a}}$, and k is the soil stiffness per meter. Thus,

$$c_{cr} = 2\sqrt{k(m + m_a)} \tag{6.7}$$

We assume that where we have pipe-soil interaction, the penetration of the seabottom is 1/4 of the pipe radius. Statically, the force on the seafloor from the pipe is due to the submerged weight of the pipe (see Figure 6.3). From this, assuming a linear spring stiffness we can make use of Hooke's Law, and hence find the spring stiffness.

$$k_s = \frac{F}{\Delta x} \tag{6.8}$$

where the force $F = (m - \rho A)g$ and $\Delta x = \frac{1}{4}R$. R is the radius of the pipe, A is the cross section area, $\rho$ is the density of water and g is the gravitational acceleration.

The relative damping ratio is given as $\lambda = \frac{c_s}{c_{cr}}$. This implies that the vertical soil damping as function of the relative damping ratio can be found as:

$$c_s = \lambda 2\sqrt{k(m + m_a)} \tag{6.9}$$

Using these assumptions, the soil stiffness is given directly from Equation 6.8. The vertical soil damping can be determined from an estimation of the damping ratio according to Equation 6.9. This procedure is used in the case studies in Chapter 8, where the damping ratio $\lambda$ is used to indicate the magnitude of the soil damping.

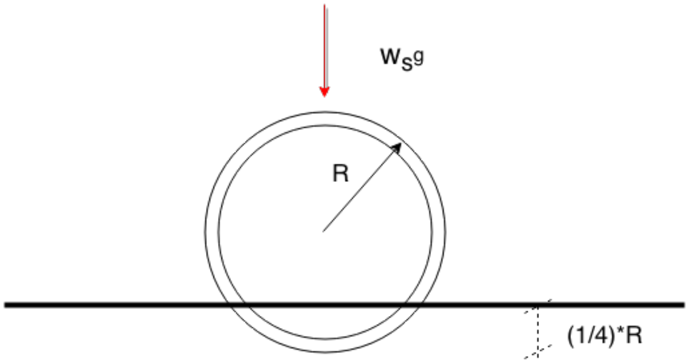


Figure 6.3: Pipeline penetrates the seafloor due to the submerged weight of the pipeline

### 6.1.4 Soil-pipe interaction implemented in a FE model

In a FE model, soil-pipe interaction must be included using discrete damper and stiffeners at vertical translations dofs where the pipe penetrates the seafloor. This is discussed in (Lie et al., 2001) and (Larsen and Passano, 2006), where VIV calculations on catenary risers are the topic. The same theory is applicable for free spanning pipelines.

There are three models that can be used for analyses of seafloor-pipe interaction. In Figure 6.4, the principles are illustrated.

- **Truncated model**: The free span is simply supported at both ends. All energy is reflected back from the support points, and the seabed is not taken into account. This simple model is often used in VIV analyses.

- **Elastic spring support**: The seabed at the pipe shoulders is modeled with soil springs. Thus the energy is not reflected from a single point, but from all the elastic springs. This model is more realistic and accurate than the truncated model, concerning local stress distribution around the touch down point.

- **Springs and dampers**: In addition to the elastic springs, discrete dampers at the soil contact points will transfer some of the energy to the fluid. This is the most realistic model due to the soil-pipe damping as described in Section 6.1.2.

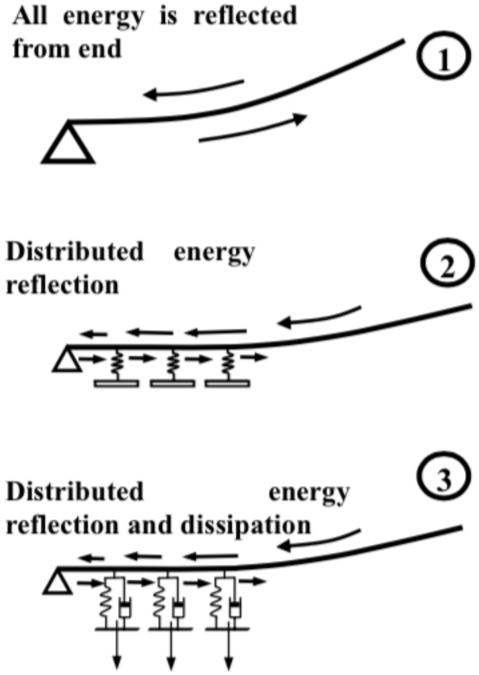

All energy is reflected from end
1
Distributed energy reflection
2
Distributed energy reflection and dissipation
3

Figure 6.4: Modeling of soil-pipe interaction in a FE-analysis (Larsen and Passano, 2006)

The most accurate FE model, will be the third item in the list above, using nonlinear springs and dampers. The nonlinearity is due to the varying soil contact. The springs and dampers should be turned off when the pipeline rises from the seafloor, and should be turned on again in case of seafloor penetration. Linear springs and dampers do not allow this. The touch down point will, in the linear case, be fixed, at the static value. This makes it impossible to include new springs if the pipeline penetrates the soil at new positions. Also, springs are not removed in case of spring tension. The difference between linear and nonlinear springs is shown in Figure 6.5. The same concept applies for soil dampers.

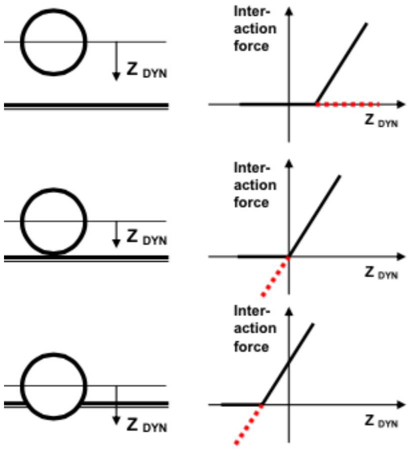


Figure 6.5: Linear soil springs (red dashed line) and nonlinear soil springs (black line) (Larsen and Passano, 2006)

## 6.2 Modeling of VIV for pipelines

To include nonlinearities in the VIV model related to nonlinear springs at the shoulders, we must perform a nonlinear time domain analysis. Such a nonlinear time domain method is presented in (Larsen and Koushan, 2005) and (Larsen et al., 2004), where it is applied to free spanning pipelines. From Section 3.1 we know that VIVANA solves the dynamic equilibrium equation in frequency domain. A procedure for nonlinear time domain analysis using VIVANA together with RIFLEX is still possible. The procedure is as follows:

- Nonlinear static analysis in RIFLEX
- Frequency domain analysis in VIVANA
- Use the result (nodal forces, dominating frequencies, added mass and damping) from VIVANA and transfer back to RIFLEX.
- Perform a nonlinear time domain analysis in RIFLEX

The first two steps in this method are the traditional steps of a VIVANA analysis. What we achieve with the addition of the two last steps is to take the nonlinearities into consideration. This will, in theory, give a better prediction of the local stress distribution around the pipe shoulders, since we here will have nonlinear behavior. Using the procedure in the list above it is important to make sure that the response pattern for the VIVANA frequency domain solution and the time domain solution is approximately the same. If not, the result from the frequency analysis is not applicable in time domain. This is because the hydrodynamic coefficients depend on response.

CFD is briefly discussed as a tool for VIV prediction in Section 2.2, but it is concluded that the computational cost is too high. However, attempts have been made to predict VIV for pipelines splitting up the pipe in 2-dimensional sections. This was done by (Halse, 1997), and the concept is illustrated in Figure 6.6. The hydrodynamic forces are calculated for predefined sections solving Navier-Stokes equations. The force is assumed constant between the sections. This force model was coupled to a linear structural model by Halse, to illustrate a free spanning pipeline.

The force model is a function of time, so that the structural model can be coupled to a nonlinear structural model taking important nonlinearities into consideration. Unfortunately, the force model fails at high Reynolds numbers, due to the highly 3-dimensional flow picture around a pipe.

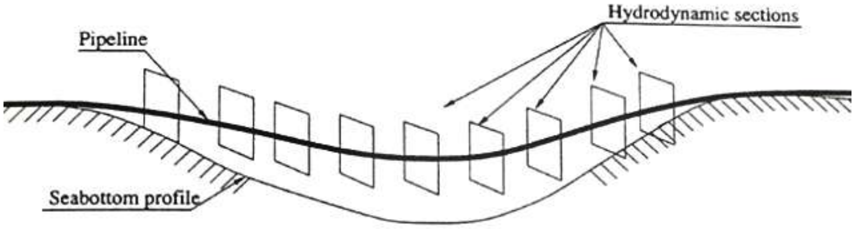


Figure 6.6: Illustration of 2-dimensional sections where CFD is applied to find hydrodynamic forces on a free spanning pipeline (Halse, 1997)

# Chapter 7

# Matlab code for VIV prediction of free spanning pipelines

In Section 5, a time domain VIV model developed by Mats Thorsen was presented. As discussed in Section 2.2 a time domain solution gives rise to more possibilities than a frequency domain model. The possibilities related to nonlinear soil dampers and nonlinear soil springs are tested and the resulting program is presented in the following. We will see how Thorsen's VIV model can be used as the basis for a simplified nonlinear analysis program of free spanning pipelines, referred to as Ulveseter's model.

## 7.1 Nonlinear time domain analysis

Ulveseter's model is based on a general procedure for nonlinear dynamic analysis. The theory is based on (Langen and Sigbjörnsson, 1979). It is assumed that the most important nonlinearity for a free spanning pipeline is due to soil-pipe interaction, so geometric nonlinearities due to large displacements and nonlinear material properties are not discussed, and not taken into account in Ulveseter's model.

Our goal is to solve the dynamic equilibrium equation given as:

$$M\ddot{r}(t) + F^D(t) + F^S(t) = Q(t, r, \dot{r}) \tag{7.1}$$

Here $F^D(t)$ is the nonlinear damping force (due to varying contact between soil and pipe) and $F^S(t)$ is the nonlinear restoring force (also due to varying contact between soil and pipe).

Equation 7.1 is solved using time integration. Subscripts are introduced indicating equilibrium at a given time step. Subtracting the equilibrium equation at time step k from equilibrium at time step k+1, we get the following incremental formulation:

$$M(\ddot{r}_{k+1} - \ddot{r}_k) + ({F^D}_{k+1} - {F^D}_k) + ({F^S}_{k+1} - {F^S}_k) = Q_{k+1} - Q_k \tag{7.2}$$

We write this equation as:

$$\Delta {F^I}_k + \Delta {F^D}_k + \Delta {F^S}_k = \Delta Q_k \tag{7.3}$$

Where the following equations are valid:

$$\Delta r_k = r_{k+1} - r_k \tag{7.4}$$

$$\Delta {F^I}_k = M \Delta \ddot{r}_k \tag{7.5}$$

$$\Delta {F^D}_k = {F^D}_{k+1} - {F^D}_k = C_{Ik} \Delta \dot{r}_k \tag{7.6}$$

$$\Delta {F^S}_k = {F^S}_{k+1} - {F^S}_k = K_{Ik} \Delta r_k \tag{7.7}$$

$$\Delta Q_k = Q_{k+1} - Q_k \tag{7.8}$$

$K_{Ik}$ and $C_{Ik}$ are the incremental stiffness matrix and damping matrix between time step k and k+1 respectively. In the case of a pipeline these matrices will depend on the soil contact. If, at a node around the shoulders, the pipe penetrates the seabed, a soil damping term and soil stiffness term is added to corresponding positions in the incremental matrices.

In theory we do a linearization of the stiffness and damping matrix within the time increment. The linearization depends on results from both time step k and time step k+1. However, since the result at time step k+1 is unknown, the initial incremental value is used.

The governing equation can now be written as:

$$M \Delta \ddot{r}_k + C_{Ik} \Delta \dot{r}_k + K_{Ik} \Delta r_k = \Delta Q_k \tag{7.9}$$

Equation 7.9 can be solved for $\Delta r_k$, $\Delta \dot{r}_k$ and $\Delta \ddot{r}_k$ using nonlinear Newmark-$\beta$ time integration. The problem is that we have no guarantee that there is equilibrium between internal and external forces, because the linearization of $K_{Ik}$ and $C_{Ik}$ only depends on the initial time step $t_k$. To solve this challenge iteration has to be introduced, or the time step has to be so small that convergence still is achieved. As discussed in Section 7.2, Ulveseter's model does not use iteration.

However, iteration schemes where tested in the code, but was later rejected due to satisfactory results, without iteration. General iteration procedures, which were considered in Ulveseter's model, are based on finding the load residual.

The load residual (error) can be expressed as:

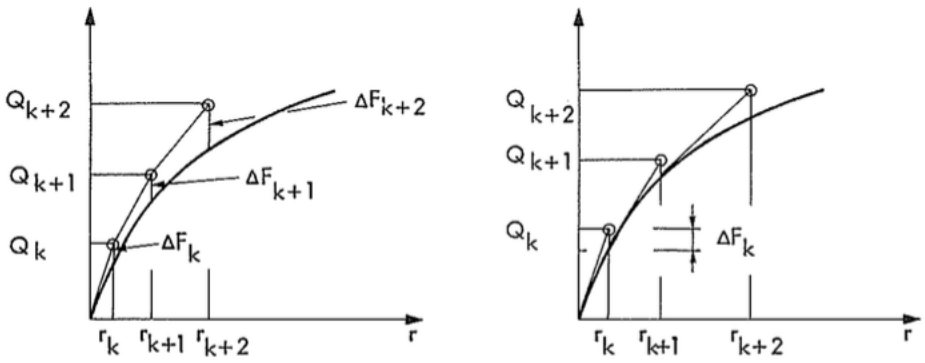
$$\Delta F_{k+1} = Q_{k+1} - ({F^I}_{k+1} + {F^D}_{k+1} + {F^S}_{k+1}) \tag{7.10}$$

To iterate on the error we substitute $\Delta Q_k$ in Equation 7.9 with $\Delta F_{k+1}$ in Equation 7.10 and compute a correction $\Delta_k$ to $\Delta r_k$. In Figure 7.1 we see the difference between iteration and no iteration. The algorithm for iteration, using i as the iteration number, is:

$$M\ddot{\Delta}_k^i + C_{Ik}\dot{\Delta}_k^i + K_{Ik}\Delta_k^i = Q_{k+1} - ({F^I}_{k+1}{}^{i-1} + {F^D}_{k+1}{}^{i-1} + {F^S}_{k+1}{}^{i-1}) \tag{7.11}$$

The new displacement increment is now:

$$\Delta r_k^i = \Delta r_k^{i+1} + \Delta^i \tag{7.12}$$



a) Incremental method (Euler's method)

b) Incremental method with equilibrium correction

Figure 7.1: Internal and external force with and without equilibrium correction (Langen and Sigbjörnsson, 1979)

The iteration using Equation 7.11 has constant $C_{Ik}$ and $K_{Ik}$ for every iteration loop. This procedure is called Modified Newton-Raphson. It is typically used due to lower computational effort than Newton-Raphson, where the incremental matrices are updated for each iteration. The difference is shown in Figure 7.2.

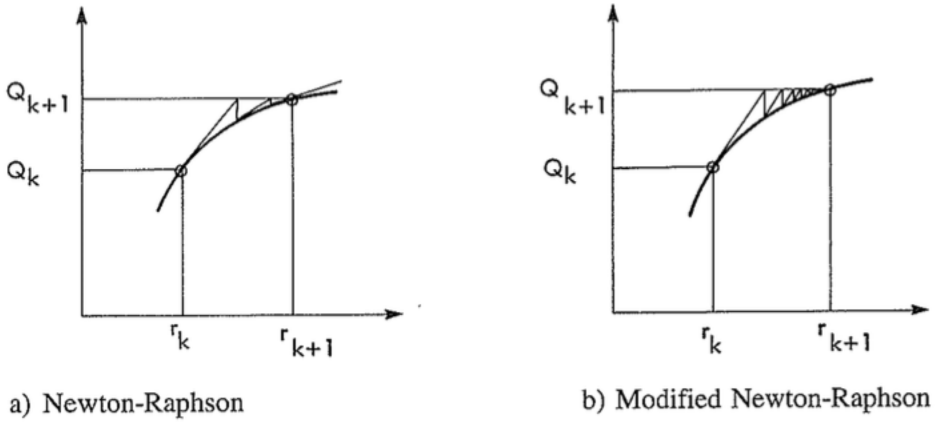


Figure 7.2: Equilibrium iteration (Langen and Sigbjörnsson, 1979)

### Newmark-$\beta$

Newark-$\beta$ is the time integration procedure used in Ulveseter's model. In the linear case, the equation for displacement and velocity at time step $t_{k+1}$ is given as:

$$\dot{r}_{k+1} = \dot{r}_k + (1-\lambda)h\ddot{r}_k + \lambda h\ddot{r}_{k+1} \tag{7.13}$$

$$r_{k+1} = r_k + h\dot{r}_k + (\frac{1}{2} - \beta)h^2\ddot{r}_k + \beta h^2\ddot{r}_{k+1} \tag{7.14}$$

where $h$ is the length of the time step, and $\beta$ and $\lambda$ are weighting factors. The weighting terms determine the nature of the time integration procedure. Using $\beta = 0.25$ and $\lambda = 0.5$, the time integration procedure correspond to the method of constant average acceleration. $\lambda$ determines the artificial damping. If this value equals 0.5, no artificial damping is present. The method can also be formulated on incremental form, which is convenient for nonlinear analyses. More information about this can be found in (Langen and Sigbjörnsson, 1979). The inspiration for the nonlinear Newmark-$\beta$ algorithm used in Ulveseter's model is given in Appendix B.

## 7.2 Limitations and general remarks

Ulveseter's model is limited to cross-flow vortex induced vibrations. Thorsen's original hydrodynamic force model is used even though it is based on a cylinder oscillating in infinite fluid. This is questionable for a pipeline close to the seabed. At positions along the pipe where we have initial contact between the pipe and the seabed, the hydrodynamic force is assumed equal to zero. This is done because of large uncertainties related to the hydrodynamics around a pipe resting on the seabed. Thus the hydrodynamic external force is only active along the initial free span. The program is 2D and axial dofs are neglected as for the original program by Thorsen.

Thus soil friction is not considered.

## Semi nonlinear model

As discussed in (Larsen, 2014), a finite element analysis should start from a stress free configuration. A nonlinear time domain analysis can then be applied to find the static configuration of the pipeline, giving equilibrium between gravitational forces, buoyancy forces and stiffness forces due to the seabed and the structural characteristics of the pipe. In a linear analysis the dynamic response is found around the static configuration, including dynamic forces only. A full nonlinear procedure should apply both static and dynamic loads in the calculation.

Due to convergence problems finding the static equilibrium of the pipeline in still water, Ulveseter's model uses the static configuration found from RIFLEX as input. The response pattern is found around this static equilibrium configuration, and forces due to gravity and buoyancy are not included in the force model. The nonlinearities that are accounted for in the simplified program is varying soil stiffness and damping. Physically, the idea is that when the pipeline is in contact with the seabed both soil damping and soil stiffness is added to the initial damping and stiffness matrix, at the vertical translation dofs. When there is no contact between the pipe and the seabed, the soil damping and soil stiffness terms are removed.

However, when gravitational forces are not included, the pipeline will lift off from the soil in an unphysical way, because there are no gravitational forces holding it down. To be able to include gravity and buoyancy in a satisfactory way, some of the soil springs and dampers are allowed to restrict lift off of, up to a certain limit. We let the soil springs at node positions that are initially penetrating the seabed, be active until the vertical uplift above the seabed equals the static equilibrium position due to gravitational forces only. Nodes initially penetrate the seafloor is referred to as section 1 in Figure 7.3. The behavior of the nonlinear spring force relative to the vertical uplift over the seabed is illustrated in Figure 7.4.

Looking at element number i initially penetrating the seabed, the static equilibrium yields:

$$k_s l \Delta_i = w_s l g \tag{7.15}$$

Where $k_s$ is the soil spring stiffness per length, $l$ is the element length, $\Delta_i$ is the vertical uplift above the seabed at node i, $w_s$ is the submerged weight per length and g is the gravitational acceleration.

Solving Equation 7.15 with respect to the vertical uplift, we get:

$$\Delta_i = \frac{w_s}{k_s} \tag{7.16}$$

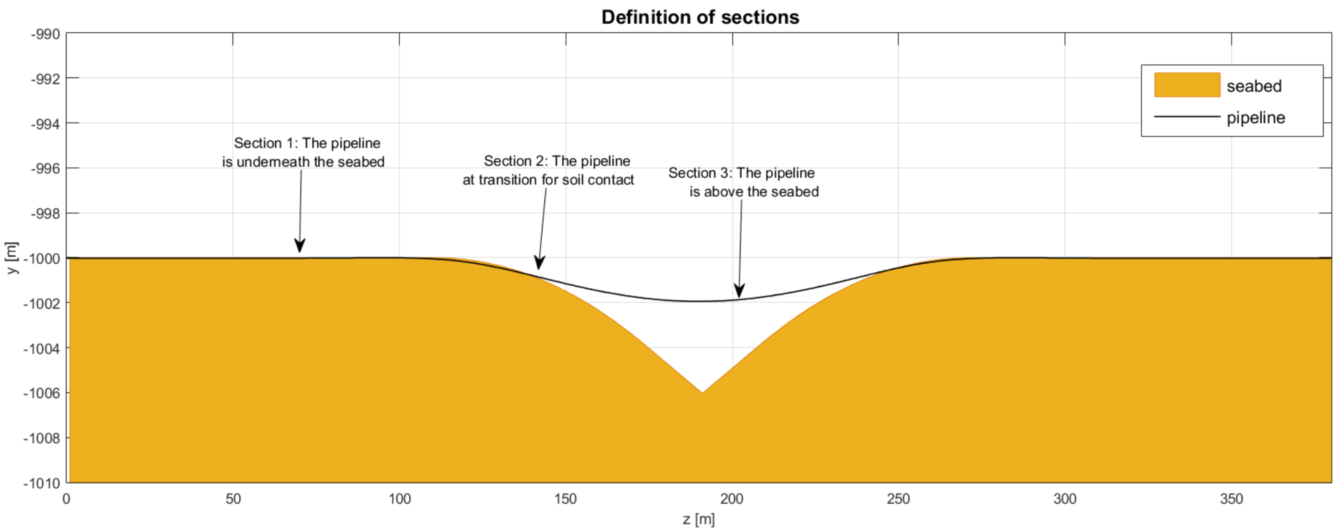


Figure 7.3: Illustration of the three sections where the soil springs have different characteristics to take into account the submerged weight of the pipeline

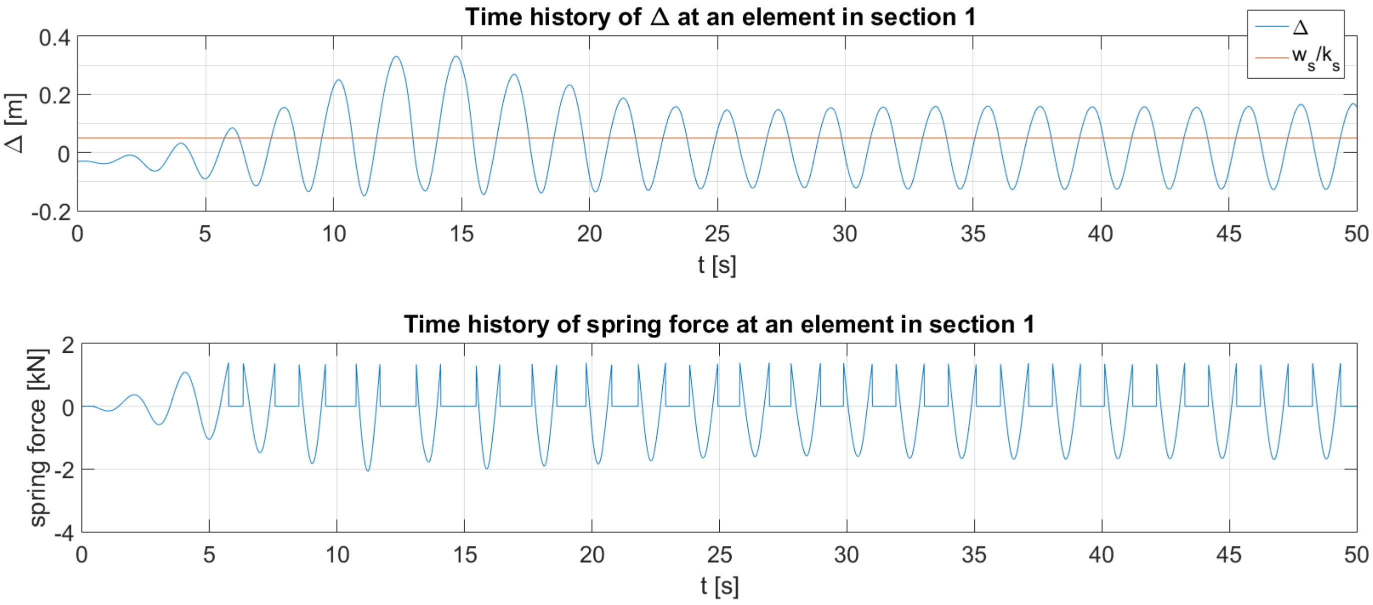


Figure 7.4: The behavior of the nonlinear spring at an element along the pipeline initially penetrating the seabed (section 1), compared to the vertical uplift above the seabed

This means that the vertical translation dofs along the pipe, at nodes initially penetrating the seabed, the soil springs will be active for $\Delta < \frac{w_s}{k_s}$ ($\Delta = 0$ means vertical position of pipeline equals vertical position of seabed).

As an estimate for the element at the transition between initial seabed penetration and no penetration (referred to as section 2 in Figure 7.3), the vertical translation dof is allowed to have a

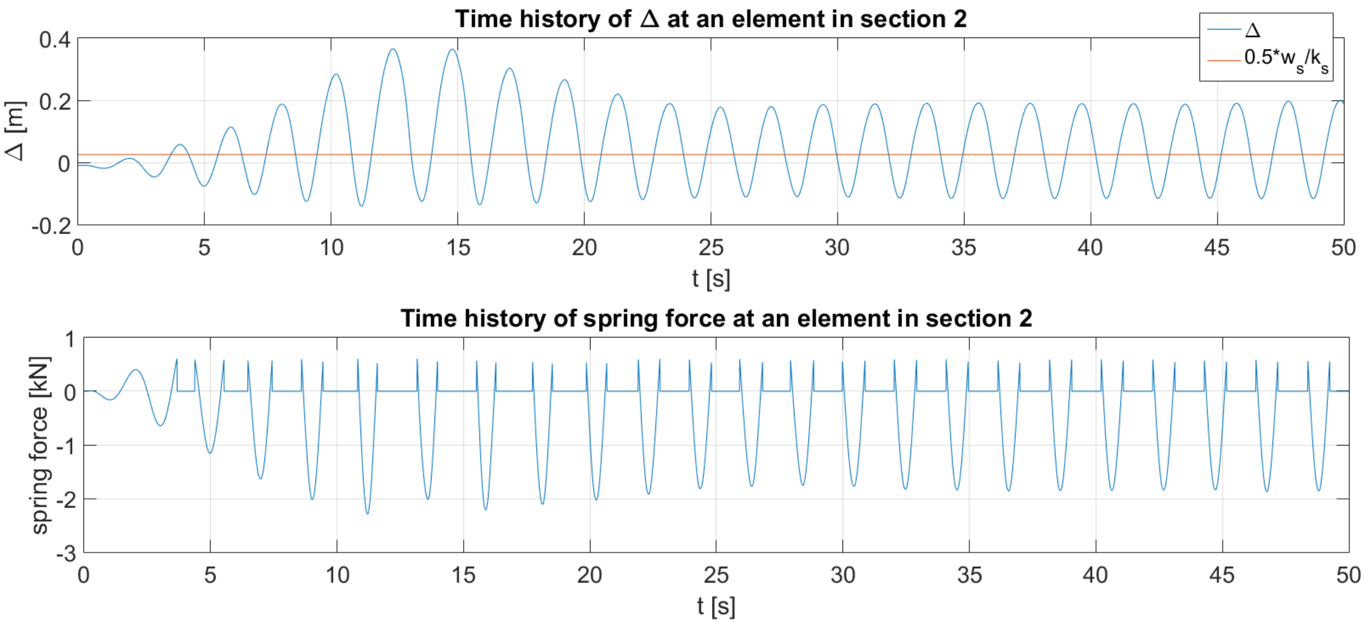


Figure 7.5: The behavior of the nonlinear spring at the transition element for soil contact (section 2), compared to the vertical uplift above the seabed

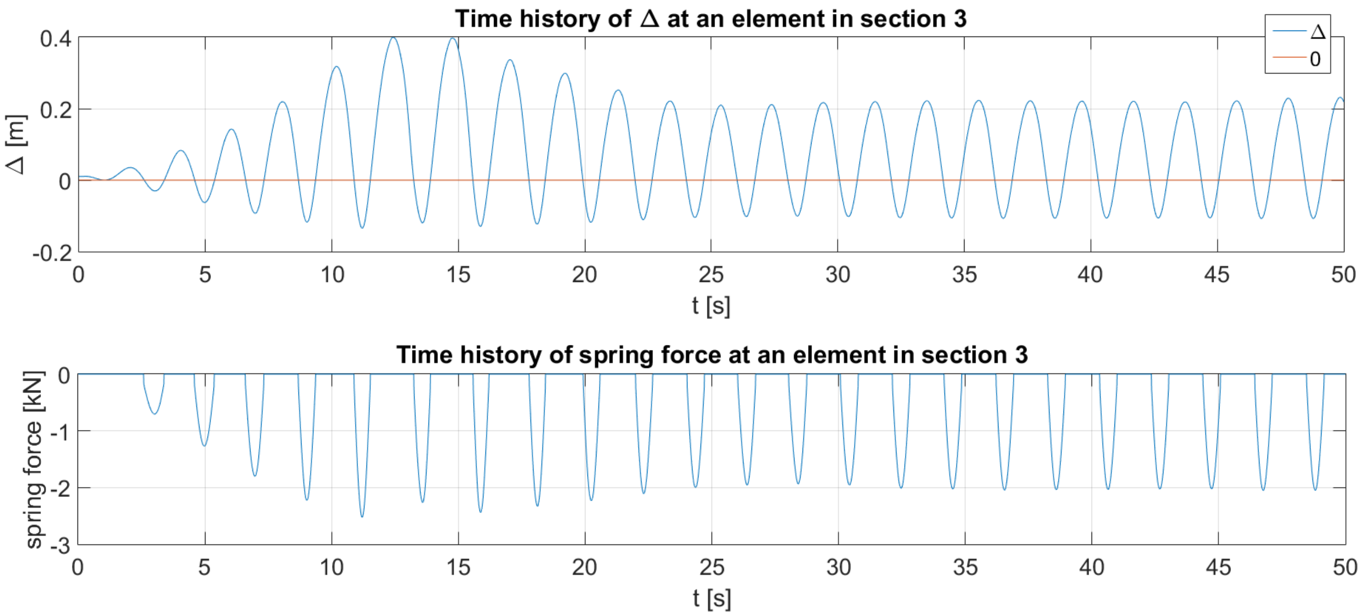


Figure 7.6: The behavior of the nonlinear spring at an element along the pipeline with no initial penetrating the seabed (section 3), compared to the vertical uplift above the seabed

positive displacement of $\Delta < 0.5\frac{w_s}{k_s}$ for the soil spring to be active. The resulting spring force is illustrated in Figure 7.5. The factor 0.5 is added because we assume that half the element is penetrating the seabed, while the other half is not. For the vertical translation dofs in the free span (section 3 in Figure 7.3) the soil springs are only active for $\Delta < 0$. This means that the springs in the free span are only active when the pipe tries to penetrate the seafloor. The resulting nonlinear spring force is shown in Figure 7.6. It is observed that no spring tension is allowed in this case.

### Other limitations

An obvious limitation with Ulveseter's model is the assumption of constant geometry. Ideally, the geometry should be updated for each time step, introducing a local coordinate system for each element. What is done in the present model is that soil contact is investigated for a geometry that is not changing. This is a contradiction. Since VIV is a self limiting process the response amplitudes are considered small, so the assumption of constant geometry is not a significant contribution to errors.

When it comes to the nonlinear springs and dampers used in Ulveseter's model, they are linear with respect to stiffness and damping coefficients. This will not be the case for a real pipeline with soil contact. Other nonlinear effects that can be important for VIV on free spanning pipelines is varying tension and elasto-plastic material behavior. This is not implemented in the MATLAB code.

## 7.3 Validation

### Equilibrium and convergence

To ensure equilibrium between internal and external forces will in theory demand iteration due to the nonlinear soil contact. Due to the fact that Ulveseter's model is not a fully nonlinear model, it is hard to implement a general iteration procedure. It is possible to create an iteration algorithm where we find the exact time the pipeline penetrates the seabed. Then the soil stiffness and soil damping can be added at the correct time giving equilibrium between external and intern forces. However, when the time step is fixed, the pipeline will not feel the soil penetration at the correct time. This results in deviations from the dynamic equilibrium. But when the time step is close to zero, the error introduced is tolerable.

For simplicity, Ulveseter's model uses small time steps, and no iteration procedures. There has been performed several tests to document that the dynamic equilibrium is satisfied. The dynamic equilibrium at time step k is given as:

$$M\ddot{y}_k + C_k\dot{y}_k + K_k y_k = Q_k \tag{7.17}$$

Here, $C_k$ is the sum of structural damping and soil damping at time step k, and $K_k$ is the sum of structural stiffness and soil stiffness at time step k.

Study of Ulveseter's model shows that the equation is satisfied almost at all dofs at all time steps. However, some discrepancies are observed. The equality between left hand side and right hand side of Equation 7.17 is not always satisfied in the area around the shoulders. This must be expected when iteration is not implemented. However, the study shows that the residuals of significant size are rear. This is probably due to the small time step. Equilibrium is always achieved at the time steps before and after the residual. Since no effect of the residual is observed in the pipeline response, the conclusion is that iteration is not needed.

When equilibrium is satisfied we must make sure the displacement, velocity and acceleration are consistent. It is well known that the time derivative of displacement is velocity, and the time derivative of velocity is acceleration. The time history of a translation dof in section 3 is shown in Figure 7.7. We see that the displacement is zero when the velocity has a minimum or maximum. It is also observed that the velocity is zero when acceleration has a maximum or minimum. The consistency is thus satisfied. However, the acceleration shows significant fluctuations. This must be expected when using Newmark-$\beta$ time integration.

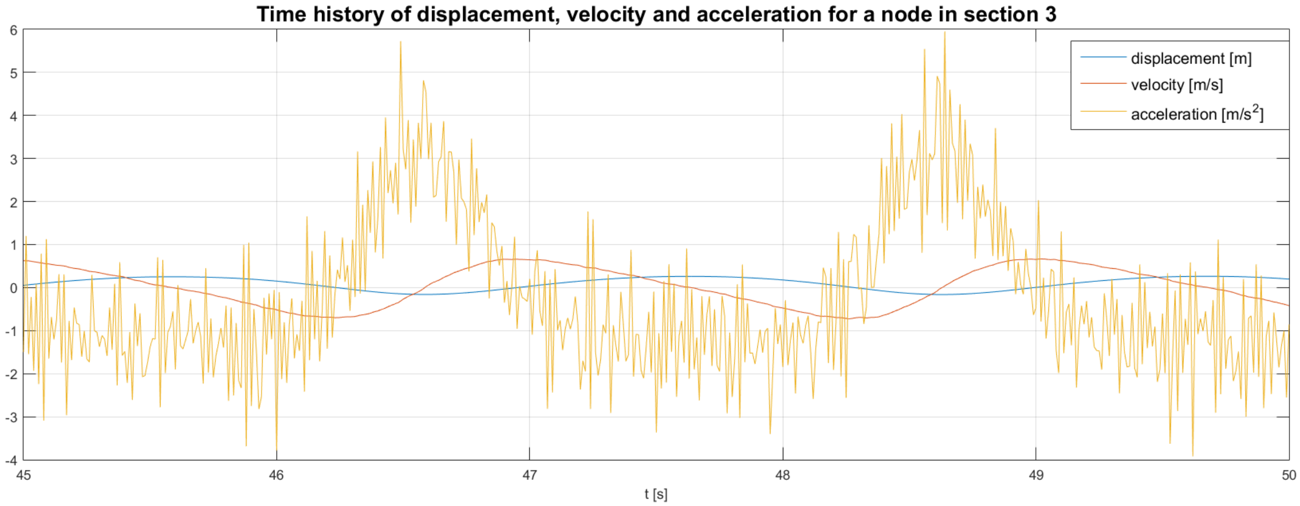


Figure 7.7: Relation between displacement, velocity and acceleration for a translation dof in section 3

## 7.4 Program overview

The simplified nonlinear analysis program is built up as indicated in Figure 7.8. The MATLAB-script "analysis.m" is the main script, and the other blocks in the figure are functions driven from "analysis.m", in the indicated order.

First input parameters need to be defined. Length of the pipe, external diameter, tension, dry mass, bending stiffness, current velocity and water density are some of these parameters. The input parameters also include the soil stiffness and soil damping values. After the initial parameters are defined, numerical inputs are chosen. These include simulation time, length of time step, number of elements in the FE model and parameters used in the Newmark-$\beta$ time integration. This is done in "input_parameters.m".

In "RIFLEX_input.m" the static configuration of the pipe relative to the seabed is extracted from the RIFLEX analysis. This is also the case for the geometry of the seabed. Based on the given input the static length of the free span is calculated and an equivalent horizontal seabed with a square valley in the free span is created. The horizontal seabed is later to be used in the linear analysis.

"FE_model.m" produces the global stiffness matrix as the sum of bending stiffness and geometric stiffness due to tensioning. A consistent mass matrix is created and the damping matrix is assumed to have the form of Rayleigh damping. Rows and columns corresponding to the vertical translation dofs at pipe ends are dropped due to the boundary conditions. The pipe is assumed to be simply supported at this stage.

### 7.4.1 Ulveseter's linear model

If type= 1, as indicated in Figure 7.8, a linear analysis will be performed. The structure of "linear_analysis.m" is given in Figure 7.9. A linear analysis means that the damping matrix and stiffness matrix are constant through the time integration process of the dynamic equilibrium equation. Thus to be able to account for the pipe-soil interaction, the soil stiffness and soil damping values must be added to all the vertical translations dofs initially in contact with the seabed, before the time integration starts. This is performed in the functions "soil_stiffness.m" and "soil_damping.m".

The linear time integration is performed in the function "linear_time_domain.m". Here, the current is aloud to build up slowly from 0 m/s to the value given as input. The hydrodynamic forces are calculated for the translation dofs in the free span. Here Thorsen's model is applied. Then, an algorithm for linear Newmark-$\beta$ time integration gives the response, velocity and acceleration for the next time step. This loop will continue until the simulation time corresponds to the chosen value.

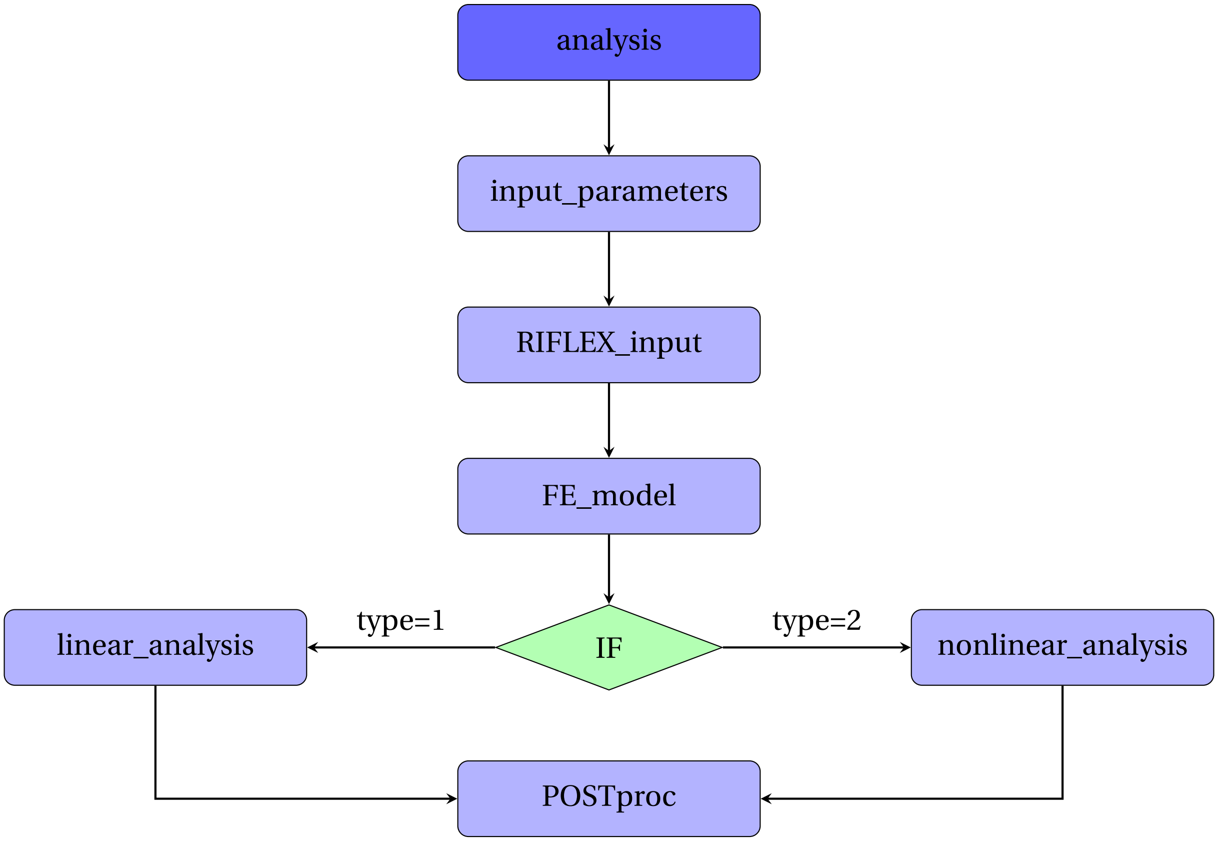


Figure 7.8: Program overview

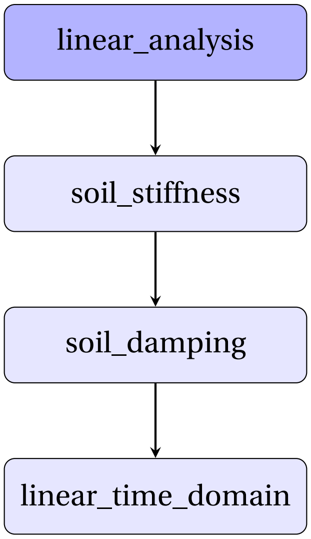


Figure 7.9: Overview of the function "linear_analysis.m"

### 7.4.2 Ulveseter's nonlinear model

If type= 2 (see Figure 7.8) "analysis.m" will instead of performing a linear analysis, do a nonlinear analysis. The structure of "nonlinear_analysis.m" is shown in Figure 7.10. The main difference between the linear and nonlinear analysis is that the nonlinear analysis requires that the stiffness matrix and damping matrix are calculated for every time step of the time integration. This is done in the function "nonlinear_soil.m". At every time step this function uses the information about the seabed position and the response of the pipe to check if the pipe is in contact with the soil. If there is contact, the input value of the soil stiffness and damping stiffness is added to the vertical translation dofs where we have pipe-soil contact. If there is no contact, "nonlinear_soil.m" will check the next node along the pipeline without adding stiffness and damping terms. To include gravity, the pipe is partly restricted against lift off in section 1 and 2 (see Figure 7.3, Figure 7.4 and Figure 7.5). This is implemented in "nonlinear_soil.m".

The hydrodynamic force is calculated in "hydroforce.m" for time step $t_{k+1}$. This is the same hydrodynamic force model as applied in the linear analysis. We now have enough information to apply an algorithm for nonlinear Newmark-$\beta$ time integration. This method solves the dynamic equilibrium equation incrementally. The output is response, velocity and acceleration for time step $t_{k+1}$. We then look at the next time step, indicated with the block $k = k+1$ in Figure 7.10. The same procedure is applied over and over again until the simulation time is reached.

### 7.4.3 Post-processing

From the linear or nonlinear analysis, we get matrices consisting of the pipeline response at all dofs for all time steps. This data is post-processed in the function "POSTproc.m" (see Figure 7.8). The function "STRESSamp.m" finds the curvature and calculates the maximum stress amplitude looking at the maximum and minimum value of the stresses at each node. The stress amplitudes are found considering the last 10% of the simulation time to avoid transient effects. The stress is found as:

$$\sigma = E\frac{D}{2}\frac{\partial^2 y}{\partial z^2} \tag{7.18}$$

Where E is the E-module, D is the pipe diameter and $\frac{\partial^2 y}{\partial z^2}$ is the curvature of the pipeline.

The stress amplitude is given as:

$$\Delta\sigma_a = \frac{1}{2}(\Delta\sigma_{max} + \Delta\sigma_{min}) \tag{7.19}$$

Where $\Delta\sigma_{max}$ is the maximum stress at a specific node and $\Delta\sigma_{min}$ is the minimum stress at the same node. This information makes it possible to find the stress amplitude distribution over the

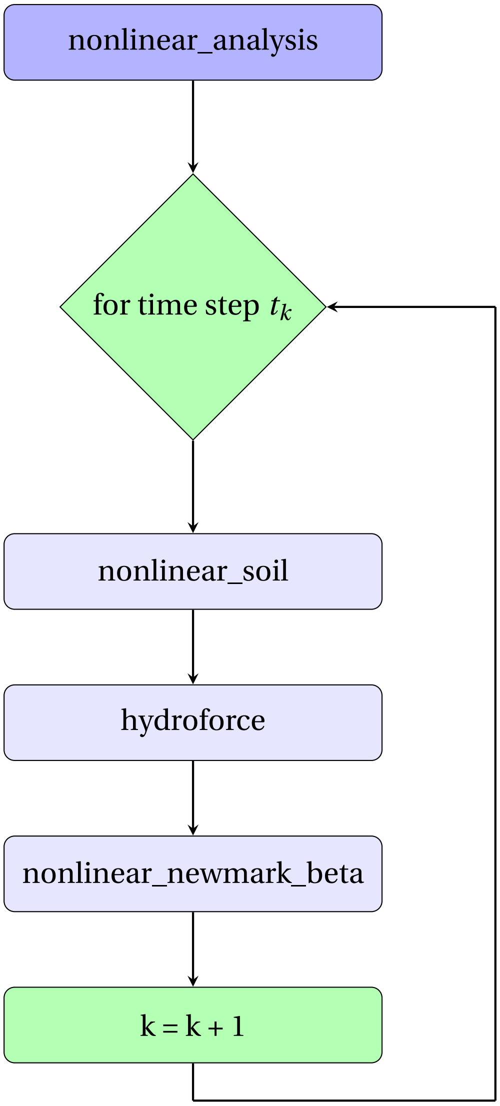


Figure 7.10: Overview of the function "nonlinear_analysis.m"

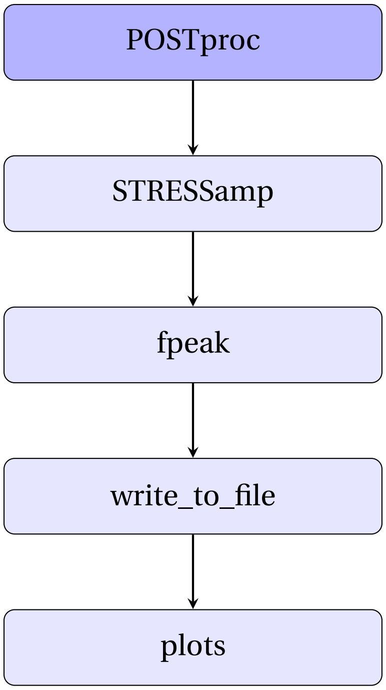


Figure 7.11: Overview of the function "POSTproc.m"

length of the pipeline.

Next, the response frequency is found from the spectrum in "fpeak.m". Then, "write_to_file.m" stores results in text-files. Lastly, the results are visualized through the function "plots.m".

# Chapter 8

# Case studies

In this chapter Ulveseter's model is tested against VIVANA analyses and a VIVANA/RIFLEX time-domain analysis. For most cases, the soil damping is varied to see how it influences the pipeline response, using linear models and the nonlinear model. The different vertical soil damping values are referred to as a ratio between applied soil damping and the estimated critical soil damping. The symbol $\lambda$ is used as the damping ratio, and the procedure for finding this value is outlined in Section 6.1.3. A test on how the current speed influences the response modes has also been performed.

## 8.1 Case 1 - Extension of the Project Thesis

The first case is an extension of the Project Thesis (Ulveseter, 2014). A pipeline on an irregular seabed is subjected to a constant current in the x-direction (see Figure 5.1). Both VIVANA and Ulveseter's model are used to calculate the vortex induced vibrations in the cross-flow direction. The data for the case is given in Table 8.1.

Table 8.1: Data for Case 1

| **Name** | **Symbol** | **Size** | **Dimension** |
|---|---|---|---|
| Length | L | 180 | $m$ |
| Diameter | D | 0.4 | $m$ |
| Bending stiffness | EI | $8.9 * 10^7$ | $Nm^2$ |
| Mass per unit length (air) | m | 217.96 | $kg/m$ |
| End tension | T | $10 * 10^3$ | $N$ |
| Current velocity | U | 1.2 | $m/s$ |

All linear models are tested for soil damping ratio between 0 and 25% of estimated critical soil damping (see Equation 6.9), and the soil stiffness is determined from Equation 6.8. The non-linear Ulveseter's model is used between 10% and 25% critical soil damping. The structural damping is set equal to zero in all cases. For both VIVANA and Ulveseter's model, the static configuration of the pipe relative to the seabed is found from RIFLEX. This is the starting point for the VIV analysis. It is shown in Figure 8.1.

## Results

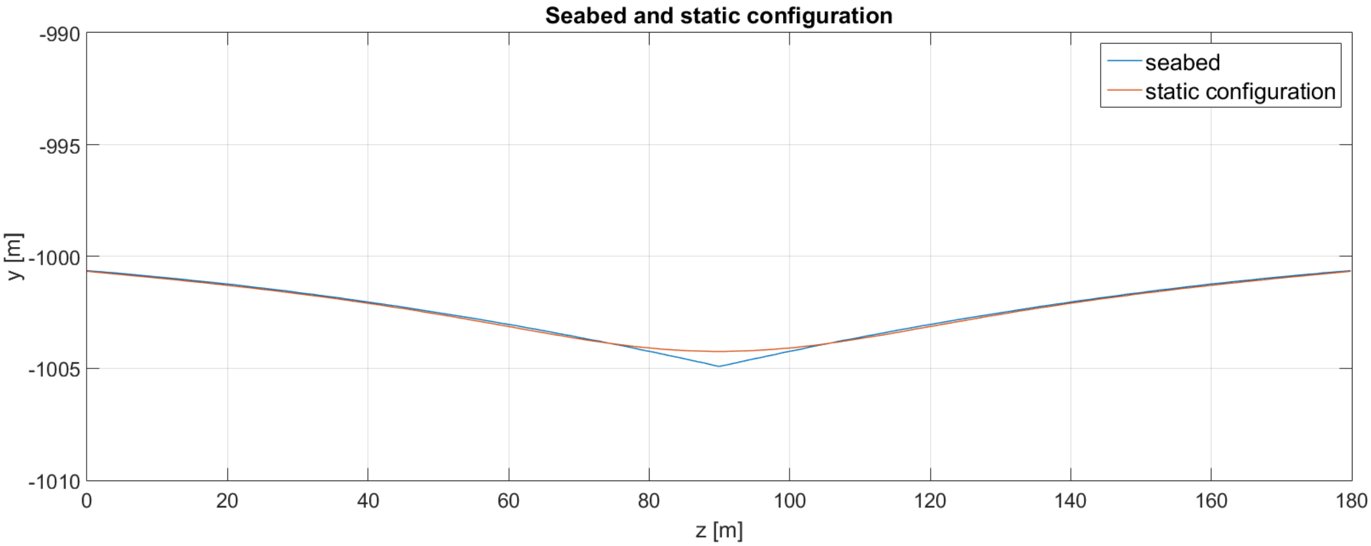


Figure 8.1: The seabed and the static configuration found from RIFLEX

### Ulveseter's linear model

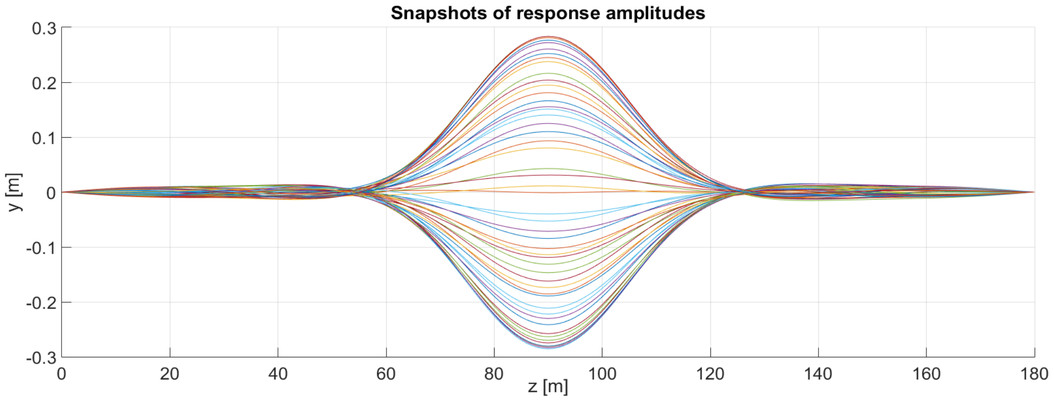


Figure 8.2: Snapshots of the cross-flow response from Ulveseter's linear model, for $\lambda = 0.10$

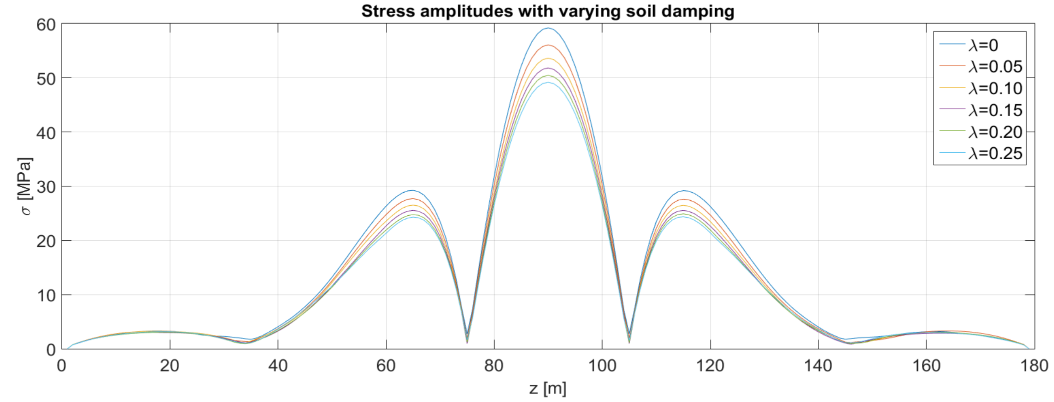


Figure 8.3: Stress amplitudes for different vertical soil damping values, from Ulveseter's linear model

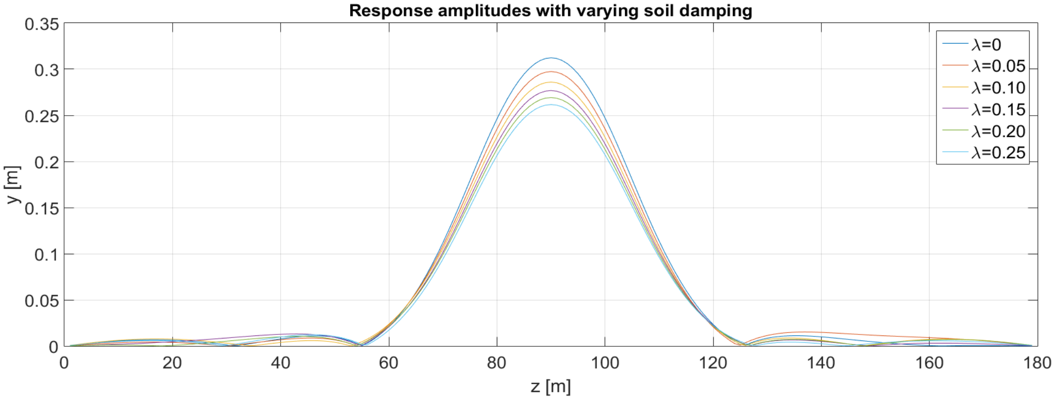


Figure 8.4: Response amplitudes for different vertical soil damping values, from Ulveseter's linear model

### Ulveseter's nonlinear model

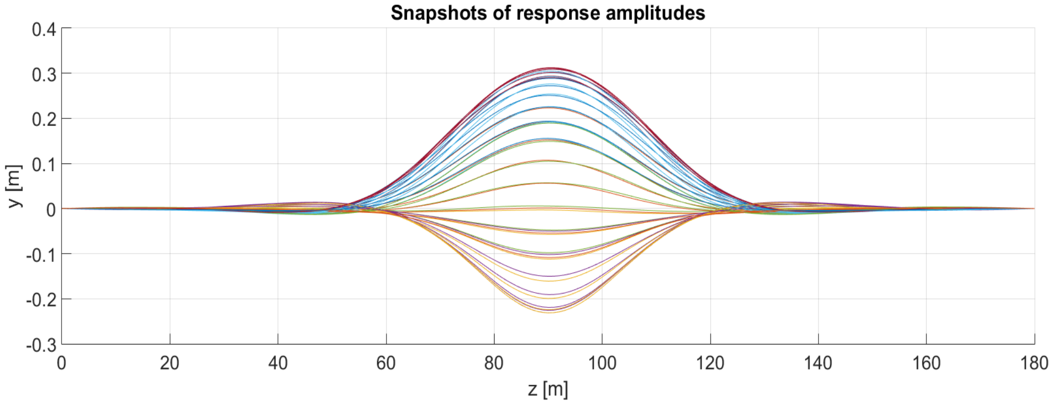


Figure 8.5: Snapshots of the cross-flow response from Ulveseter's nonlinear model, for $\lambda = 0.10$

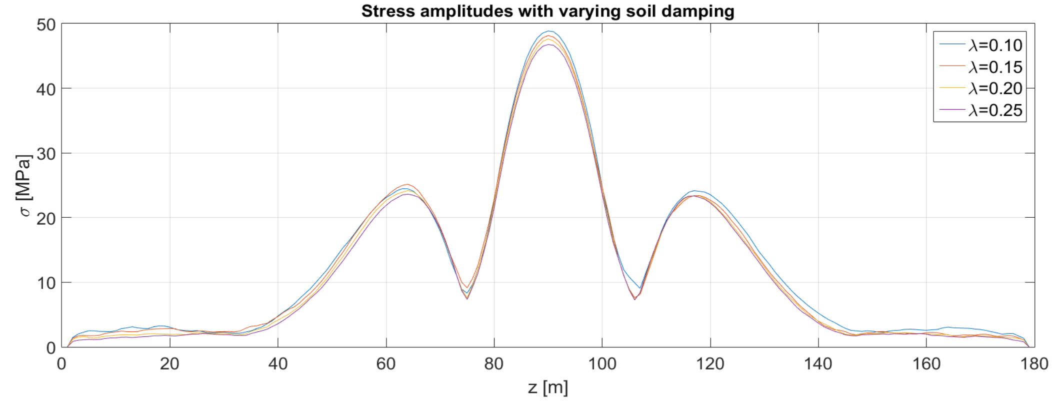


Figure 8.6: Stress amplitudes for different vertical soil damping values, from Ulveseter's nonlinear model

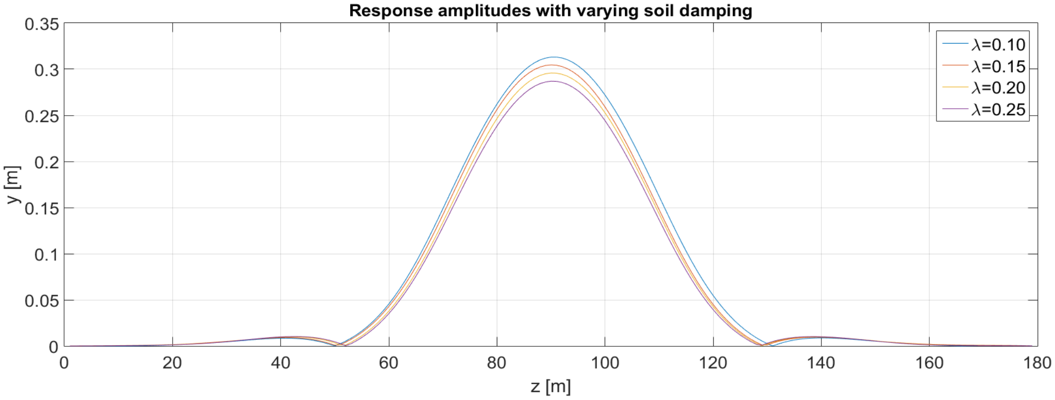


Figure 8.7: Response amplitudes for different vertical soil damping values, from Ulveseter's nonlinear model

### VIVANA

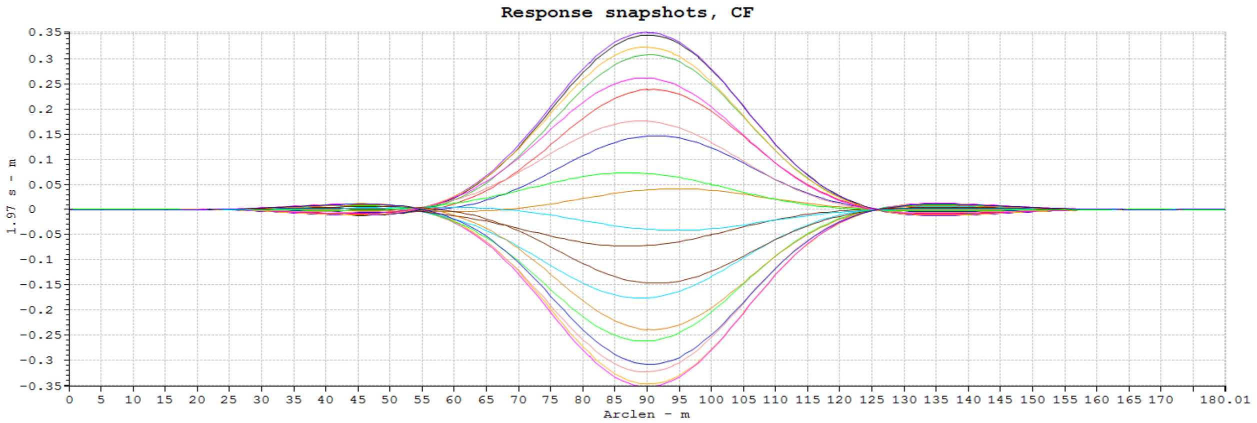


Figure 8.8: Snapshots of the cross flow response from VIVANA, for $\lambda = 0.10$

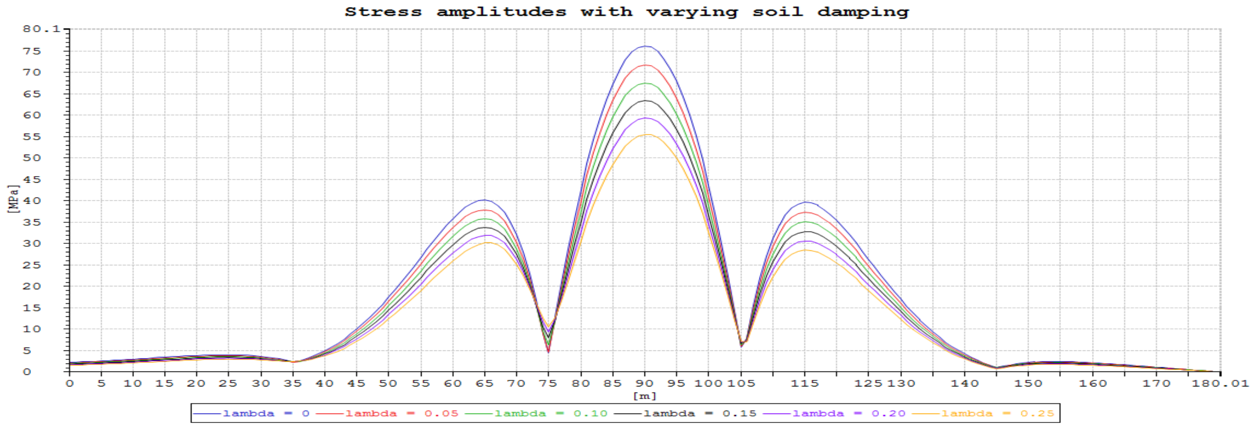


Figure 8.9: Stress amplitudes for different values of the vertical soil damping, from VIVANA

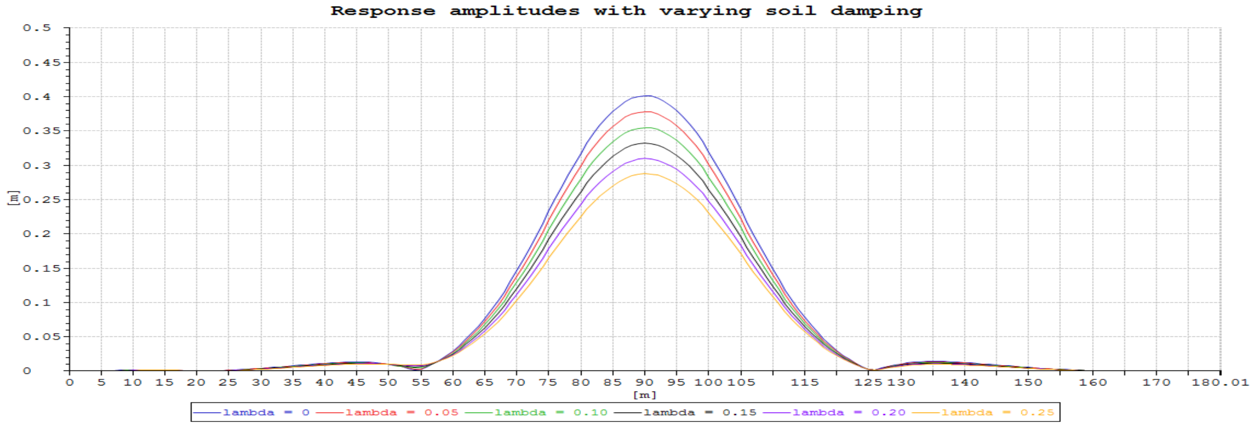


Figure 8.10: Response amplitudes for different values of the vertical soil damping, from VIVANA

### Comparison between the models

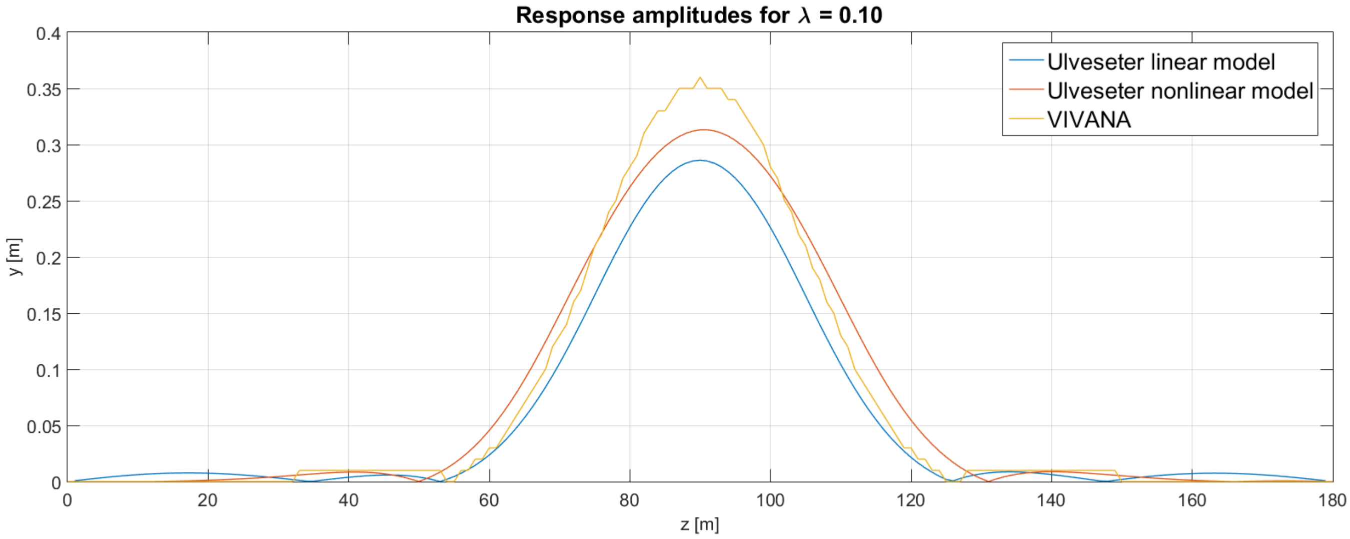


Figure 8.11: Comparison of maximum cross-flow response, for VIVANA and Ulveseter's model

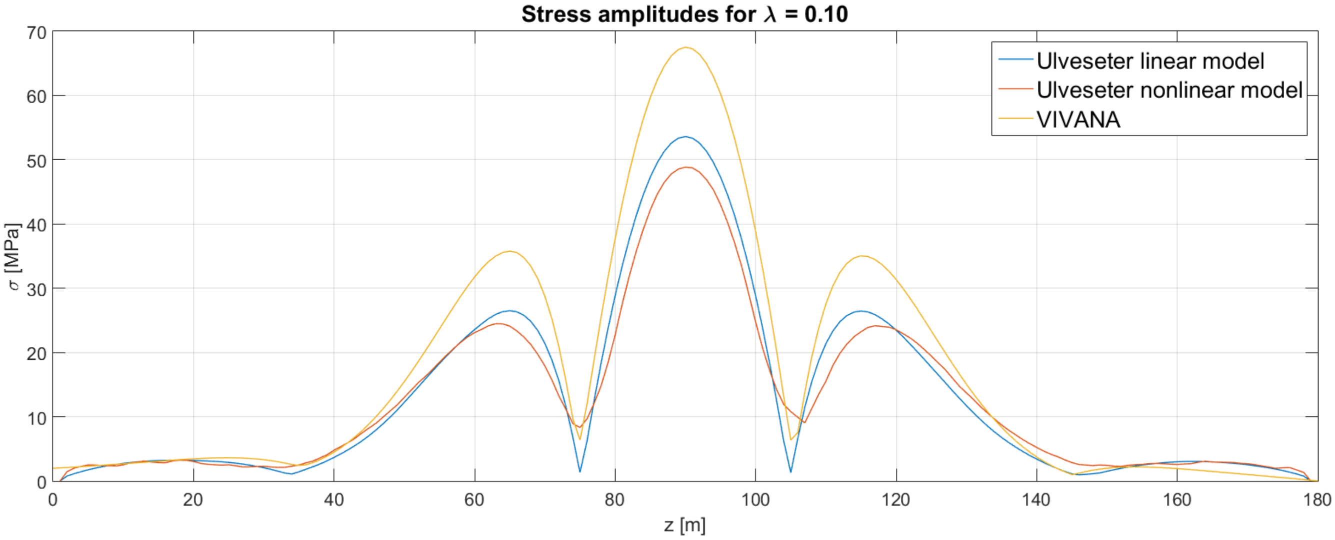


Figure 8.12: Comparison of maximum cross-flow stress amplitudes, for VIVANA and Ulveseter's model

### Observations

- For increasing soil damping, response amplitudes and stress amplitudes decrease in magnitude (Figure 8.3, 8.4, 8.6, 8.7, 8.9, 8.10). This trend is the same for all models. However, how much the response and stress amplitudes decrease for increasing soil damping is different. Ulveseter's nonlinear model is less affected by the soil damping.
- From the comparison between the results at 10% of estimated local critical damping (Figure 8.11, 8.12), it is clear that VIVANA predicts larger stresses and response amplitudes than Ulveseter's model.
- Ulveseter's nonlinear model has varying touch down points (Figure 8.5), a larger response amplitude than Ulveseter's linear model (Figure 8.11), but smaller stress amplitudes (Figure 8.12).

## 8.2 Case 2 - Realistic pipeline model

Case 2 is VIV analysis for a pipeline based on the same realistic pipeline dimensions and current condition as used by (Larsen et al., 2004). Ulveseter's linear and nonlinear model is compared to results from VIVANA. The VIV analysis is a cross-flow analysis only, and the pipeline is facing a constant current in the x-direction, as for Case 1.

Due to convergence problems in VIVANA, the case with 0 soil damping is not part of the VIVANA analysis. The VIVANA results are presented for a range between 5% and 25% of estimated critical soil damping, where convergence was achieved. Ulveseter's linear model is tested for 0 to 25% of critical soil damping, and Ulveseter's nonlinear model is tested for 10% to 25% of critical soil damping. The data for the case is given in Table 8.2

Table 8.2: Data for Case 2

| Name | Symbol | Size | Dimension |
|---|---|---|---|
| Length | L | 380 | $m$ |
| Diameter | D | 0.55 | $m$ |
| Bending stiffness | EI | $2.9 * 10^8$ | $Nm^2$ |
| Mass per unit length (air) | m | 315 | $kg/m$ |
| End tension | T | $450 * 10^3$ | $N$ |
| Current velocity | U | 0.7 | $m/s$ |
| Soil stiffness | $k_s$ | $40 * 10^3$ | $N/m^2$ |

Table 8.2 gives information about the soil stiffness $k_s$. This value is the same as used by (Larsen et al., 2004), where it is referred to as the soft bottom case. Equation 6.9 is used to find the relative vertical soil damping, as for Case 1.

## Results

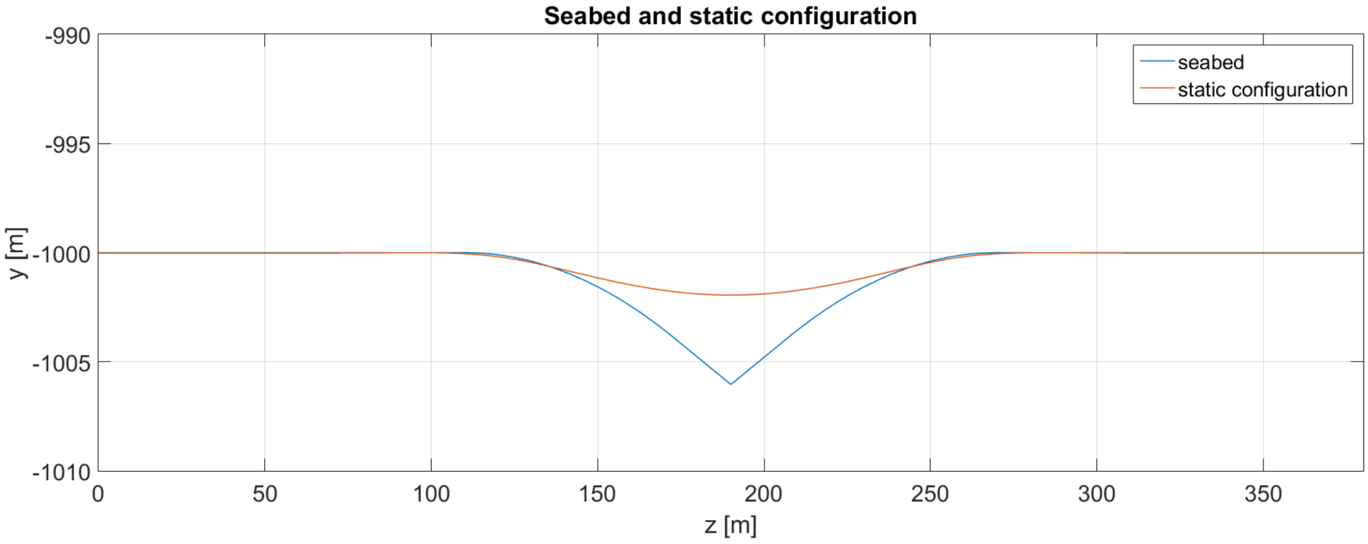


Figure 8.13: The seabed and the static configuration found from RIFLEX

### Ulveseter's linear model

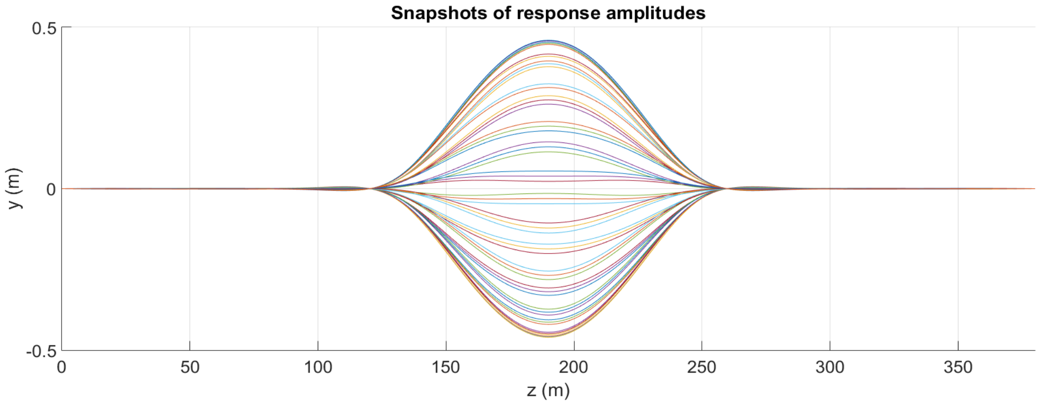


Figure 8.14: Snapshots of the cross-flow response, from Ulveseter's linear model, for $\lambda = 0.10$

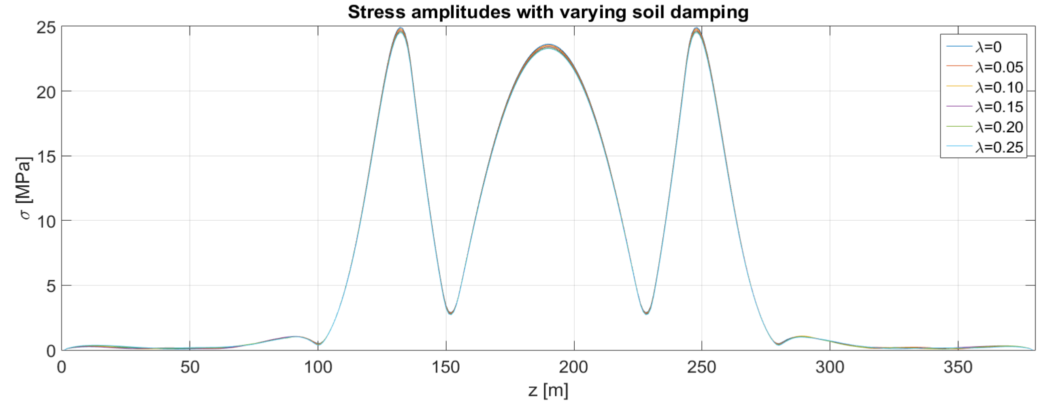


Figure 8.15: Stress amplitudes for different vertical soil damping values, from Ulveseter's linear model

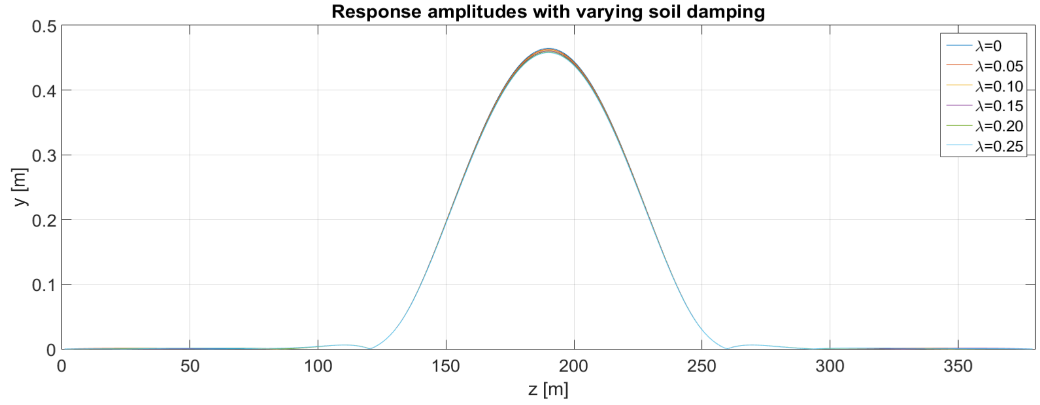


Figure 8.16: Response amplitudes for different vertical soil damping values, from Ulveseter's linear model

### Ulveseter's nonlinear model

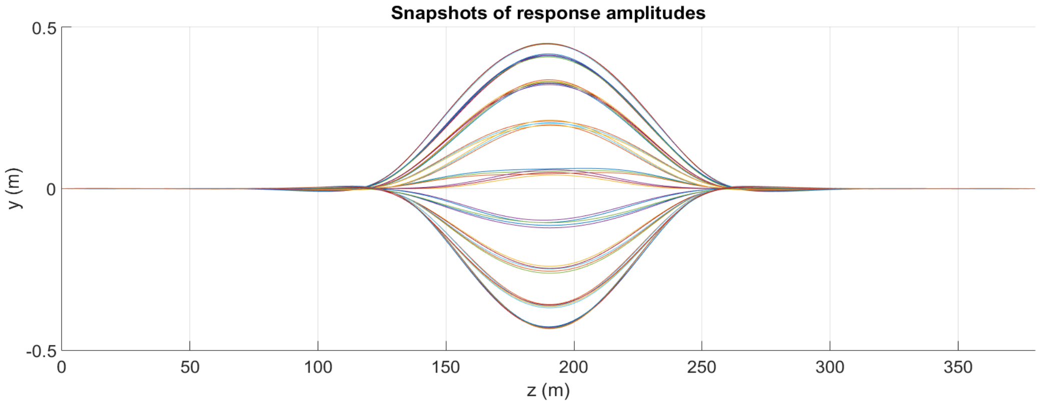


Figure 8.17: Snapshots of the cross-flow response, from Ulveseter's nonlinear model, for $\lambda = 0.10$

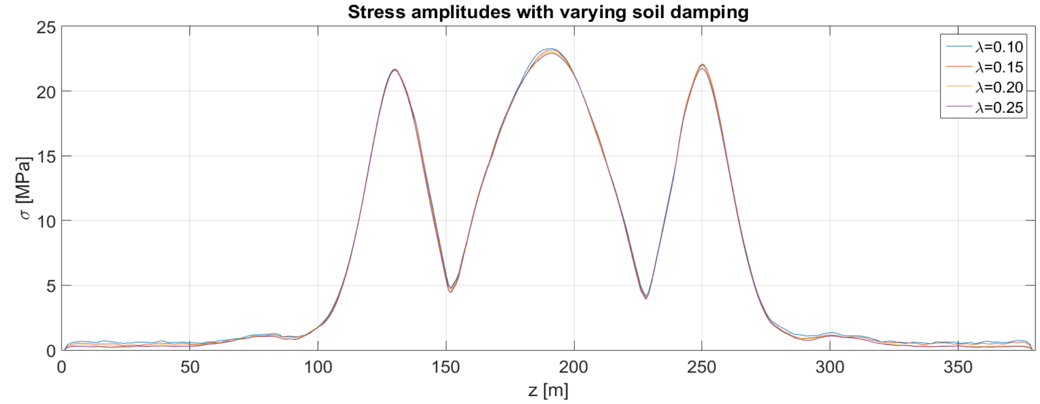


Figure 8.18: Stress amplitudes for different values of the vertical soil damping, from Ulveseter's nonlinear model

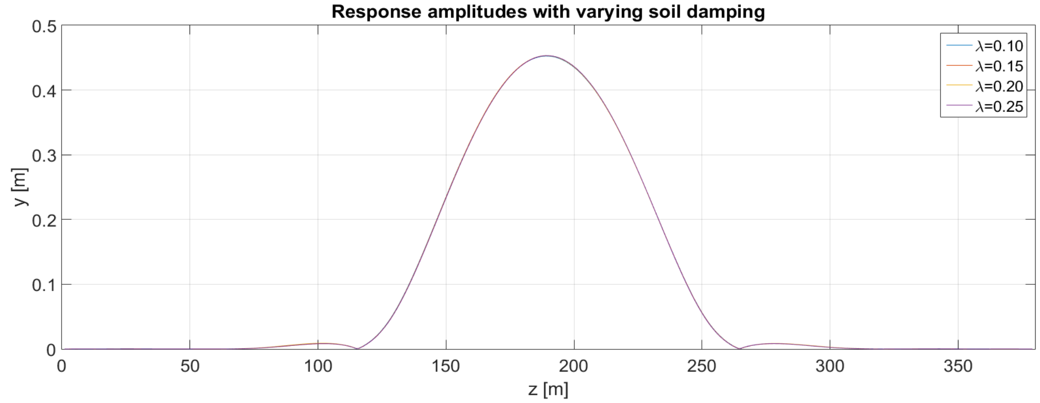


Figure 8.19: Response amplitudes for different vertical soil damping values, from Ulveseter's nonlinear model

### VIVANA

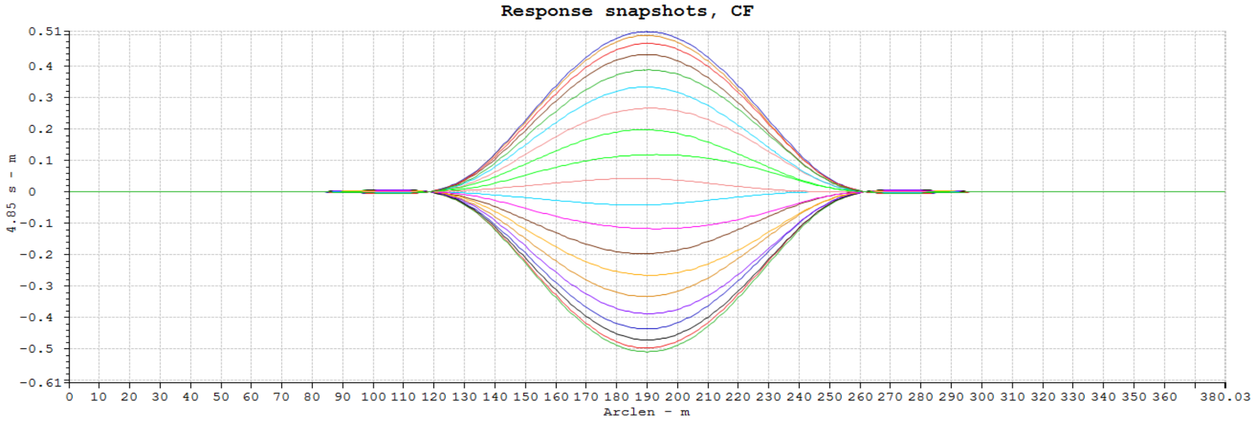


Figure 8.20: Snapshots of the cross-flow response, from VIVANA, for $\lambda = 0.10$

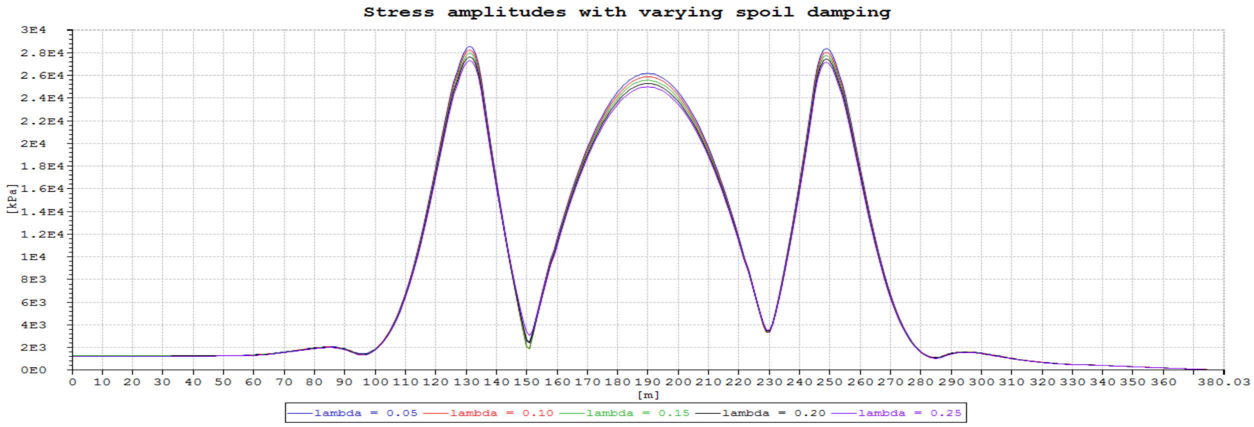


Figure 8.21: Stress amplitudes for different values of the vertical soil damping, from VIVANA

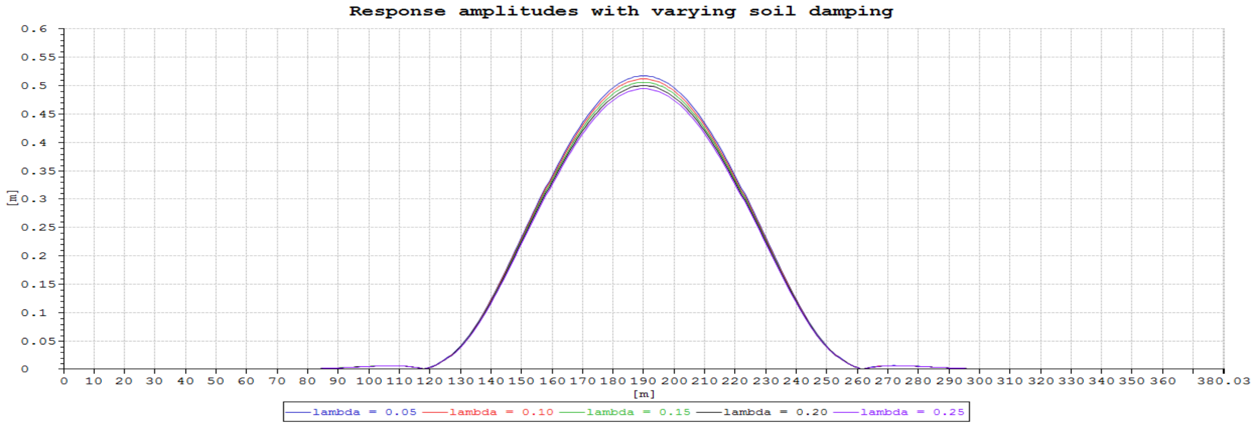


Figure 8.22: Response amplitudes for different values of the vertical soil damping, from VIVANA

### Comparison between the models

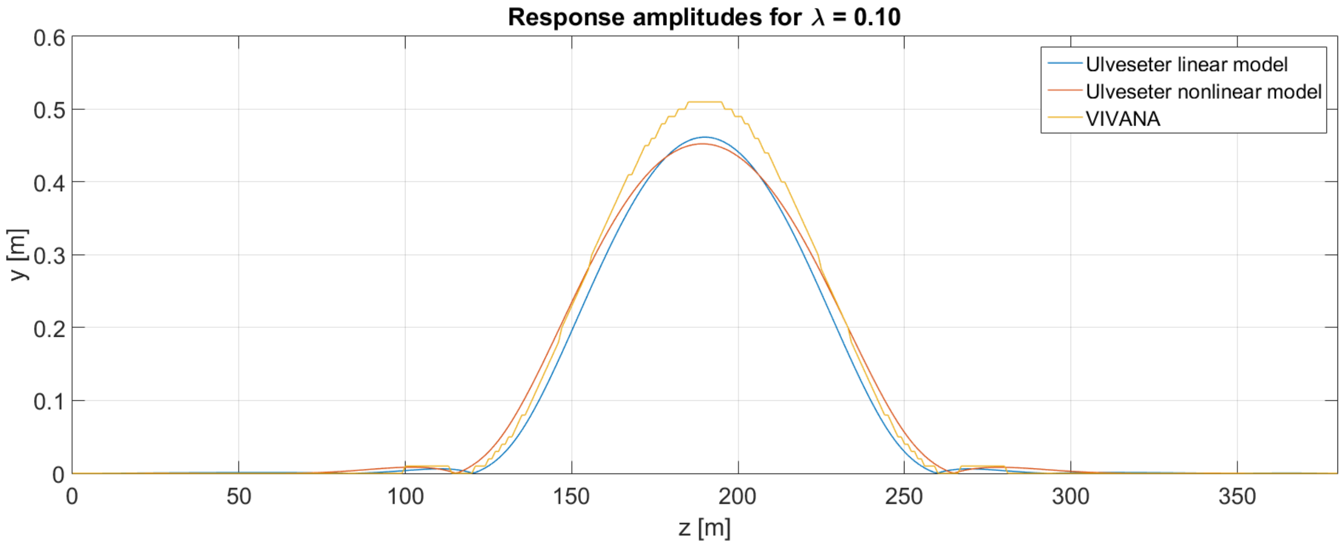


Figure 8.23: Comparison of maximum cross-flow response, for VIVANA and Ulveseter's model

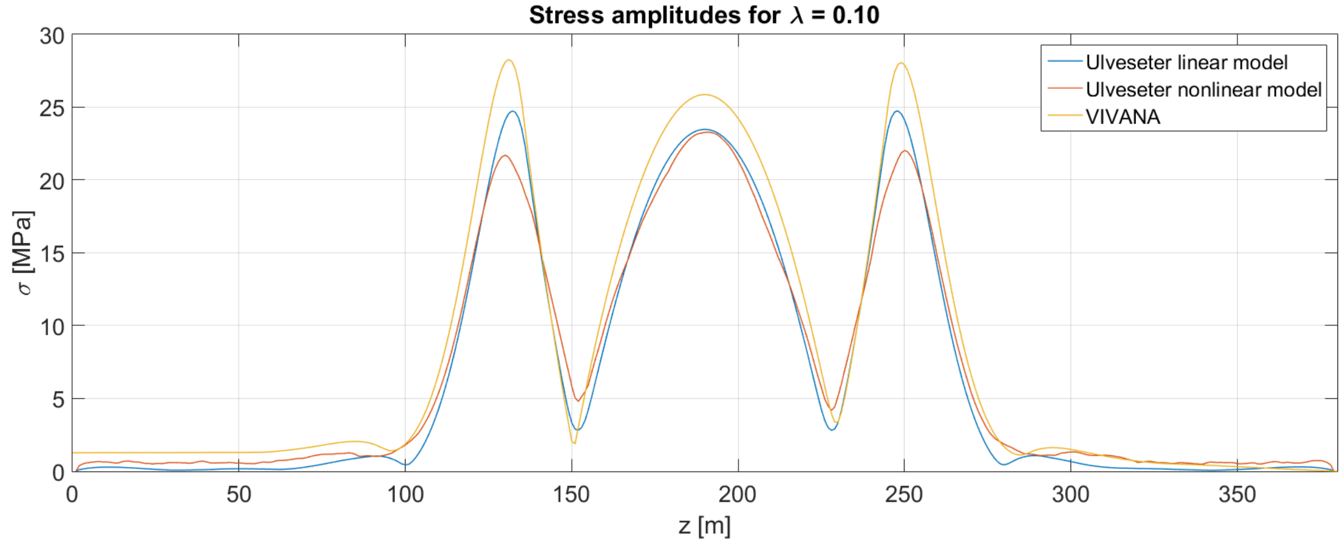


Figure 8.24: Comparison of maximum cross-flow stress amplitudes, for VIVANA and Ulveseter's model

### Observations

- From Figure 8.15, 8.16, 8.18, 8.19, 8.21, 8.22, the response amplitudes and stress amplitudes are little affected by the soil damping. This is the case for both Ulveseter's model and VIVANA.
- In Figure 8.24, the stress amplitudes around the touch down points are smaller for Ulveseter's nonlinear model than for the linear models.
- The seabed profile has a large curvature at the shoulders (Figure 8.13) compared to Case 1 (Figure 8.1).
- Ulveseter's nonlinear model predicts almost symmetric response about the z-axis, and the touch down points are almost fixed (Figure 8.17).
- As for Case 1, VIVANA predicts larger responses than Ulveseter's model. The difference is smaller than for Case 1.

## 8.3 Case 3 - Realistic pipeline model with different seabed profile

In Case 3, the goal is to investigate the effect of the seabed profile. The same pipeline facing the same current as for Case 2 is investigated (see Table 8.2 for data). The seabed in Case 3 has a smaller curvature at the shoulders, than what was applied in Case 2. This can be seen in Figure 8.25 and 8.26. As for Case 2, Ulveseter's linear model is tested for relative soil damping ratio between 0 and 25% of estimated critical soil damping. Ulveseter's nonlinear model is tested for soil damping values between 10% and 25% of estimated critical soil damping. Due to convergence problems VIVANA is only tested for $\lambda = 0.5, 0.10, 0.20$ and 0.25. It is still possible to see the trends and do comparisons between the different results.

## Results

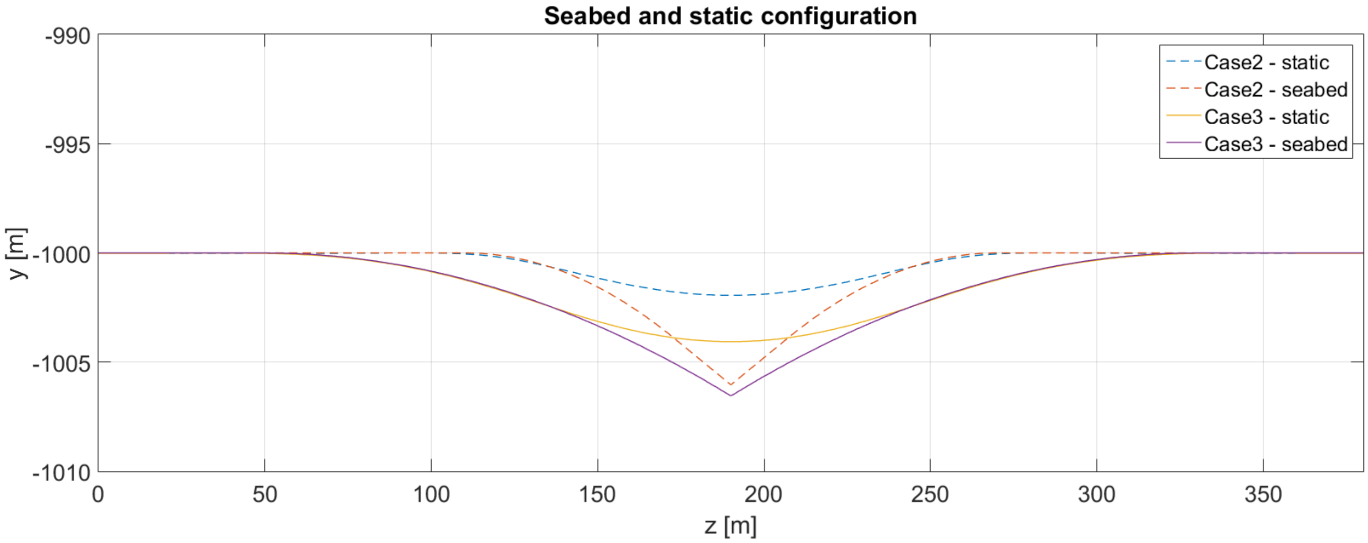


Figure 8.25: The seabed and the static configuration for Case 2 and Case 3 found from RIFLEX

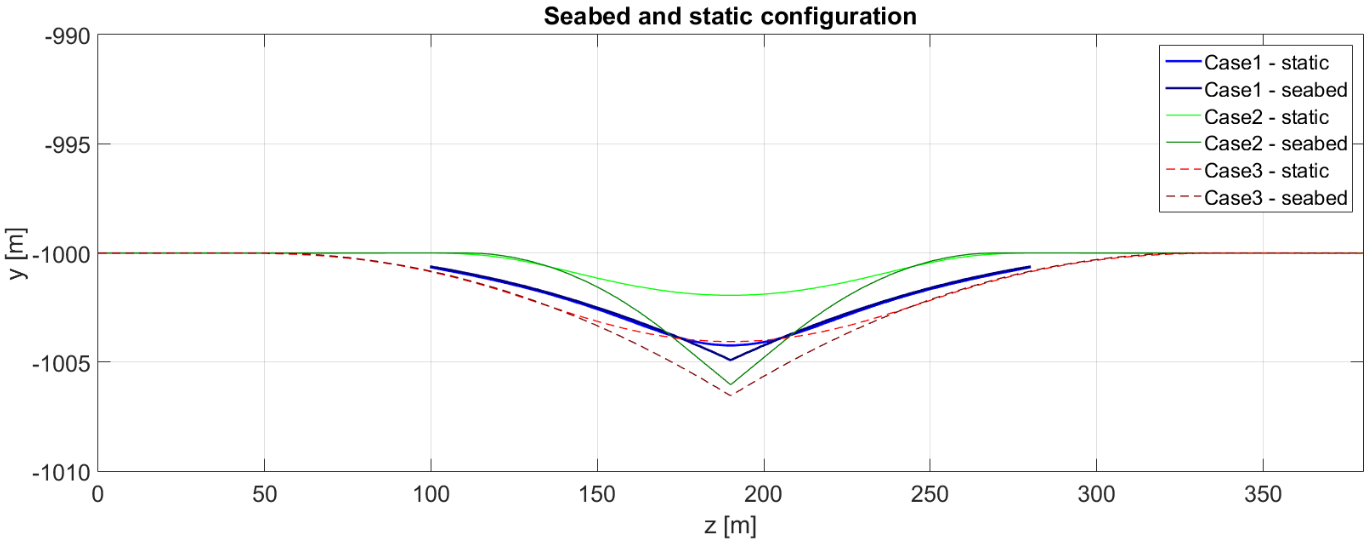


Figure 8.26: Comparison of seabed and static configuration for Case 1, 2 and 3

### Ulveseter's linear model

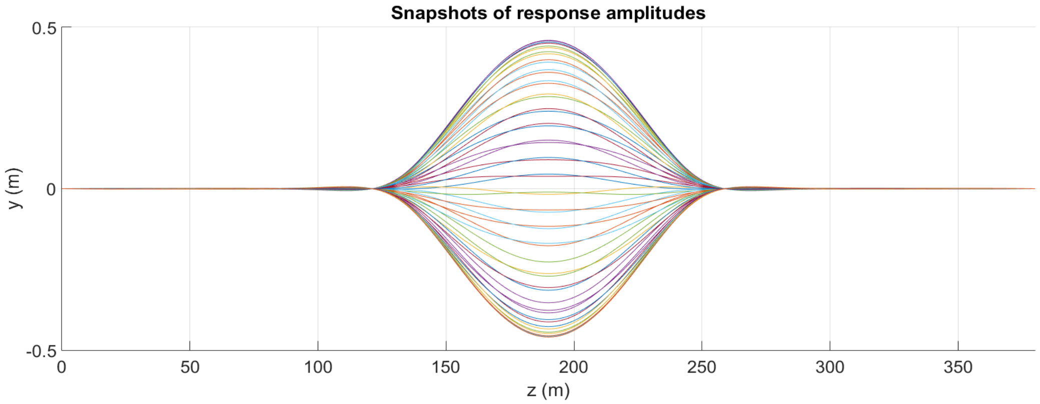


Figure 8.27: Snapshots of the cross-flow response, from Ulveseter's linear model, for $\lambda = 0.10$

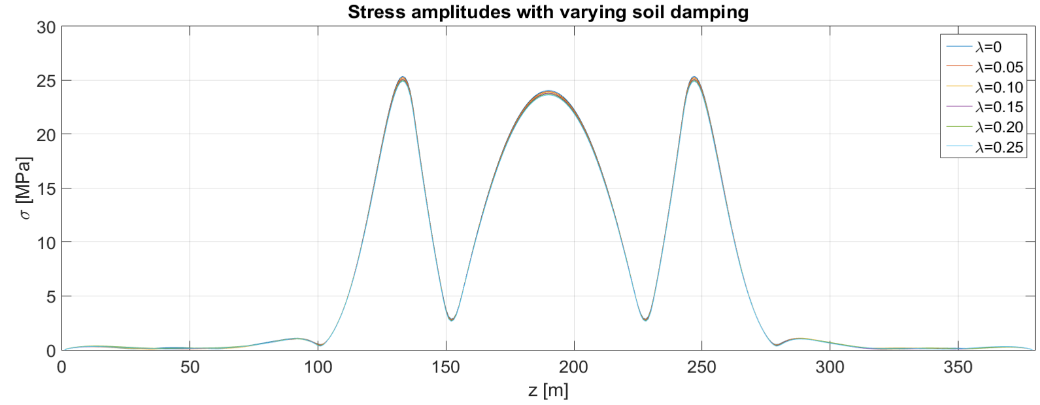


Figure 8.28: Stress amplitudes for different vertical soil damping values, from Ulveseter's linear model

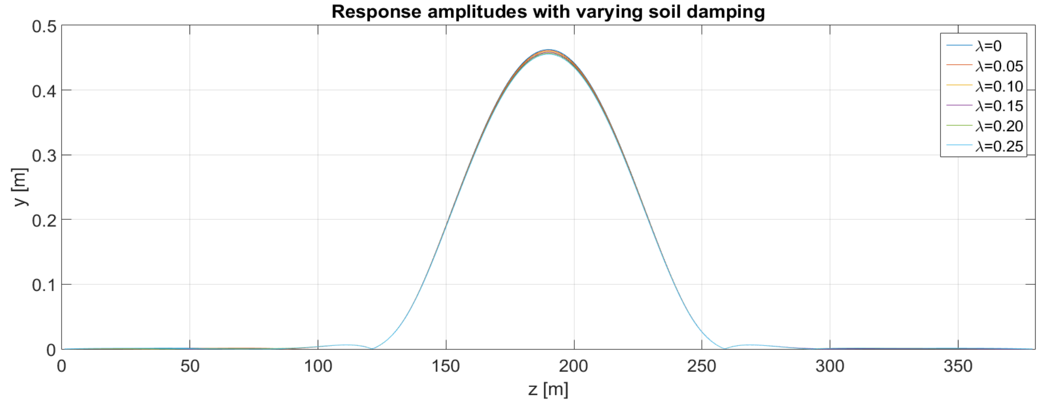


Figure 8.29: Response amplitudes for different vertical soil damping values, from Ulveseter's linear model

### Ulveseter's nonlinear model

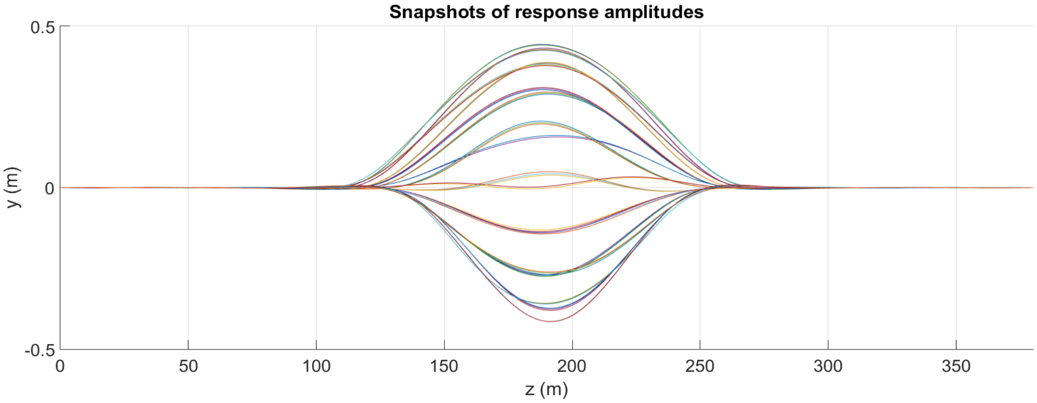


Figure 8.30: Snapshots of the cross-flow response from Ulveseter's nonlinear model, for $\lambda = 0.10$

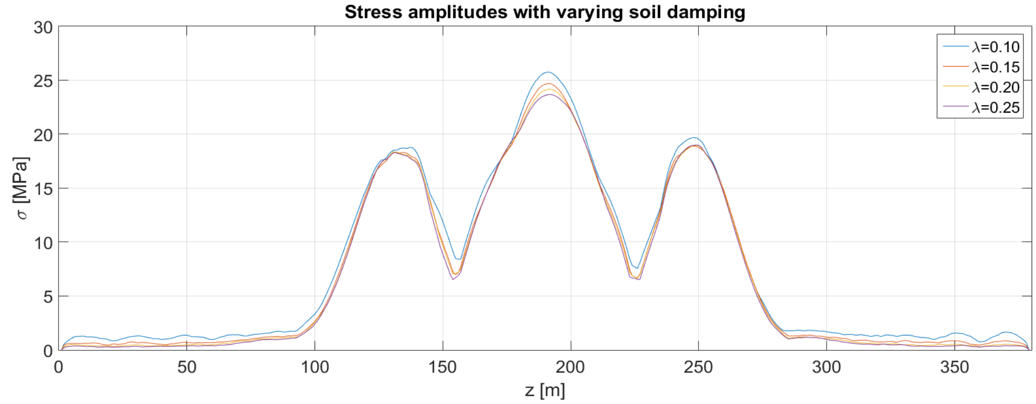


Figure 8.31: Stress amplitudes for different values of the vertical soil damping, from Ulveseter's nonlinear model

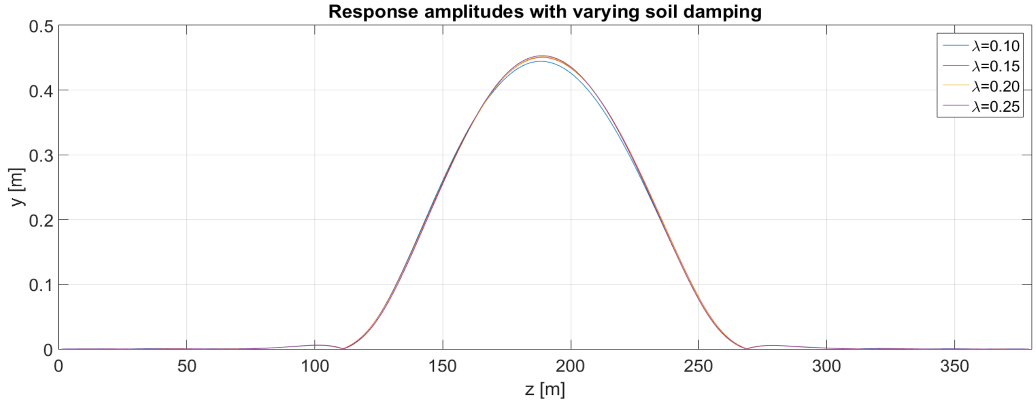


Figure 8.32: Response amplitudes for different vertical soil damping values, from Ulveseter's nonlinear model

**VIVANA**

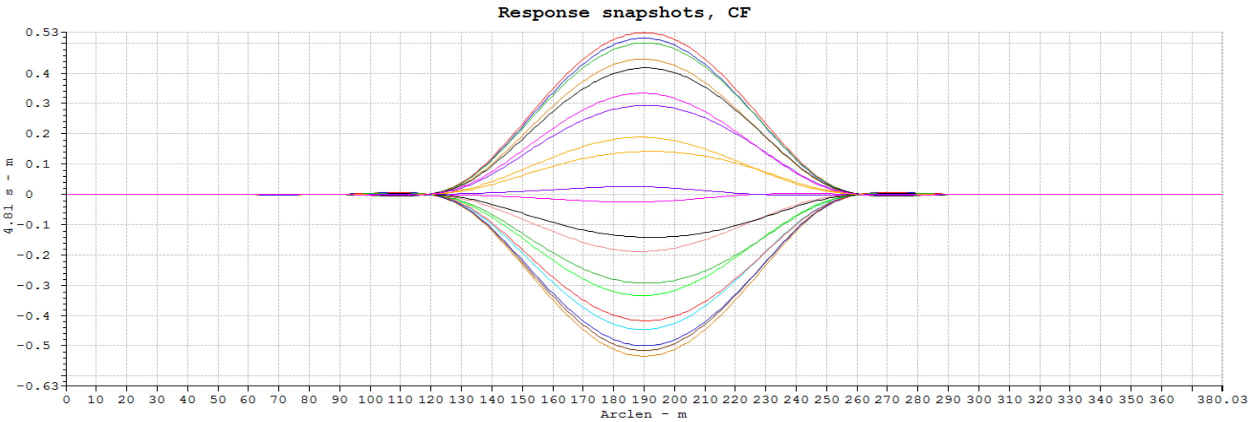


Figure 8.33: Snapshots of the cross-flow response, from VIVANA, for $\lambda = 0.10$

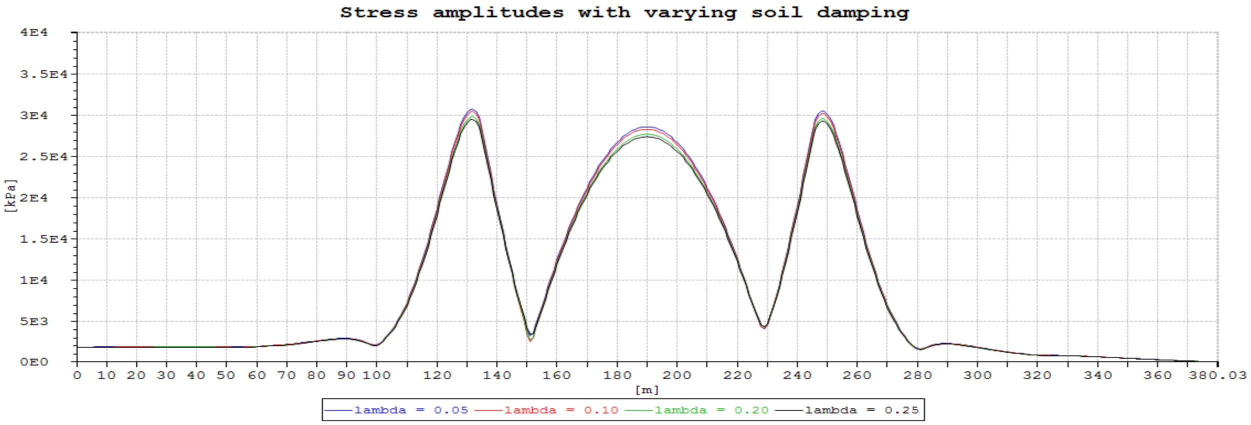


Figure 8.34: Stress amplitudes for different values of the vertical soil damping, from VIVANA

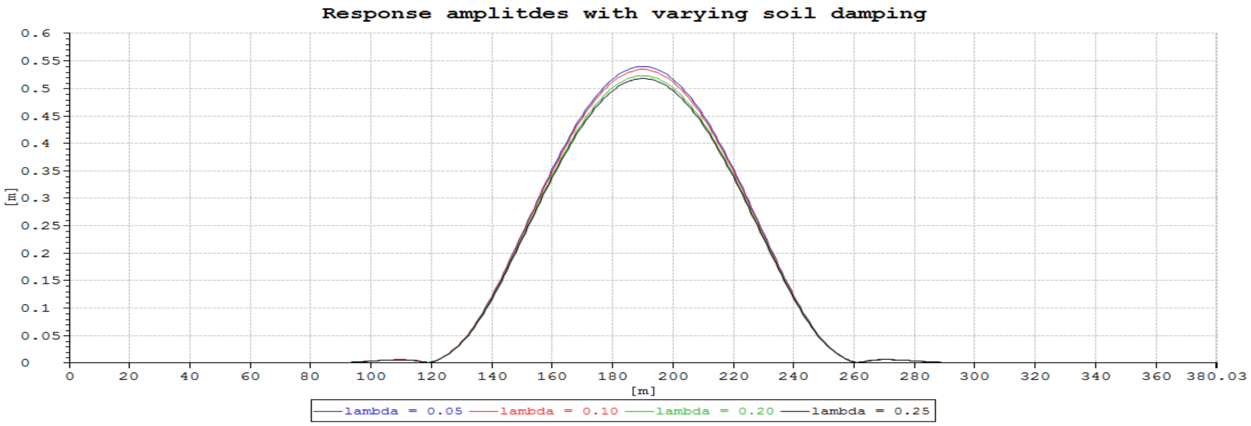


Figure 8.35: Response amplitudes for different values of the vertical soil damping, from VIVANA

### Comparison between the models

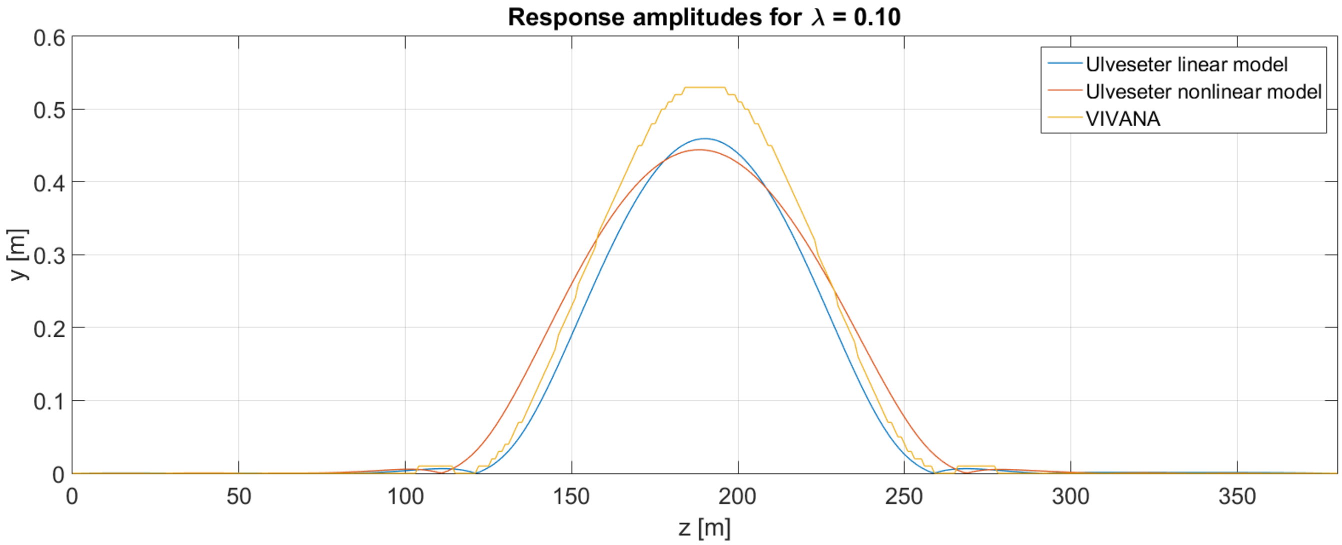


Figure 8.36: Comparison of maximum cross-flow response, for VIVANA and Ulveseter's models

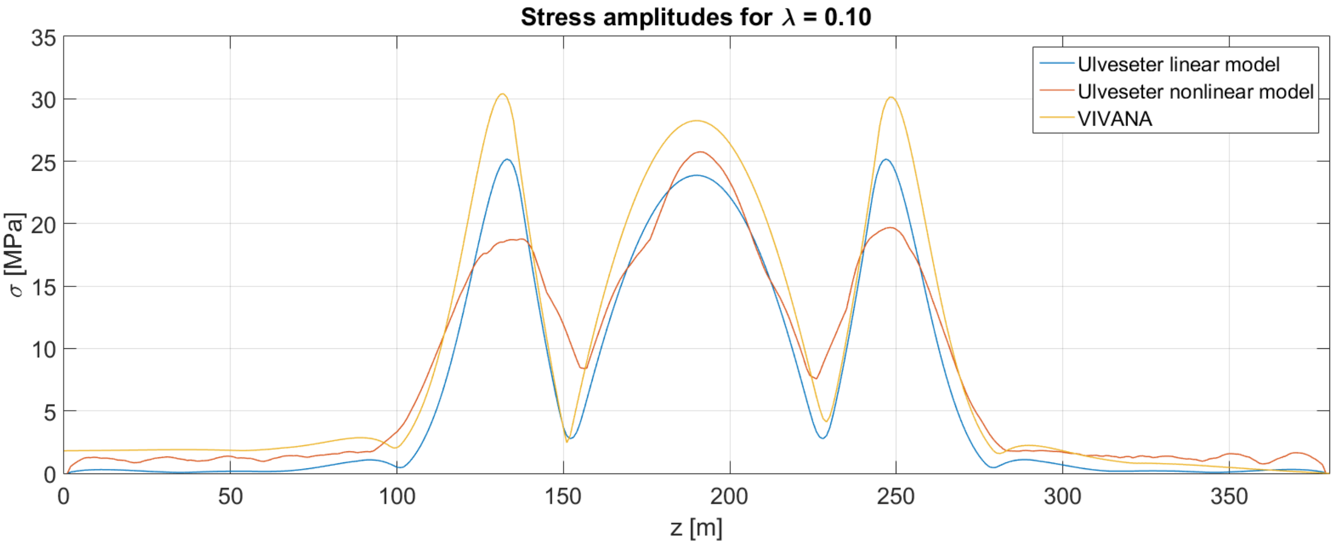


Figure 8.37: Comparison of maximum cross-flow stress amplitudes, for VIVANA and Ulveseter's models

### Observations

- in Figure 8.37, Ulveseter's nonlinear model gives a great reduction in stress amplitudes at the shoulders, compared to Ulveseter's linear model, but there is a lager stress peak at the middle of the free span.
- in Figure 8.30 the asymmetry about the z-axis of the response from Ulveseter's nonlinear model is clear, so the touch down point varies more than for Case 2.
- Response amplitudes, for both VIVANA and Ulveseter's model, are little affected by the soil damping, as was observed for Case 2 (Figure 8.28, 8.29, 8.31, 8.32, 8.34, 8.35). However, in this case Ulveseter's nonlinear model is more sensitive to soil damping, than Ulveseter's linear model, which is opposite from the observation in Case 1.

## 8.4 Case 4 - Realistic pipeline model with varying soil stiffness

In Case 4 we use the same pipeline model, and seabed profile as in Case 2. The difference is that the soil stiffness $k_s$ is varied to see how it influences the effect of soil damping and response behavior. Two new values of the soil stiffness have been tested. In Case 2 the soil stiffness was set to 40kN/m$^2$. Here, the soil stiffness values tested are 80kN/m$^2$ and 10.2kN/m$^2$. The first value is chosen to see the effect of a hard bottom. The last value is chosen based on the procedure outlined in Section 6.1.3, Equation 6.8, to look at the effect of a soft bottom.

Due to convergence problems in VIVANA, the soft bottom case is only tested for soil damping ratios of 0.15, 0,20 and 0.25. For the same reason, the hard bottom case is tested for soil damping ratios of 0.05 - 0.25. Ulveseter's linear model applies soil damping ratios in the range of 0 to 0.25, while Ulveseter's nonlinear model applies soil damping ratios of 0.10-0.25.

## Results

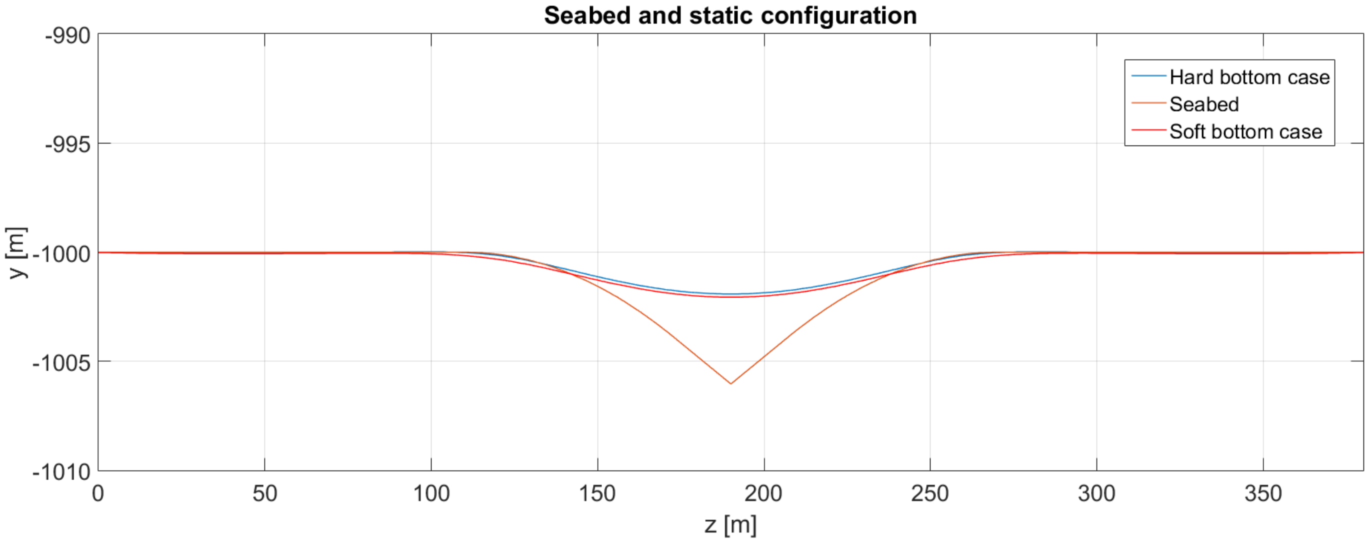


Figure 8.38: Static configuration relative to seabed: The hard bottom case ($k_s = 80\text{kN}/\text{m}^2$) and the soft bottom case ($k_s = 10.2\text{kN}/\text{m}^2$)

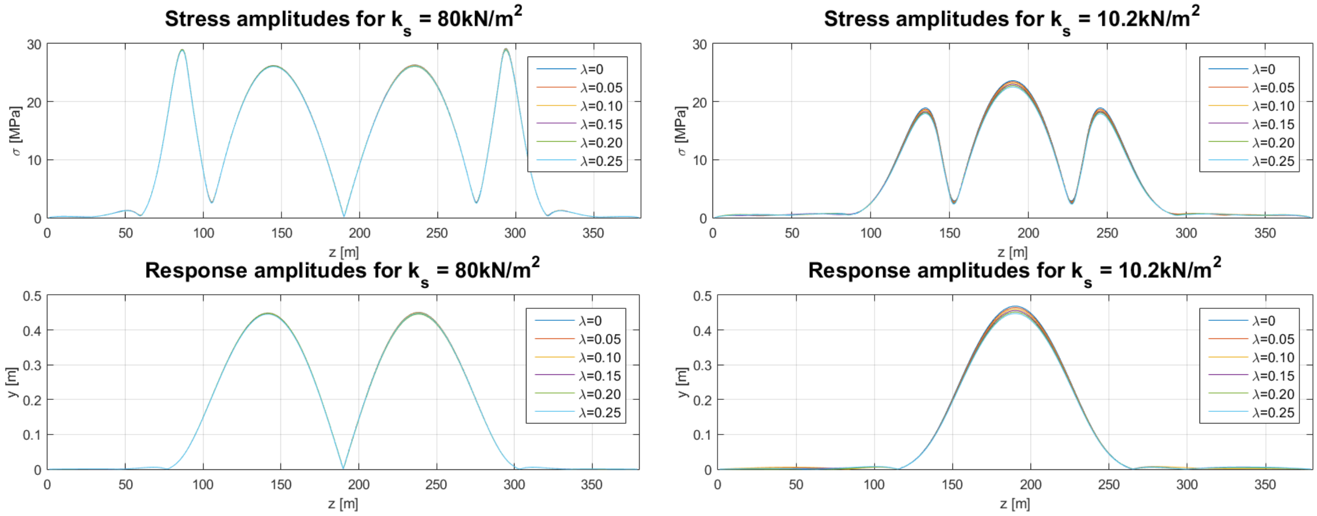


Figure 8.39: Results from Ulveseter's linear model: The hard bottom case ($k_s = 80\text{kN}/\text{m}^2$) and the soft bottom case ($k_s = 10.2\text{kN}/\text{m}^2$)

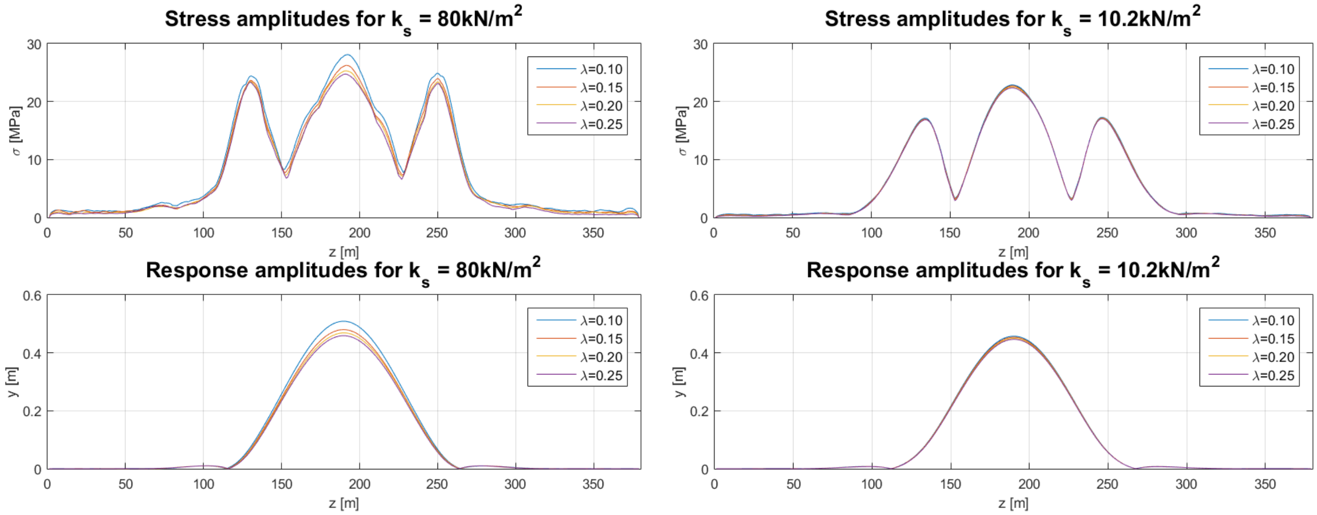


Figure 8.40: Results from Ulveseter's nonlinear model: The hard bottom case ($k_s$ = 80kN/m$^2$) and the soft bottom case ($k_s$ = 10.2kN/m$^2$)

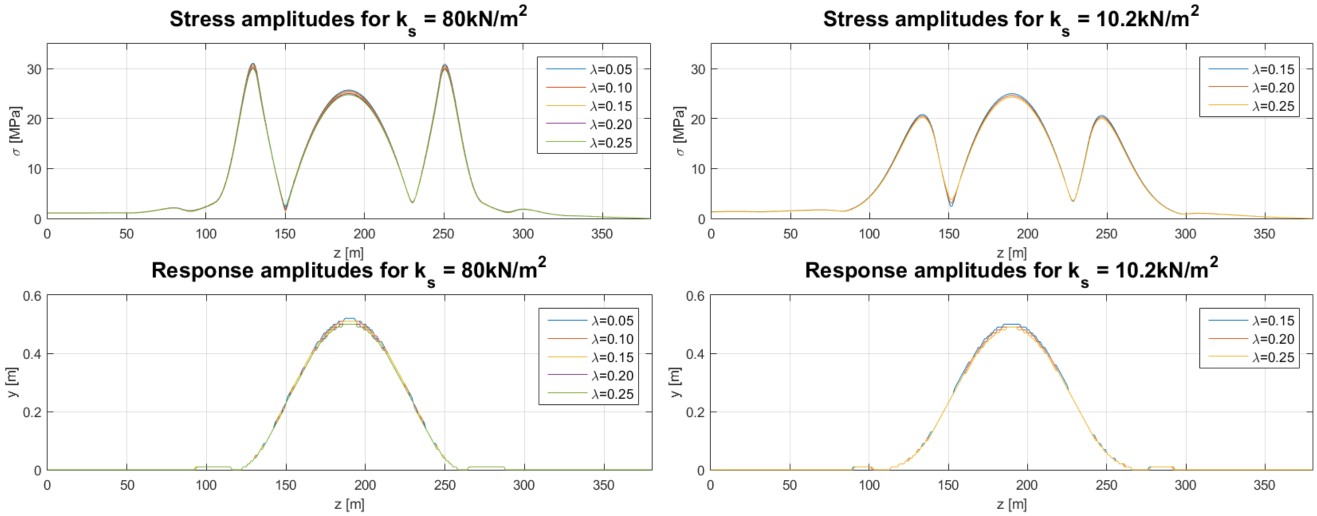


Figure 8.41: Results from VIVANA: The hard bottom case ($k_s$ = 80kN/m$^2$) and the soft bottom case ($k_s$ = 10.2kN/m$^2$)

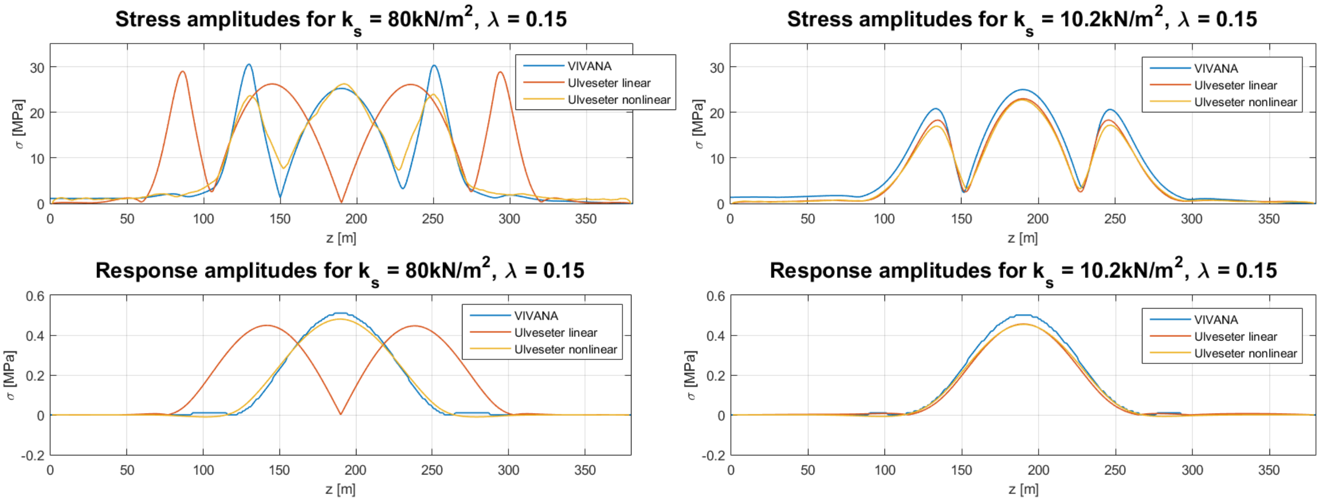


Figure 8.42: Results from all models with soil damping ratio 0.15: The hard bottom case ($k_s$ = 80kN/m$^2$) and the soft bottom case ($k_s$ = 10.2kN/m$^2$)

### Observations

- The static free span will be longer for the hard bottom case than the soft bottom case (Figure 8.38).
- Ulveseter's linear model predicts response oscillations of mode two (Figure 8.39), while Ulveseter's nonlinear model (Figure 8.40) and VIVANA (Figure 8.41) give mode one response.
- All results are little influenced by the soil damping. However, Ulveseter's nonlinear model has the largest variations due to soil damping, for the hard bottom case.
- All models give largest stresses for the hard bottom case.
- The difference between the models gets smaller when the soil stiffness decreases (Figure 8.42).

## 8.5 Case 5 - Realistic pipeline model with different current velocities

It is of interest to compare VIVANA to Ulveseter's model for different current velocities, to see how the response behaves. Case 5 is an investigation using the same data as applied in Case 2, only that the current velocity is varied, instead of the soil damping or the soil stiffness. Ulveseter's model and VIVANA are tested for a a vertical soil damping value of 10% of critical damping.

## Results

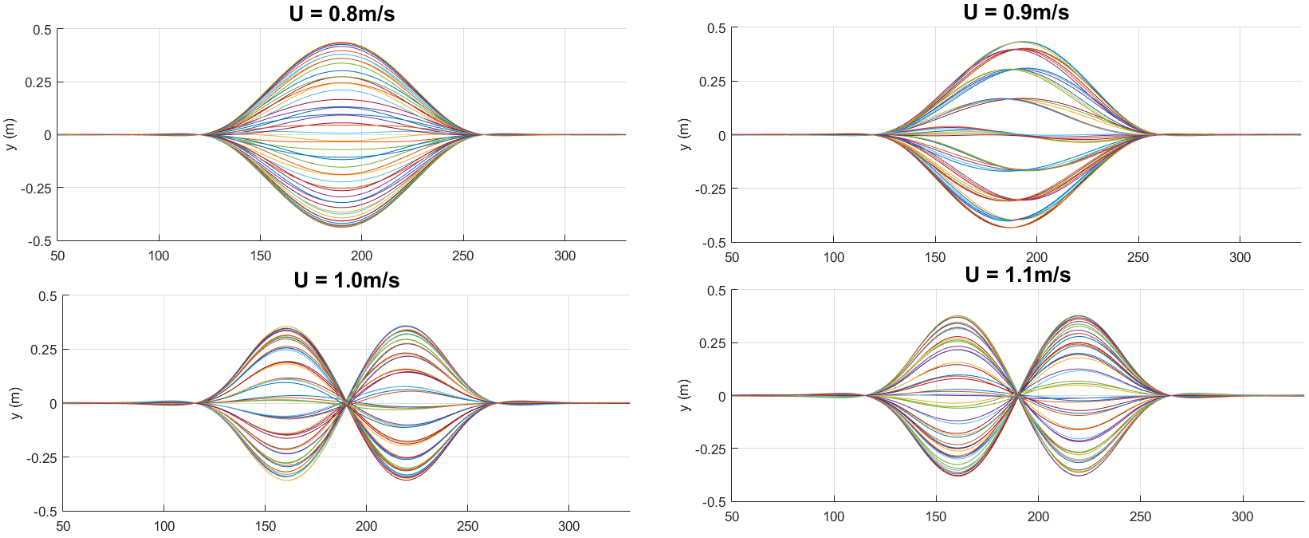


Figure 8.43: Snapshots of pipeline responses from Ulveseter's linear model, for current velocities 0.8m/s to 1.1m/s

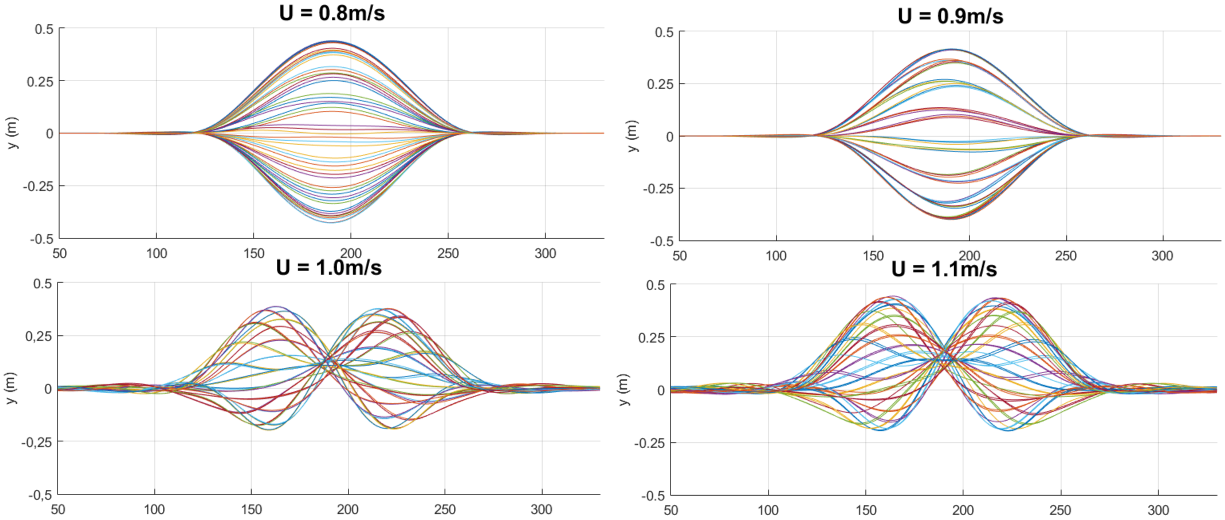


Figure 8.44: Snapshots of pipeline responses from Ulveseter's nonlinear model, for current velocities 0.8m/s to 1.1m/s

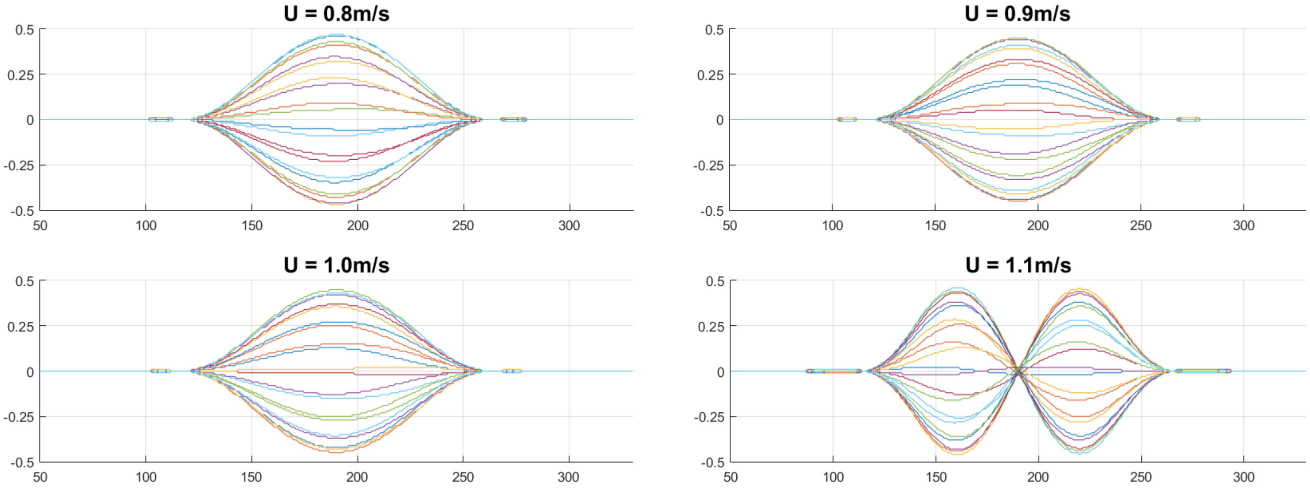


Figure 8.45: Snapshots of pipeline responses from VIVANA, for current velocities 0.8m/s to 1.1m/s

### Observations

- The change from the first mode to the second mode for Ulveseter' linear model takes place for a current velocity between 0.9m/s and 1m/s (Figure 8.43).
- For Ulveseter's nonlinear model, there is also a mode shift between a current velocity of 0.9m/s and 1m/s. However, the response picture becomes more chaotic, than for Ulveseter's linear model, at the mode shift (Figure 8.44).
- VIVANA predicts a mode shift from the first to second mode for a current velocity between 1.0m/s and 1.1m/s. Thus, in this case, VIVANA needs about 10% larger current velocity for the mode to change, than what was observed using Ulveseter's model (Figure 8.45).

## 8.6 Case 6 - Comparison to time domain analysis

In (Larsen et al., 2004) a VIV analysis using VIVANA and RIFLEX for a realistic pipeline model is presented. Both a nonlinear time domain solution combining RIFLEX and VIVANA as discussed in Section 6.2, and a traditional VIVANA frequency domain solution is presented and compared. Some of the results from the study will in the following be compared to Ulveseter's model. It is of special interest to see how Ulveseter's nonlinear model fits the results from the nonlinear time domain analysis using RIFLEX/VIVANA.

As one of the authors of (Larsen et al., 2004), Carl Martin Larsen has access to RIFLEX and VIVANA files that where used in the paper. Some of the files contain case studies that where not in the paper, but were part of the study. In these files comparisons between the traditional VIVANA analysis and the RIFLEX/VIVANA time domain analysis are done. It is one of these files Case 5 is based upon. The data for the case, as was applied by Larsen, is given in Table 8.3.

Table 8.3: Data for Case 5

| **Name** | **Symbol** | **Size** | **Dimension** |
|---|---|---|---|
| Length | L | 380 | $m$ |
| Diameter | D | 0.556 | $m$ |
| Bending stiffness | EI | $2.93*10^7$ | $Nm^2$ |
| Mass per unit length (air) | m | 314.86 | $kg/m$ |
| End tension | T | $450*10^3$ | $N$ |
| Current velocity | U | 0.7 | $m/s$ |
| Soil stiffness | $k_s$ | $40*10^3$ | $N/m^2$ |

## Results

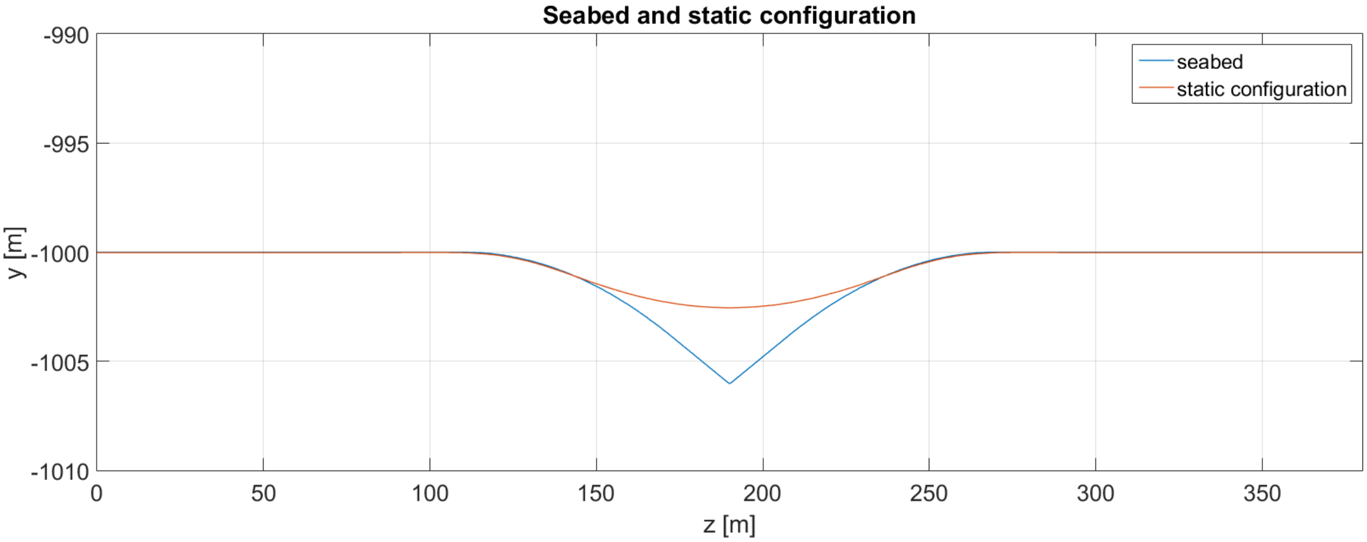


Figure 8.46: Seabed and static configuration for Case 4

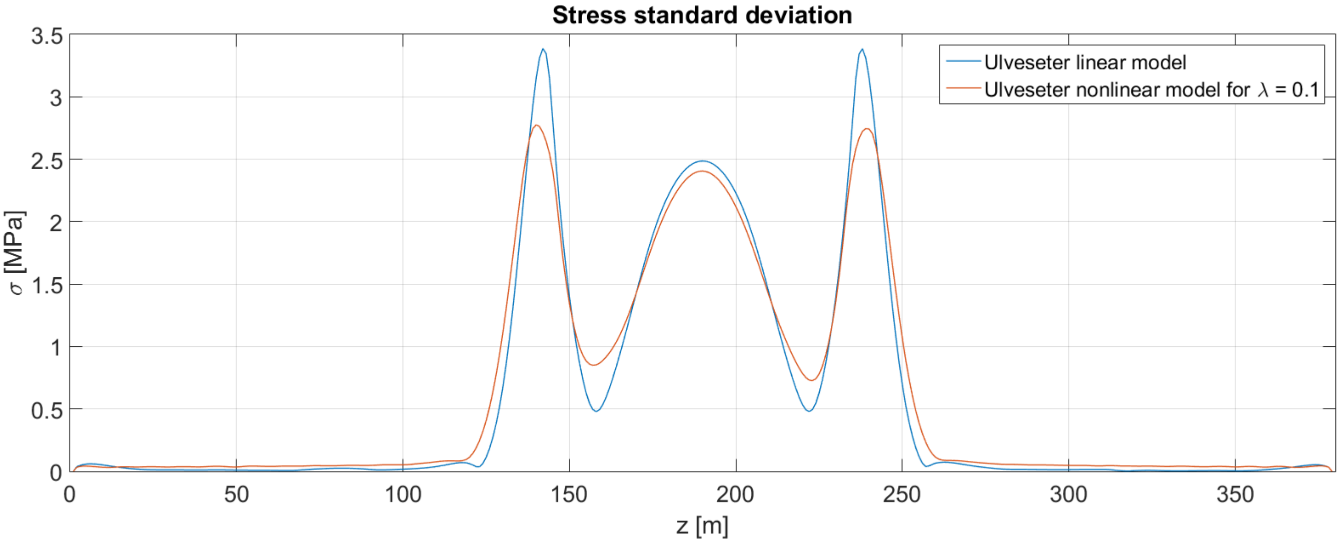


Figure 8.47: Comparison between Ulveseter's nonlinear and linear model

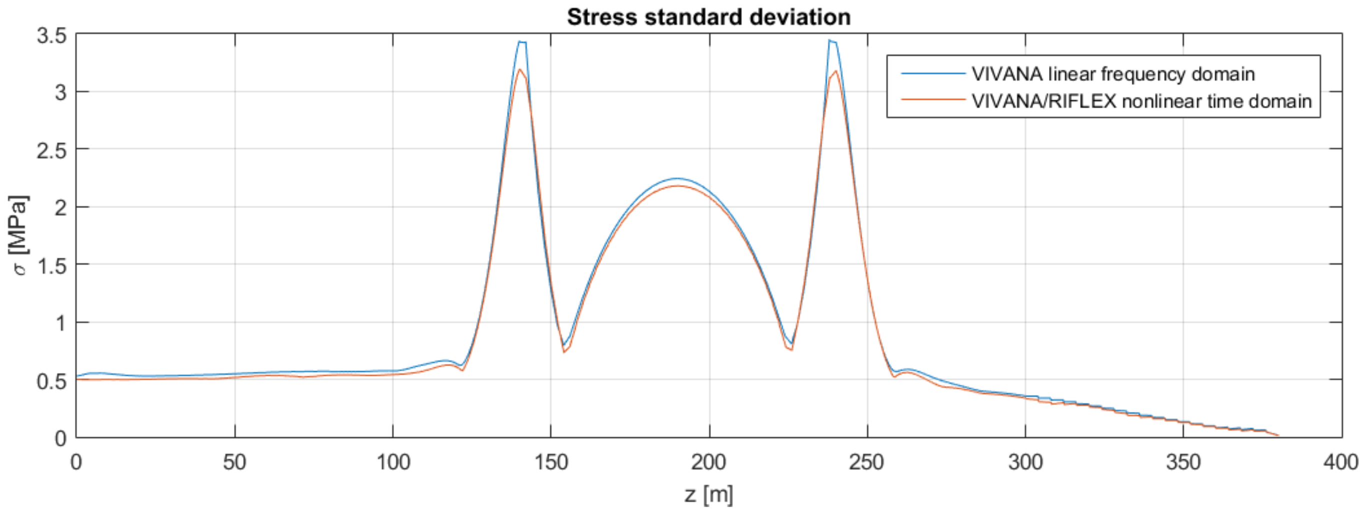


Figure 8.48: Comparison between traditional VIVANA analysis and VIVANA/RIFLEX time domain analysis

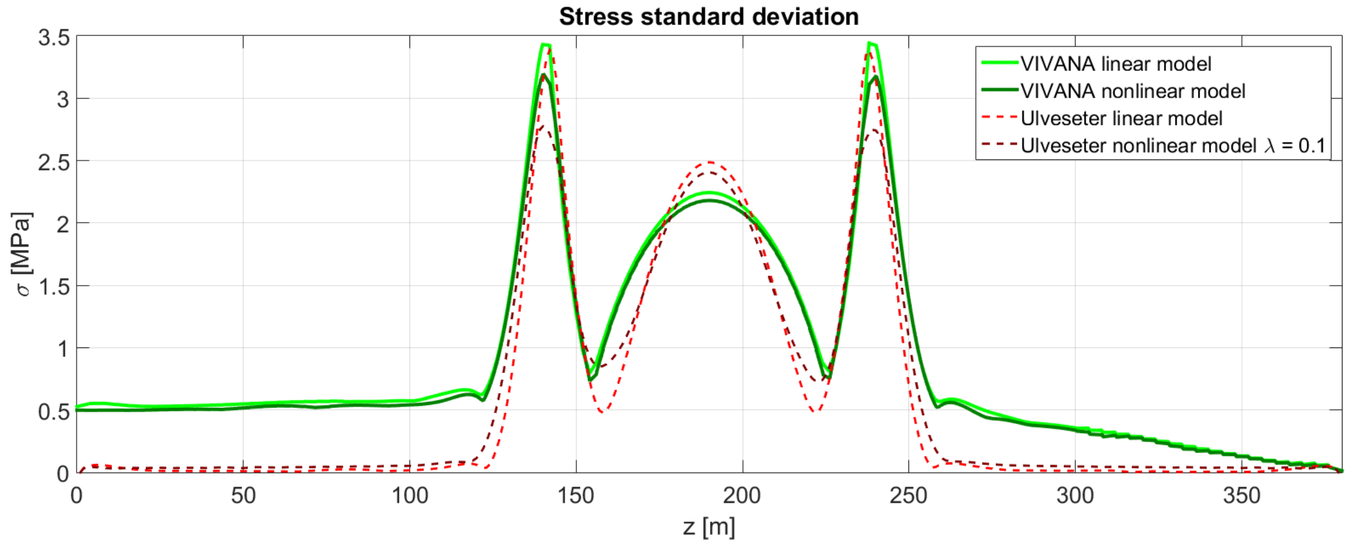


Figure 8.49: Comparison between Ulveseter's model and VIVANA/RIFLEX

## Observations

- Both Ulveseter's nonlinear model and RIFLEX/VIVANA nonlinear time domain model behaves very similarly, relative to the corresponding linear solutions (Figure 8.46 and 8.47). Both nonlinear models get a reduction of stress amplitude, significantly at the shoulders, and some at the middle of the free span. A difference between the two models is that Ulveseter's nonlinear model has a small stress reduction between the shoulder peak and the midspan peak, giving a larger stress value than Ulveseter's linear model. This is not seen in RIFLEX/VIVANA nonlinear model, where the stress curve more or less follows the linear frequency domain VIVANA solution.
- Looking at Figure 8.49, some differences regarding the shape of the curves, comparing Ulveseter's model and VIVANA, are observed.

# Chapter 9

# Discussion

## 9.1 Discussion of results

The six case studies give a good description of Ulveseter's model, and how it compares to VIVANA. Many of the findings are explainable using physical understanding of the nonlinear soil-pipe interaction mechanism and dynamic behavior of pipes.

For all cases comparing stress amplitudes from Ulveseter's model and VIVANA, we see a reduction of stress amplitudes at the pipe shoulders, for the nonlinear model. This, even though the nonlinear model may predict larger response amplitudes, as for Case 1. This can be explained knowing that the bending stress is a function of curvature. The nonlinear soil-pipe interaction allows the pipeline to lift up from the seabed at the pipe shoulders. This results in a curvature reduction around the touch down points, so that the accumulation of fatigue damage at the shoulders is reduced for Ulveseter's nonlinear model.

The seabed profile affects the behavior of Ulveseter's nonlinear model. The case with the smallest curvature at the pipe shoulders, Case 1, has a very asymmetric response shape. In other words, the positions of the touch down points vary easily with the pipeline response, which is as physically expected for a relatively flat seabed. For Case 2, a steep valley is used as the sea bottom profile. This results in the response being almost symmetric about the longitudinal axis. The touch down points varies more for Case 3, where a less steep valley is used as seabed.

In Case 4, the soil stiffness is varied. The result is that the hard bottom case will induce response mode two for Ulveseter's linear model. Ulveseter's nonlinear model will oscillate with mode one. For the soft bottom case all VIV models give response of first mode. The reason for the mode shift in Ulveseter's linear model is not trivial. A larger soil stiffness will add to the total stiffness matrix, increasing the eigenfrequency. Thus a larger load frequency is need to excite mode two. However, increased soil stiffness will make the free span longer. Hence there are fewer soil stiffness terms that contribute to the total stiffness compared to the soft bottom case. The total effect is that the total stiffness is smaller in the hard bottom case giving smaller eigenfrequencies. From Figure 2.5, we know that the maximum response for cross-flow VIV is typically when the reduced velocity is between 6 and 8. Since Thorsen's VIV model is based on empirical coefficients, the model should be capable of capturing this experimental observation. The soft bottom case is compared against the hard bottom case for Ulveseter's linear model in Table 9.1. We see that the hard bottom case will have a reduced velocity of 6.5 for the second mode, while the soft bottom case will have a reduced velocity of 6.9 for the first mode.

Table 9.1: Case 4: Eigenfrequencies and reduced velocity for different mode shapes

| **Ulveseter's linear model** | | | | |
|---|---|---|---|---|
| Mode | Hard bottom case | | Soft bottom case | |
| 1 | $f_1 = 0.09s^{-1}$ | $U_r = 14.7$ | $f_1 = 0.18s^{-1}$ | $U_r = 6.9$ |
| 2 | $f_2 = 0.20s^{-1}$ | $U_r = 6.5$ | $f_2 = 0.42s^{-1}$ | $U_r = 3.0$ |
| 3 | $f_3 = 0.34s^{-1}$ | $U_r = 3.7$ | $f_3 = 0.65s^{-1}$ | $U_r = 2.0$ |

Another interesting finding is how the response shape changes mode with increasing current velocity. Physically, increasing velocity will increase the vortex shedding frequency. At one current velocity, the vortex shedding frequency will be so large that instead of exciting the eigenfrequency corresponding to the first mode shape, it will excite the eigenfrequency of the second mode shape. Thus, we get a mode shift. For Case 5 we see that the current velocity giving mode shift is different from VIVANA and Ulveseter's model. This is probably due to the different added mass models applied in VIVANA and Thorsen's force model.

The nonlinear time domain analysis applying RIFLEX and VIVANA compares well to Ulveseter's model. However, the shape of the stress amplitude curves is a little different. This can be a result of axial stresses. Ulveseter's model is limited to vertical and rotation dofs. Applied to a case were bending stresses are small, axial stresses may not be negligible, affecting the results.

## Soil damping

When it comes to the influence of soil damping, it is only for Case 1 we can conclude that the soil damping effect is of significant magnitude for all models. There can be several reasons why the other cases were less dependent on the soil damping. In general, it looks like the influence of soil damping is a function of the pipeline properties, and the hydrodynamic properties. Case 1 deals with a shorter pipe, with less bending stiffness, with less end tension and with a stronger current, than the other cases. Also, the seabed profile applied for Case 1 has a smaller curvature at the pipe shoulders than the rest of the cases. However the seabed profile do not significantly change the influence of soil damping alone, as investigated for Case 3.

From Case 3 and 4, Ulveseter's nonlinear is more sensitive to soil damping than Ulveseter's linear model and VIVANA. Especially for small soil damping values. Comparing $\lambda = 0.10$ to $\lambda = 0.15$

in Figure 8.31 and 8.40, the stress amplitudes are seen to change significantly. This might be a result of convergence problems in Ulveseter's nonlinear model. From test cases, the model is not capable of finding a steady-state solution if the soil stiffness is too high, and the soil damping too low. This is why the nonlinear model is only applied to cases where the soil damping is larger or equal to 10% of the estimated critical damping. When the case studies were performed, the time response at a node in the shoulder region was plotted for every case study. This was done to try to ensure that a steady-state solution was found. However, it might be that Case 3 and 4 are on the limit of reaching steady-state, and hence the stress amplitudes are affected by this.

## 9.2 Discussion of current and pipeline properties

Concerning the structural properties of the pipeline model in Case 1, the tension in the pipe is low compared to the bending stiffness and dimensions of the pipeline. Professor Svein Sævik at the Department of Marine Technology at NTNU has more than 10 years experience with pipeline engineering. From his experience, a 14-inch pipeline (≈0.36m) should have a tension of the order of magnitude 100kN. This is a factor of 10 more than what was applied in Case 1. The structural characteristics of the other pipeline cases are based on the realistic model in (Larsen et al., 2004). Here the tension is higher, and the structural dimensions fits better with Sævik's experience.

(Yttervik, 2004) describes current conditions at locations of special interest for the oil and gas industry. One of these places is the Norwegian Sea, where a current called Continental Slope Current passes through west of Shetland and Norway. The mean flow velocity is approximately 0.2m/s, but peak values of more than 1m/s have been observed. In the Atlantic Ocean, West of Africa, oil and gas fields such as Girassol, Bonga and Kizomba are present. The Guinea Current outside the coast of West Africa can have current velocities up to 1m/s (Richardson and Reverdin, 1987).

In the case studies performed in Chapter 8, a velocity range of 0.7m/s to 1.2m/s is considered. From the discussion above, these are extreme current conditions. At least for Case 1, where the flow velocity is 1.2m/s. Realistically, the current will be affected by the boundary layer induced by the flow-seabed interaction. Thus a uniform inflow, as analyzed in the present study, is not accurate. However, since there are large uncertainties and variations in current flow conditions, extreme vales are chosen to make conservative estimates.

# Chapter 10

# Summary and Recommendations for Further Work

## 10.1 Summary and Conclusion

In this Master Thesis, several aspects of VIV have been addressed. Chapter 2 gave a brief introduction to VIV as a phenomenon starting with hydrodynamics around a fixed circular cylinder in current. The modeling procedures, looking at CFD as a VIV prediction tool and empirical VIV models, were presented. Then, examples on how to perform VIV experiments were discussed, focusing on experimental results and how they can be applied in empirical VIV models. Chapter 3 was a presentation of VIVANA, which is the only frequency domain VIV model that has been studied in detail.

The main part of the present work is related to time domain VIV models, and how to apply them on free spanning pipelines. Chapter 4,5 and 6 handled these topics. In Chapter 4, several time domain VIV models were presented. In Chapter 5, Thorsen's VIV model was explained in detail. Chapter 6 gave a discussion on free spanning pipelines subjected to VIV, how to account for and quantify soil-pipe interaction, and examples on how to perform time domain VIV analyses. The first two objectives are related to literature study and presentation of time domain VIV, nonlinear FEM as applied for free spanning pipelines, and soil-pipe interaction. These objectives were answered in these three chapters.

My contribution to the field is presented in Chapter 7 and 8. Based on Thorsen's time domain VIV model, a nonlinear analysis tool, referred to as Ulveseter's model, was programmed using MATLAB. Ulveseter's model gives the option of a linear analysis and a nonlinear analysis. This is explained in Chapter 7, and both options are used in the case studies in Chapter 8. Ulveseter's nonlinear model takes into account the nonlinear soil-pipe interaction. This is done by updating the stiffness and damping matrices for every time step of the time integration, by checking if the translations dofs are in contact with the seabed. If there is contact, a soil damping term and a soil stiffness term is added to the associated matrices. If there is no contact, no soil damping term or soil stiffness term is added.

The goal was to make a simple MATLAB program looking at the effect of nonlinear soil-pipe interaction. As discussed in Chapter 7 are there many simplifications in Ulveseter's nonlinear model. However, the program is capable of predicting physically meaningful results, and compares well to Ulveseter's linear model and VIVANA. Also, convergence was tested to make sure that the dynamic equilibrium equation was solved correctly. Thus we conclude that the third objective is achieved.

The last objective was to compare Ulveseter model with other frequency domain and time domain solutions. The study is presented in Chapter 8. The effect of pipeline characteristics, seabed profile, soil stiffness, soil damping and current velocity was investigated. VIVANA was used as the frequency domain model. One comparison to another nonlinear time domain predictions was performed. This accomplishes the last objective.

The case study shows that:

- The soil damping has little influence on the overall response. It is only for Case 1 that the soil damping is of importance for all models. As discussed in Chapter 9, the other cases are more realistic both concerning structural properties of the pipeline, and the current velocity. In Case 3 and 4, Ulveseter's nonlinear model is more affected by the soil damping than the linear prediction tools. However, this might be a result of transient effects.
- Ulveseter's model compares good to VIVANA, but the general trend is that VIVANA predicts larger response and stress amplitudes.
- Physically meaningful results are observed. For Ulveseter's nonlinear model we have that the stress amplitudes at pipe shoulders are reduced compared to the linear models, the seabed profile affects the variation of touch down points, increased current velocity induces a mode shift of the response, and increasing soil stiffness increases stress amplitudes.

## 10.2 Recommendations for Further Work

Results from nonlinear VIV predictions are limited in the literature, so there are many possibilities for further work. More case studies can be performed, increasing the understanding of the soil damping effect on pipeline response. In this thesis, three different pipeline models are used for case studies. An extension can be to look at other pipelines with different structural properties. It is also possible to test for new seabed configurations, nonuniform current, variations in free span length, etc.

A more natural extension might be to improve the developed program, referred to as Ulveseter's model, to improve the quality of the results. Limitations of the model have been addressed i Section 7.2. A major limitation is that the static configuration of the pipeline is found from RIFLEX. This makes it impossible to perform a fully nonlinear analysis where all forces, both dynamic and static, are included at the same time. To spend more time on developing a static solution procedure in Ulveseter's model can therefore be a good improvement of the present work. Also, an extension to 3D is possible by implementing Thorsen's in-line model to Ulveseter's model, which again makes it possible to perform new case studies.

Thorsen's VIV model can also be applied to other slender marine structures than pipelines. VIV analyses on risers, especially drilling risers, are by the industry considered to yield too conservative results. Combining Thorsen's VIV model with a more advanced nonlinear structural model, using for example RIFLEX or SIMLA, makes it possible to investigate this issue. We can for example consider drill string geometry stiffness, inertia and damping forces from mud interaction. The plan is to investigate this in my PhD study.

# Appendix A

# MATLAB program

In this Appendix, the MATLAB-functions used in Ulveseter's model, are listed.

## A.1 analysis

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                                  %
% code name:                       %
% ULVESETER'S MODEL                %
%                                  %
% Based on Thorsen's VIV model     %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

clear all
close all
clc

%% Input options %%
%%%%%%%%%%%%%%%%%%%
text1 = 'Choose Case 1,2,3,4,5,6 or 7: ';
text2 = 'Choose linear analysis (1) or nonlinear analysis (2): ';
Case = input(text1);
type = input(text2);

%% Input parameters %%
%%%%%%%%%%%%%%%%%%%%%%
[D,L,EI,E,T,U,mdry,rho_sw,Ca,alpha1,alpha2,k_s,c_s,n,g,Tsim,h,m,l,n_dof,...
n_dof_red,n_nodes,transdofs,gamma,beta] = input_parameters(Case);

%% Inputs from RIFLEX %%
%%%%%%%%%%%%%%%%%%%%%%%%
[seabed_VIVANA, static_config, free_span_length,sea_bottom,static_gap] = ...
RIFLEX_input(Case,l,D,n_nodes,L);

```

```
%% Establishing FE model %%
%%%%%%%%%%%%%%%%%%%%%%%%%%%
[Ksys,Csys,Msys,Ksys_initial,Csys_initial] = ...
FE_model(EI,l,T,m,n,n_dof,alpha1,alpha2);

%% Eigenfrequencies %%
%%%%%%%%%%%%%%%%%%%%%%
MATLAB_f0_initial = (1/(2*pi)) * sqrt(eig(Ksys,Msys));

%% LINEAR ANALYSIS BEGINS %%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%
if type == 1

    %linear analysis
    [y,ytrans,dy,dytrans,ddy,ddytrans,phi_exc,Qy,Qytrans,time,Ay0,Ay1,...
    dymax,ddymax,N,MATLAB_f0] = linear_analysis(Ksys,Msys,Csys,k_s,c_s,...
    sea_bottom,transdofs,gamma,beta,h,rho_sw,D,l,Tsim,U,static_gap);

%% NONLINEAR ANALYSIS BEGINS %%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
elseif type == 2

    %nonlinear analysis
    [y,ytrans,dy,dytrans,ddy,ddytrans,phi_exc,Qy,Qytrans,time,Ay0,Ay1,...
    dymax,ddymax,N] = nonlinear_analysis(Ksys_initial,...
    Csys_initial,Ksys,Msys,Csys,k_s,c_s,sea_bottom,transdofs,gamma,beta,h,...
    rho_sw,D,l,Tsim,U,static_gap,static_config,mdry,seabed_VIVANA,n,g);

end %if type

%% Post-processing %
%%%%%%%%%%%%%%%%%%%%
[stress_amp, stress_rms,stress_std,fp] = POSTproc(ytrans,l,L,D,E,N,n,h,...
Tsim,c_s,k_s,type,sea_bottom,seabed_VIVANA,static_config,Case);
```

## A.2 input_parameters

```
%input parameters

function [D,L,EI,E,T,U,mdry,rho_sw,Ca,alpha1,alpha2,k_s,c_s,n,g,Tsim,h,m,...
l,n_dof,n_dof_red,n_nodes,transdofs,gamma,beta] = input_parameters(Case)


if Case == 1 %Pipeline data for same model as used in my Project Thesis

%Physical:
A_int    = 0.1018;                  %[m^2]      internal area
A_ext    = 0.1256;                  %[m^2]      total area
D        = 2 * sqrt(A_ext/pi);      %[m]        external diameter
L        = 180;                     %[m]        model length
EI       = 8.9*10^7;                %[Nm^2]     bending stiffness
E        = 207*10^9;                %[N/m^2]    E-module (steel, used in VIVANA)
T        = 10*10^3;                 %[N]        tension (STAMOD CALLAS)
U        = 1.2;                     %[m/s]      incoming current velocity
mdry     = 217.96;                  %[kg/m]     mass per unit length, dry
rho_sw   = 1027;                    %[kg/m^3]   density of water
Ca       = 1.0;                     %[-]        added mass coefficient
alpha1   = 0;                       %           proportional damping factor 1
alpha2   = 0;                       %           proportional damping factor 2
k_s      = 17460.105;               %[N/m^2]    vertical soil_stiffness per meter
c_s      = 492.25;                  %[Ns/m^2]   vertical soil damping per meter
n        = 180;                     %[-]        number of elements

elseif Case == 2 || Case == 3 %% model used in omae2004

%Physical:
D        = 0.55;                    %[m]        external diameter
L        = 380;                     %[m]        model length
EI       = 2.9*10^8;                %[Nm^2]     bending stiffness
E        = 208*10^9;                %[N/m^2]    E-module (steel)
T        = 450*10^3;                %[N]        tension
U        = 0.7;                     %[m/s]      incoming current velocity
mdry     = 315;                     %[kg/m]     mass per unit length, dry
rho_sw   = 1025;                    %[kg/m^3]   density of water
Ca       = 1.0;                     %[-]        added mass coefficient
alpha1   = 0;                       %           proportional damping factor 1
alpha2   = 0;                       %           proportional damping factor 2
k_s      = 40*10^3;                 %[N/m^2]    vertical soil_stiffness per meter
c_s      = 945.32;                  %[Ns/m^2]   vertical soil damping per meter
n        = 380;                     %[-]        number of elements

elseif Case == 4 %% model used for comparison to time domain analysis
%Physical:
D        = 0.556;                   %[m]        external diameter
L        = 380;                     %[m]        model length
EI       = 2.93*10^7;               %[Nm^2]     bending stiffness
E        = 20.6*10^9;               %[N/m^2]    E-module (steel)
```

```
T       = 450*10^3;             %[N]        tension
U       = 0.7;                  %[m/s]      incoming current velocity
mdry    = 315;                  %[kg/m]     mass per unit length, dry
rho_sw  = 1025;                 %[kg/m^3]   density of water
Ca      = 1.0;                  %[-]        added mass coefficient
alpha1  = 0;                    %           proportional damping factor 1
alpha2  = 0;                    %           proportional damping factor 2
k_s     = 40*10^3;              %[N/m^2]    vertical soil_stiffness per meter
c_s     = 954.11;               %[Ns/m^2]   vertical soil damping per meter
n       = 380;                  %[-]        number of elements

elseif Case == 5 %% model used in omae2004, with varying current speed

%Physical:
D       = 0.55;                 %[m]        external diameter
L       = 380;                  %[m]        model length
EI      = 2.9*10^8;             %[Nm^2]     bending stiffness
E       = 208*10^9;             %[N/m^2]    E-module (steel)
T       = 450*10^3;             %[N]        tension
U       = 1.1;                  %[m/s]      incoming current velocity
mdry    = 315;                  %[kg/m]     mass per unit length, dry
rho_sw  = 1025;                 %[kg/m^3]   density of water
Ca      = 1.0;                  %[-]        added mass coefficient
alpha1  = 0;                    %           proportional damping factor 1
alpha2  = 0;                    %           proportional damping factor 2
k_s     = 40*10^3;              %[N/m^2]    vertical soil_stiffness per meter
c_s     = 945.32;               %[Ns/m^2]   vertical soil damping per meter
n       = 380;                  %[-]        number of elements

elseif Case == 6 %% model used in omae2004, hard bottom case

%Physical:
D       = 0.55;                 %[m]        external diameter
L       = 380;                  %[m]        model length
EI      = 2.9*10^8;             %[Nm^2]     bending stiffness
E       = 208*10^9;             %[N/m^2]    E-module (steel)
T       = 450*10^3;             %[N]        tension
U       = 0.7;                  %[m/s]      incoming current velocity
mdry    = 315;                  %[kg/m]     mass per unit length, dry
rho_sw  = 1025;                 %[kg/m^3]   density of water
Ca      = 1.0;                  %[-]        added mass coefficient
alpha1  = 0;                    %           proportional damping factor 1
alpha2  = 0;                    %           proportional damping factor 2
k_s     = 80*10^3;              %[N/m^2]    vertical soil_stiffness per meter
c_s     = 945.32;               %[Ns/m^2]   vertical soil damping per meter
n       = 380;                  %[-]        number of elements

elseif Case == 7 %% model used in omae2004, soft bottom case

%Physical:
D       = 0.55;                 %[m]        external diameter
L       = 380;                  %[m]        model length
EI      = 2.9*10^8;             %[Nm^2]     bending stiffness
E       = 208*10^9;             %[N/m^2]    E-module (steel)
```

```
T       = 450*10^3;             %[N]        tension
U       = 0.7;                  %[m/s]      incoming current velocity
mdry    = 315;                  %[kg/m]     mass per unit length, dry
rho_sw  = 1025;                 %[kg/m^3]   density of water
Ca      = 1.0;                  %[-]        added mass coefficient
alpha1  = 0;                    %           proportional damping factor 1
alpha2  = 0;                    %           proportional damping factor 2
k_s     = 10199.187;            %[N/m^2]    vertical soil_stiffness per meter
c_s     = 945.32;               %[Ns/m^2]   vertical soil damping per meter
n       = 380;                  %[-]        number of elements
end
g       = 9.81;                 %[m/s^2]    acceleration of gravity
%% Numerical input

Tsim    = 150;      % simulation time
h       = 0.01;     % time step size
%n        = 180;     % number of elements
gamma   = 0.5;      % Newmark parameter
beta    = 0.25;     % Newmark parameter

m           = mdry+Ca*rho_sw*pi/4*D^2;  %mass + added mass per unit length
l           = L/n;                      %length of each element
n_dof       = (n+1)*2;                  %mumber of dofs
n_dof_red   = n_dof - 2;                %reduced number of dofs after BCs
n_nodes     = n + 1;                    %number of nodes
transdofs   = [2:2:2*n-1];              %id translation dofs, reduced set
```

## A.3 RIFLEX_input

```
%all inputs from RIFLEX for simplified nonlinear time domain analysis
function [seabed_VIVANA, static_config, ...
    free_span_length,sea_bottom,static_gap]...
= RIFLEX_input(Case,l,D,n_nodes,L)

%static displacemenet and seabed coordiantes (from RIFLEX/VIVANA)
[seabed_VIVANA, static_config, free_span_length] = read_input(Case);

%for square seabed and no static displacement (linear case)
[sea_bottom] = seabed(l,n_nodes,L,free_span_length,D);

%static gap as given by RIFLEX time domain analysis
[static_gap] = staticGap(static_config,seabed_VIVANA);

end
```

## A.4 read_input

```
function [seabed, static_config, free_span_length] = read_input(Case)

%load static configuration from RIFLEX-files
if Case == 1
    static_config = load('project_thesis\static_XZ_configuration.txt');
elseif Case == 2
    static_config = load('omae2004\static_XZ_configuration_omae04.txt');
elseif Case == 3
    static_config = load('Case3\static_XZ_configuration_case3.txt');
elseif Case == 4
    static_config = load('Case4\static_XZ_configuration_case4.txt');
elseif Case == 5
    static_config = load('Case5\static_XZ_configuration_case5.txt');
elseif Case == 6
    static_config = load('Case6\static_XZ_configuration_case6.txt');
elseif Case == 7
    static_config = load('Case7\static_XZ_configuration_case7.txt');
end

%load 3D-bottom from RIFLEX
if Case == 1
    seabed = load('project_thesis\seabed_XZ_configuration.txt');
elseif Case == 2
    seabed = load('omae2004\seabed_XZ_configuration_oma04.txt');
elseif Case == 3
    seabed = load('Case3\seabed_XZ_configuration_case3.txt');
elseif Case == 4
    seabed = load('Case4\seabed_XZ_configuration_case4.txt');
elseif Case == 5
    seabed = load('Case5\seabed_XZ_configuration_case5.txt');
elseif Case == 6
    seabed = load('Case6\seabed_XZ_configuration_case6.txt');
elseif Case == 7
    seabed = load('Case7\seabed_XZ_configuration_case7.txt');
end

if static_config(1,1) == seabed(1,1)

    if seabed(1,1) ≠ 0

        %starting both seabed and static configuration on z = 0
        static_config(:,1)  = static_config(:,1) - static_config(1,1);
        seabed(:,1)         = seabed(:,1) - seabed(1,1);

    end

else

    disp('static configuration and seabed configuration do not match');

```

```
end
%% %finding the length of the free-span considering the static configuration
%and seabed as used in RIFLEX

res2 = [];
for i = 1 : length(seabed(:,1))

    if static_config(i,2) > seabed(i,2)

        res = seabed(i,1);
        res2 = [res2; res];

    end

end

free_span_length = max(res2) - min(res2);

end
```

## A.5 seabed

```
%function producing a square seabed
function [sea_bottom] = seabed(elem_length,num_nodes,model_length, ...
free_span_length,D)

%depth of seabottom relative to the pipe
Depth = - 5*D;

%defining coordiantes of sea-bottom (end nodes not included)
sea_bottom = zeros(num_nodes-2,2);

%z-coordinates to the seabottom
sea_bottom(:,2) = [elem_length : elem_length : model_length - elem_length];

%% coordinates matching the free_span_length

for k = 1:num_nodes - 2

    if sea_bottom(k,2) < 0.5 * (model_length - free_span_length) ||...
       sea_bottom(k,2) > 0.5 * (model_length + free_span_length)

        sea_bottom(k,1) = 0;

    else

        sea_bottom(k,1) = Depth;

    end

end


end
```

## A.6 staticGap

```
%function calculating the static gap
function [static_gap] = staticGap(static_config,seabed_VIVANA)

static_gap = static_config(2:length(static_config(:,1))-1,2) - ...
seabed_VIVANA(2:length(seabed_VIVANA(:,1))-1,2);

end
```

## A.7 FE_model

```
%function estabilshing FE-model (stiffness, damping and intertia)
function [Ksys,Csys,Msys,Ksys_initial,Csys_initial] = ...
FE_model(EI,l,T,m,n,n_dof,alpha1,alpha2)

%element matrices:

%bending stiffness
Ke = EI*[12/(l^3) -6/(l^2) -12/(l^3) -6/(l^2);
        -6/(l^2)  4/l       6/(l^2)  2/l;
       -12/(l^3)  6/(l^2)  12/(l^3)  6/(l^2);
        -6/(l^2)  2/l       6/(l^2)  4/l];

% geometric stiffness
Kg = T*[6/(5*l) -1/10  -6/(5*l) -1/10;
      -1/10    2*l/15 1/10    -1/30;
      -6/(5*l) 1/10   6/(5*l)  1/10;
      -1/10   -1/30   1/10     2*l/15];

K = Ke + Kg;

%mass matrix
M = m*l/420*[156  -22*l   54    13*l;
           -22*l  4*l^2 -13*l -3*l^2;
           54     -13*l  156   22*l;
           13*l  -3*l^2  22*l  4*l^2];

%system matrices
Ksys = zeros(n_dof);
Msys = zeros(n_dof);

%Assemble system matrices
for i = 1 : n
    j = 2*i - 2;
    for x = 1 : 4
        for y = 1 : 4
            Ksys(j+x,j+y) = Ksys(j+x,j+y)+K(x,y);
            Msys(j+x,j+y) = Msys(j+x,j+y)+M(x,y);
        end
    end
end

%introduce BCs
Ksys(1,:) = [];
Ksys(:,1) = [];
Msys(1,:) = [];
Msys(:,1) = [];

Ksys(n_dof-2,:) = [];
Ksys(:,n_dof-2) = [];
Msys(n_dof-2,:) = [];
Msys(:,n_dof-2) = [];

%damping matrix
Csys = alpha1*Msys+alpha2*Ksys;

%% Initial stiffness and damping used in non-linear analysis
Ksys_initial = Ksys;
Csys_initial = Csys;
```

## A.8 linear_analysis

```
 %linear analysis
function [y,ytrans,dy,dytrans,ddy,ddytrans,phi_exc,Qy,Qytrans,time,Ay0,Ay1,...
dymax,ddymax,N, MATLAB_f0] = ...
    linear_analysis(Ksys,Msys,Csys,k_s,c_s,sea_bottom,...
transdofs,gamma,beta,h,rho_sw,D,l,Tsim,U,static_gap)

%% adding soil stiffness to global stiffness matrix
[Ksys] = soil_stiffness(Ksys,sea_bottom,k_s,l,transdofs);

%% adding soil damping to global damping matrix
[Csys] = soil_damping(Csys,sea_bottom,c_s,l,transdofs);

%% eigenfrequencies when seabed is included
MATLAB_f0 = (1/(2*pi)) * sqrt(eig(Ksys,Msys));

%% inverting effective stiffness matrix:
Khat=Ksys+gamma/(beta*h)*Csys+1/(beta*h^2)*Msys;
Khatinv=Khat\eye(size(Khat));

%% linear time domain integration
[y,ytrans,dy,dytrans,ddy,ddytrans,phi_exc,Qy,Qytrans,time,Ay0,Ay1,dymax,...
ddymax,N] = ...
    linear_time_domain(Tsim,h,transdofs,Ksys,U,rho_sw,D,l,beta,gamma,...
Msys,Csys,Khatinv,static_gap);

end
```

## A.9 soil_stiffness

```
%adding vertical soil stiffness where we have bottom contact
function [Ksys, k_soil] = soil_stiffness(Ksys,sea_bottom,k_s,elem_length,...
transdofs_id)

%% linear stiffness matrix

%defining size of stiffness matrix
k_soil = zeros(size(Ksys));

for i = 1:length(transdofs_id)

    if sea_bottom(i) == 0

        k_soil(transdofs_id(i),transdofs_id(i)) = k_s*elem_length;

    end

end

%new global stiffness matrix including vertical springs at translation dofs
Ksys = k_soil + Ksys;

end
```

## A.10 soil_damping

```
%adding vertical soil damping where we have bottom contact
function [C] = soil_damping(Csys,sea_bottom,c_s,elem_length,transdofs_id)

%defining size of stiffness matrix
c_soil = zeros(size(Csys));

for i = 1:length(transdofs_id)

    if sea_bottom(i) == 0

        c_soil(transdofs_id(i),transdofs_id(i)) = c_s*elem_length;

    end

end

%new global damping matrix including vertical dampers at translation dofs
C = c_soil + Csys;
```

## A.11 linear_time_domain

```
%linear time domain procedure
function [y,ytrans,dy,dytrans,ddy,ddytrans,phi_exc,Qy,Qytrans,time,Ay0,Ay1...
,dymax,ddymax,N] = ...
   linear_time_domain(Tsim,h,transdofs,Ksys,U,rho_sw,D,l,beta,...
gamma,Msys,Csys,Khatinv,static_gap)

N=ceil(Tsim/h); %number of time steps

%preallocate space
y=zeros(size(Ksys,1),N);
ytrans=zeros(length(transdofs),N);              %translation
dy=zeros(size(Ksys,1),N);
dytrans=zeros(length(transdofs),N);             %translation velocity
ddy=zeros(size(Ksys,1),N);
ddytrans=zeros(length(transdofs),N);            %translation acceleration
phi_exc=zeros(length(transdofs),N);             %(only for translation dofs)
phi_exc(:,1)=2*pi*rand(length(transdofs),1);    %random phase angle
Qy=zeros(size(Ksys,1),N);
Qytrans=zeros(length(transdofs),N);
time=h*[0:N];
Ay0=zeros(length(transdofs),1);
Ay1=zeros(length(transdofs),1);
dymax=zeros(length(transdofs),1);
ddymax=zeros(length(transdofs),1);

Ui=0;           %start with zero current speed
nramp=0.1*N;    %use 10 % of sim time to ramp up current velocity
for i=1:N
    if i>nramp
        Ui=U;
    else
        Ui=Ui+U/nramp;   %ramping up current speed
    end

    %External (hydrodynamic) forces
    [Qytrans(:,i+1), phi_exc(:,i+1)] = hydroforce(h, rho_sw,...
    U*ones(length(transdofs),1), D, Ay0, dytrans(:,i), dymax, ...
       ddytrans(:,i),...
    ddymax, phi_exc(:,i));
    Qytrans(:,i+1)=Qytrans(:,i+1)*l;

    %putting in zero values at moment dofs
    for k = 1 : length(transdofs)

    Qy(transdofs(k),i+1) = Qytrans(k,i+1);

    end

    %hydrodynamic force initially at seabottom equal to zero
    [Qytrans(:,i+1)] = hydro_force_soil(Qytrans(:,i+1),static_gap);
```

```

     %% Initial conditions to use in linear Newmark-beta integration
     y0      = y(:,i);
     dy0     = dy(:,i);
     ddy0    = ddy(:,i);
     Qy1     = Qy(:,i+1);

     [y1, dy1, ddy1] = linear_newmark_beta(beta,gamma,h,Msys,Csys,Khatinv,...
     i,y0,dy0,ddy0,Qy1);

     %displacement, velocity and acceleration assigned to the matrices
     y(:,i+1)        = y1;
     dy(:,i+1)       = dy1;
     ddy(:,i+1)      = ddy1;

     %extract translations
     ytrans(:,i+1) = y(transdofs,i+1);
     dytrans(:,i+1) = dy(transdofs,i+1);
     ddytrans(:,i+1) = ddy(transdofs,i+1);

    %check for new value of maxima:
    ind3=find(abs(dytrans(:,i)>dymax));
    for j=ind3
        dymax(j)=abs(dytrans(j,i));
    end
    ind4=find(abs(ddytrans(:,i)>ddymax));
    for j=ind4
        ddymax(j)=abs(ddytrans(j,i));
    end

    %check for zero-crossing (y)

    indx1=find(dytrans(:,i).*dytrans(:,i+1) < 0);
    %if zero crossing, store current amplitude, and start over
    for j=indx1
%         Ay0(j)=Ay1(j);
%         Ay0(j)=0.5*(Ay0(j)+Ay1(j)); %average
        Ay0(j)=0.9*Ay0(j)+0.1*Ay1(j); %weighted average
        Ay1(j)=0.5*abs(dytrans(j,i+1))*h;
    end
    indx2=find(dytrans(:,i).*dytrans(:,i+1) > 0);
    %if no zero crossing, continue integration
    for j=indx2
        Ay1(j)=Ay1(j)+0.5*abs(dytrans(j,i+1))*h;
    end
end
end
```

## A.12 hydroforce

```
function [ Fy, phi_exc] = hydroforce(dt, rho, U, D, Ay, dy, dymax, ddy, ...
ddymax, phi_exc)

%find phase of velocity
cos_phi_dy=dy./abs(dymax+eps);
sin_phi_dy=-ddy./abs(ddymax+eps);
phi_dy=angle(complex(cos_phi_dy,sin_phi_dy));

%syncronize
theta=phi_dy-phi_exc; %phase difference
theta=angle(complex(cos(theta),sin(theta)));    %get phase between +-pi
omega_exc=2*pi*fhat(theta).*U./D;    %instantaneous angular frequency

%update excitation force phase
phi_exc=phi_exc+omega_exc*dt;

Cdy=0.3092+0.8929.*(Ay./D); %damping coefficient
%total cross-flow force
Fy=0.5*rho*D.*U.^2.*cv(Ay./D).*cos(phi_exc)-0.5*rho*D*Cdy.*abs(dy).*dy;
end
```

## A.13 cv

```
function [ out ] = cv(a)

x=[0
    0.1000
    0.1500
    0.2000
    0.3000
    0.4000
    0.5000
    0.6000
    0.7000
   10.0000];

    f=[0.3500
    0.5976
    0.6990
    0.7855
    0.9138
    0.9824
    0.9629
    0.5181
         0
         0];


out=interp1(x,f,a);

end
```

## A.14 fhat

```
function [ out ] = fhat( theta )
%Synchronization function, returns non-dim freq. as function of phase diff.

x=[-3.141592654
-2.857316679
-2.744445104
-2.680010174
-2.654550274
-2.604040705
-2.526906076
-2.424547672
-2.329159971
-2.253277243
-2.160905352
-2.028886016
-1.835893451
-1.570796327
-1.451609617
-1.420379807
-1.367983441
-1.327209309
-1.249466124
-1.115005502
-0.905834291
-0.650842092
-0.380828285
-0.142440673
0
0.284275975
0.397147549
0.461582479
0.48704238
0.537551949
0.614686578
0.717044982
0.812432683
0.88831541
0.980687302
1.112706638
1.305699203
1.570796327
1.689983037
1.721212847
1.773609212
1.814383345
1.892126529
2.026587151
2.235758362
2.490750561
2.760764369
2.99915198
3.141592654];

f=[0.2082
    0.2040
    0.1997
    0.1955
    0.1870
    0.1785
    0.1700
    0.1615
    0.1530
    0.1445
    0.1275
    0.1190
    0.1105
    0.1020
    0.1063
    0.1080
    0.1105
    0.1148
    0.1190
    0.1275
    0.1360
    0.1403
    0.1428
    0.1462
    0.1487
    0.1530
    0.1573
    0.1615
    0.1700
    0.1785
    0.1870
    0.1955
    0.2040
    0.2125
    0.2295
    0.2380
    0.2465
    0.2550
    0.2507
    0.2490
    0.2465
    0.2422
    0.2380
    0.2295
    0.2210
    0.2168
    0.2142
    0.2108
    0.2082];

out=interp1(x,f,theta);
```

```

end
```

## A.15 hydro_force_soil

```
%function making hydrodynamic force at initital contact point between pipe
%and soil equal to zero
function [Qytrans] = hydro_force_soil(Qytrans, static_gap)

for i = 1:length(Qytrans(:,1))

    if static_gap(i) < 0

        Qytrans(i,1) = 0;

    end

end

end
```

## A.16 linear_newmark_beta

```
%Linear Newmark-beta
function [y1, dy1, ddy1] = ...
    linear_newmark_beta(beta,gamma,h,Msys,Csys,Khatinv,...
i,y0,dy0,ddy0,Qy1)

%solve for y, dy, ddy (Newmark-B)
a=1/(beta*h^2)*y0+1/(beta*h)*dy0+(1/(2*beta)-1)*ddy0;
b=gamma/(beta*h)*y0+(gamma/beta-1)*dy0+(gamma/(2*beta)-1)*h*ddy0;
Qhat=Qy1+Csys*b+Msys*a;
y1=Khatinv*Qhat;
dy1=gamma/(beta*h)*y1-b;
ddy1=1/(beta*h^2)*y1-a;
```

## A.17 nonlinear_analysis

```
%nonlinear analysis
function [y,ytrans,dy,dytrans,ddy,ddytrans,phi_exc,Qy,Qytrans,time,Ay0,Ay1,...
dymax,ddymax,N] = nonlinear_analysis(Ksys_initial,...
Csys_initial,Ksys,Msys,Csys,k_s,c_s,sea_bottom,transdofs,gamma,beta,h,rho_sw,...
D,l,Tsim,U,static_gap,static_config,mdry,seabed_VIVANA,n,g)
    %% Time domain analysis begins

    N=ceil(Tsim/h); %number of time steps

    %preallocate space
    y=zeros(size(Ksys,1),N);
    ytrans=zeros(length(transdofs),N);              %translation
    dy=zeros(size(Ksys,1),N);
    dytrans=zeros(length(transdofs),N);             %translation velocity
    ddy=zeros(size(Ksys,1),N);
    ddytrans=zeros(length(transdofs),N);            %translation acceleration
    phi_exc=zeros(length(transdofs),N);
    phi_exc(:,1)=2*pi*rand(length(transdofs),1);    %random phase anlge
    Qy=zeros(size(Ksys,1),N);
    Qytrans=zeros(length(transdofs),N);
    time=h*[0:N];
    Ay0=zeros(length(transdofs),1);
    Ay1=zeros(length(transdofs),1);
    dymax=zeros(length(transdofs),1);
    ddymax=zeros(length(transdofs),1);

    Ui=0;           %start with zero current speed
    nramp=0.1*N;    %use 10 % of sim time to ramp up current velocity
    for k=1:N
        if k>nramp
            Ui=U;
        else
            Ui=Ui+U/nramp;   %ramping up current speed
        end

        %% nonlinear stiffness and damping matrix due to varying contact...
        %between pipe and soil bottom
        [Ksys, Csys, k_soil] = nonlinear_soil(Ksys_initial,Csys_initial,...
        ytrans,static_config,seabed_VIVANA,k,n,c_s,k_s,transdofs,l,D,rho_sw,...
        mdry,g);

        %% External (hydrodynamic) forces
        [Qytrans(:,k+1), phi_exc(:,k+1)] = hydroforce(h, rho_sw, ...
        U*ones(length(transdofs),1), D, Ay0, dytrans(:,k), dymax, ...
        ddytrans(:,k),ddymax, phi_exc(:,k));
        Qytrans(:,k+1)=Qytrans(:,k+1)*l;

        %hydrodynamic force initially at seabootom equal to zero
        [Qytrans(:,k+1)] = hydro_force_soil(Qytrans(:,k+1),static_gap);

        %putting in zero values at moment dofs
        for p = 1 : length(transdofs)
            Qy(transdofs(p),k+1) = Qytrans(p,k+1);
        end

        %% Initial conditions to use in nonlinear Newmark-beta integration
        y0      = y(:,k);
        dy0     = dy(:,k);
        ddy0    = ddy(:,k);
        Qy1     = Qy(:,k+1);

        %% Nonlinear Newmark-beta time integration
        [y1, dy1, ddy1] = nonlinear_newmark_beta(gamma,beta,h,Msys,Ksys,...
        Csys,k,y0,dy0,ddy0,Qy1);

        % displacement, velocity and acceleration assigned to the matrices
        y(:,k+1)        = y1;
        dy(:,k+1)       = dy1;
        ddy(:,k+1)      = ddy1;

        %extract translations
        ytrans(:,k+1) = y(transdofs,k+1);
        dytrans(:,k+1) = dy(transdofs,k+1);
        ddytrans(:,k+1) = ddy(transdofs,k+1);

        %%
        %check for new value of maxima:
        ind3=find(abs(dytrans(:,k)>dymax));
        for j=ind3
            dymax(j)=abs(dytrans(j,k));
        end
        ind4=find(abs(ddytrans(:,k)>ddymax));
        for j=ind4
            ddymax(j)=abs(ddytrans(j,k));
        end

        %check for zero-crossing (y)

        indx1=find(dytrans(:,k).*dytrans(:,k+1) < 0);
        %if zero crossing, store current amplitude, and start over
        for j=indx1
%         Ay0(j)=Ay1(j);
%         Ay0(j)=0.5*(Ay0(j)+Ay1(j)); %average
            Ay0(j)=0.9*Ay0(j)+0.1*Ay1(j); %weighted average
            Ay1(j)=0.5*abs(dytrans(j,k+1))*h;
        end
        indx2=find(dytrans(:,k).*dytrans(:,k+1) > 0);
        %if no zero crossing, continue integration
        for j=indx2
            Ay1(j)=Ay1(j)+0.5*abs(dytrans(j,k+1))*h;
        end

    end

```

```
end
```

## A.18 nonlinear_soil

```
%nonlinear stiffness matrix and damping matrix
function [Ksys, Csys,k_soil,c_soil] = ...
   nonlinear_soil(Ksys_initial,Csys_initial...
,ytrans,static_config,seabed_VIVANA,time_step,num_elements,c_s,k_s,...
transdofs_id,elem_length,D,rho_sw,mdry,g)

%submerged weight per meter [N/m]
w_s = (mdry - pi*((D^2)/4)*rho_sw)*g;

%function finding the node number of the two nodes on each side which are
%closest to the free span
[left_side_node, right_side_node] = last_penetration_node(static_config,...
seabed_VIVANA,transdofs_id);

%defining size of soil stiffness matrix and put equal to zero for each
%time-step
k_soil = zeros(size(Ksys_initial));
c_soil = zeros(size(Csys_initial));

if num_elements + 1 ≠ length(seabed_VIVANA(:,1))

    disp('number of elements do not match VIVANA results');

else

    for count = 1:length(transdofs_id) %do not look at first and last node

        %section 1: the pipe is initially penetrating the seabed, so we can
        %have tensioning in the soil springs
        if static_config(count+1,2) - seabed_VIVANA(count+1,2) < 0 &&...
           (count ≠ left_side_node && count ≠right_side_node)

            if ytrans(count,time_step) + static_config(count+1,2) - ...
               seabed_VIVANA(count+1,2) < (w_s/k_s)

                k_soil(2*count,2*count) = k_s*elem_length;
                c_soil(2*count,2*count) = c_s*elem_length;

            end

        %section 2: the last nodes on each side with initial pipe penetration
        %can have 0.5*tension in the soil springs
        elseif count == left_side_node || count == right_side_node

            if ytrans(count,time_step) + static_config(count+1,2) - ...
               seabed_VIVANA(count+1,2) < 0.5*(w_s/k_s)

                k_soil(2*count,2*count) = k_s*elem_length;
                c_soil(2*count,2*count) = c_s*elem_length;

            end

        %case 3: nodes in the inital free span can not have tension in the
        %soil springs
        elseif static_config(count+1,2) - seabed_VIVANA(count+1,2) > 0

            if ytrans(count,time_step) + static_config(count+1,2) - ...
               seabed_VIVANA(count+1,2) < 0

                k_soil(2*count,2*count) = k_s*elem_length;
                c_soil(2*count,2*count) = c_s*elem_length;

            end

        end

     end

end

Ksys = Ksys_initial + k_soil;
Csys = Csys_initial + c_soil;

end
```

## A.19 last_penetration_node

```
%function finding the node number of the two nodes on each side which are
%closest to the free span
function [left_side_node, right_side_node] = ...
last_penetration_node(static_config,seabed_VIVANA,transdofs_id)

for count = 1:length(transdofs_id)

    A(count) = static_config(count+1,2) - seabed_VIVANA(count+1,2);

end

for i = 2:length(A)

    if A(i-1) ≤ 0 && A(i) > 0

        left_side_node = i-1;


    elseif A(i-1) ≥ 0 && A(i) < 0

        right_side_node = i;

    end

end

end
```

## A.20 nonlinear_newmark_beta

```
%Nonlinear Newmark-beta integration, based on procedure in Dynamic analysis
%compendium
function [y1, dy1, ddy1] = ...
    nonlinear_newmark_beta(gamma,beta,h,Msys,Ksys,Csys...
,k,y0,dy0,ddy0,Qy1)
%% Initial calculations

a1 = gamma/(beta*h);
a2 = 1/(beta*h^2);
a3 = 1/(beta*h);
a4 = (1/(2*beta)) - 1;
a5 = gamma/beta;
a6 = ( (gamma/(2*beta)) - 1)/h;
a7 = a5 - 1;
a8 = (1 - gamma)*h;
a9 = gamma*h;

%mass contribution to effective stiffness
M_hat = a2*Msys;

%% For each time step

ak = a3*dy0 + a4*ddy0;
bk = a5*dy0 + a6*ddy0;

K_hat = Ksys + a1*Csys + M_hat;
ΔQ_hat = Qy1 + Csys*bk + Msys*ak - Csys*dy0 - Ksys*y0;

%% Solve wrt displacement increment

%inverting K_hat
K_hat_inv = K_hat\eye(size(K_hat));

%displacement increment
Δ_y = K_hat_inv*ΔQ_hat;

%% acceleration, velocity and displacement at time i+1

ddy1  = a2*Δ_y - ak;
dy1   = dy0 + a8*ddy0 + a9*ddy1;
y1    = y0 + Δ_y;

end
```

## A.21 POSTproc

```
%all post-processing functions
function [stress_amp, stress_rms,stress_std,fp] = ...
   POSTproc(ytrans,l,L,D,E,N,...
n,h,Tsim,c_s,k_s,type,sea_bottom,seabed_VIVANA,static_config,Case)

%% Finding stress amplitude
[stress_amp, stress_rms,stress_std] = STRESSamp(ytrans,l,D,E,N);

%% Fiding response frequnecy
[fp ang] = fpeak(ytrans(ceil(n/2),:),[0:h:Tsim]);

%% save stress results for different damping coefficients and ...
   linear/non-linear
%analysis in text-files
write = ...
   write_to_file(stress_amp,c_s,type,ytrans,N,Case,stress_rms,stress_std);

%% post-processing, plotting results
post = plots(n,N,ytrans,stress_amp,stress_rms,stress_std,l,L,D,sea_bottom,...
seabed_VIVANA,static_config,c_s,Case,type,k_s);

end
```

## A.22 STRESSamp

```
%Calculate stress using curvature

function [stress_amp, stress_rms, stress_std] = ...
    STRESSamp(ytrans,elem_length, ...
diameter, E_module, num_timesteps)

%calculate spatial derivative
y_z = diff(ytrans)/elem_length;

%calculate 2nd derivative
y_zz = diff(y_z)/elem_length;

%strain (adding zero at both ends due to BCs)
strain = [zeros(1,num_timesteps+1); (diameter/2)*y_zz;...
zeros(1,num_timesteps+1)];

%stress
stress = E_module*strain;

%% find stress envelope

%only use last part of analysis
count        = ceil(0.9*num_timesteps);
stress_red   = stress(:,[count:num_timesteps+1]);

for j = 1:length(ytrans(:,1))

        stress_amp(j) = 0.5*(max(stress_red(j,:)) - min(stress_red(j,:)));

end

%% find root mean square of stress amplitudes

for j = 1:length(ytrans(:,1))

        stress_rms(j) = rms(stress_red(j,:));
        stress_std(j) = std(stress_red(j,:));

end

end
```

## A.23 fpeak

```
function [fp phi] = fpeak(x,t)
%INPUT:
%   x: signal
%   t: time vector assoiciated with the signal, i.e x(t(i))=x(i)
%
%   timestep must be constant!
%
%OUTPUT:
%   res: Peak frequency (Hz) of the signal and associated phase (rad)
%
%Programmed by Mats Joergen Thorsen
%April, 2012

Fs=1/(t(2)-t(1));
xdft=fft(x);

[¬,index] = max(abs(xdft(1:floor(length(x)/2+1))));
freq = 0:(Fs/length(x)):Fs/2;

fp=freq(index);
phi=angle(xdft(index));

end
```

## A.24 write_to_file

```
%function writing to text-files that can be used in post-processing
function write = ...
   write_to_file(stress_amp,c_s,type,ytrans,N,Case,stress_rms,stress_std)


if Case == 1 %project thesis

if type == 1 %linear case
    if c_s ≠ 0

        %stress amplitude
        fid1        = fopen...
        ('Case1_try3\linear\stress_amp_nonzero_damping_lin.txt','w');
        formatspec  = '%f \n';

        for i = 1:length(stress_amp)

            fprintf(fid1,formatspec,stress_amp(i));

        end

        fclose(fid1);

        %response amplitude
        FID1        = fopen...
        ('Case1_try3\linear\response_amp_nonzero_damping_lin.txt','w');
        formatspec  = '%f \n';

        [maxVAL indx] = max(max(abs(ytrans(:,[ceil(0.9*N):N+1]))));

        for i = 1:length(ytrans(:,1))

            fprintf(FID1,formatspec,ytrans(i,indx + ceil(0.9*N)-1));

        end

        fclose(FID1);

    else

        fid2        = fopen...
        ('Case1_try3\linear\stress_amp_zero_damping_lin.txt','w');
        formatspec  = '%f \n';

        for i = 1:length(stress_amp)

            fprintf(fid2,formatspec,stress_amp(i));

        end

        fclose(fid2);

        %response amplitude
        FID2        = fopen...
        ('Case1_try3\linear\response_amp_zero_damping_lin.txt','w');
        formatspec  = '%f \n';

        [maxVAL indx] = max(max(abs(ytrans(:,[ceil(0.9*N):N+1]))));

        for i = 1:length(ytrans(:,1))

            fprintf(FID2,formatspec,ytrans(i,indx + ceil(0.9*N)-1));

        end

        fclose(FID2);

    end

    %% nonlinear
elseif type == 2 %nonlinear case
    if c_s ≠ 0

        fid1        = fopen...
        ('Case1_try3\nonlinear\stress_amp_nonzero_damping_nonlin.txt','w');
        formatspec  = '%f \n';

        for i = 1:length(stress_amp)

            fprintf(fid1,formatspec,stress_amp(i));

        end

        fclose(fid1);

        %response amplitude
        FID1        = fopen...
        ('Case1_try3\nonlinear\response_amp_nonzero_damping_nonlin.txt','w');
        formatspec  = '%f \n';

        [maxVAL indx] = max(max(abs(ytrans(:,[ceil(0.9*N):N+1]))));

        for i = 1:length(ytrans(:,1))

            fprintf(FID1,formatspec,ytrans(i,indx + ceil(0.9*N)-1));

        end

        fclose(FID1);

    else

        fid2        = fopen...
        ('Case1_try3\nonlinear\stress_amp_zero_damping_nonlin.txt','w');
        formatspec  = '%f \n';

        for i = 1:length(stress_amp)

            fprintf(fid2,formatspec,stress_amp(i));

        end

        fclose(fid2);

        %response amplitude
        FID2        = fopen...
        ('Case1_try3\nonlinear\response_amp_zero_damping_nonlin.txt','w');
        formatspec  = '%f \n';

        [maxVAL indx] = max(max(abs(ytrans(:,[ceil(0.9*N):N+1]))));

        for i = 1:length(ytrans(:,1))

            fprintf(FID2,formatspec,ytrans(i,indx + ceil(0.9*N)-1));

        end

        fclose(FID2);

    end

end

elseif Case == 2

if type == 1 %linear case
    if c_s ≠ 0

        %stress amplitude
        fid1        = fopen...
        ('Case2_try2\linear\stress_amp_nonzero_damping_lin.txt','w');
        formatspec  = '%f \n';

        for i = 1:length(stress_amp)

            fprintf(fid1,formatspec,stress_amp(i));

        end

        fclose(fid1);

        %response amplitude
        FID1        = fopen...
        ('Case2_try2\linear\response_amp_nonzero_damping_lin.txt','w');
        formatspec  = '%f \n';

        [maxVAL indx] = max(max(abs(ytrans(:,[ceil(0.9*N):N+1]))));

        for i = 1:length(ytrans(:,1))

            fprintf(FID1,formatspec,ytrans(i,indx + ceil(0.9*N)-1));

        end

        fclose(FID1);

    else

        fid2        = fopen...
        ('Case2_try2\linear\stress_amp_zero_damping_lin.txt','w');
        formatspec  = '%f \n';

        for i = 1:length(stress_amp)

            fprintf(fid2,formatspec,stress_amp(i));

        end

        fclose(fid2);

        %response amplitude
        FID2        = fopen...
        ('Case2_try2\linear\response_amp_zero_damping_lin.txt','w');
        formatspec  = '%f \n';

        [maxVAL indx] = max(max(abs(ytrans(:,[ceil(0.9*N):N+1]))));

        for i = 1:length(ytrans(:,1))

            fprintf(FID2,formatspec,ytrans(i,indx + ceil(0.9*N)-1));

        end

        fclose(FID2);

    end

    %% nonlinear
elseif type == 2 %nonlinear case
    if c_s ≠ 0

        fid1        = fopen...
        ('Case2_try2\nonlinear\stress_amp_nonzero_damping_nonlin.txt','w');
        formatspec  = '%f \n';

        for i = 1:length(stress_amp)

            fprintf(fid1,formatspec,stress_amp(i));

        end

        fclose(fid1);

        %response amplitude
        FID1        = fopen...
        ('Case2_try2\nonlinear\response_amp_nonzero_damping_nonlin.txt','w');
        formatspec  = '%f \n';

        [maxVAL indx] = max(max(abs(ytrans(:,[ceil(0.9*N):N+1]))));

        for i = 1:length(ytrans(:,1))

            fprintf(FID1,formatspec,ytrans(i,indx + ceil(0.9*N)-1));

        end

        fclose(FID1);

    else

        fid2        = fopen...
        ('Case2_try2\nonlinear\stress_amp_zero_damping_nonlin.txt','w');
        formatspec  = '%f \n';

        for i = 1:length(stress_amp)

            fprintf(fid2,formatspec,stress_amp(i));

        end

        fclose(fid2);

        %response amplitude
        FID2        = fopen...
        ('Case2_try2\nonlinear\response_amp_zero_damping_nonlin.txt','w');
        formatspec  = '%f \n';

        [maxVAL indx] = max(max(abs(ytrans(:,[ceil(0.9*N):N+1]))));

        for i = 1:length(ytrans(:,1))

            fprintf(FID2,formatspec,ytrans(i,indx + ceil(0.9*N)-1));

        end

        fclose(FID2);

    end

end

elseif Case == 3

if type == 1 %linear case
    if c_s ≠ 0

```

```
        %stress amplitude
        fid1        = fopen...
        ('Case3_try2\linear\stress_amp_nonzero_damping_lin.txt','w');
        formatspec  = '%f \n';

        for i = 1:length(stress_amp)

            fprintf(fid1,formatspec,stress_amp(i));

        end

        fclose(fid1);

        %response amplitude
        FID1        = fopen...
        ('Case3_try2\linear\response_amp_nonzero_damping_lin.txt','w');
        formatspec  = '%f \n';

        [maxVAL indx] = max(max(abs(ytrans(:,[ceil(0.9*N):N+1]))));

        for i = 1:length(ytrans(:,1))

            fprintf(FID1,formatspec,ytrans(i,indx + ceil(0.9*N)-1));

        end

        fclose(FID1);

    else

        fid2        = fopen...
        ('Case3_try2\linear\stress_amp_zero_damping_lin.txt','w');
        formatspec  = '%f \n';

        for i = 1:length(stress_amp)

            fprintf(fid2,formatspec,stress_amp(i));

        end

        fclose(fid2);

        %response amplitude
        FID2        = fopen...
        ('Case3_try2\linear\response_amp_zero_damping_lin.txt','w');
        formatspec  = '%f \n';

        [maxVAL indx] = max(max(abs(ytrans(:,[ceil(0.9*N):N+1]))));

        for i = 1:length(ytrans(:,1))

            fprintf(FID2,formatspec,ytrans(i,indx + ceil(0.9*N)-1));

        end

        fclose(FID2);

    end

    %% nonlinear
elseif type == 2 %nonlinear case
    if c_s ≠ 0

        fid1        = fopen...
        ('Case3_try2\nonlinear\stress_amp_nonzero_damping_nonlin.txt','w');
        formatspec  = '%f \n';

        for i = 1:length(stress_amp)

            fprintf(fid1,formatspec,stress_amp(i));

        end

        fclose(fid1);

        %response amplitude
        FID1        = fopen...
        ('Case3_try2\nonlinear\response_amp_nonzero_damping_nonlin.txt','w');
        formatspec  = '%f \n';

        [maxVAL indx] = max(max(abs(ytrans(:,[ceil(0.9*N):N+1]))));

        for i = 1:length(ytrans(:,1))

            fprintf(FID1,formatspec,ytrans(i,indx + ceil(0.9*N)-1));

        end

        fclose(FID1);

    else

        fid2        = fopen...
        ('Case3_try2\nonlinear\stress_amp_zero_damping_nonlin.txt','w');
        formatspec  = '%f \n';

        for i = 1:length(stress_amp)

            fprintf(fid2,formatspec,stress_amp(i));

        end

        fclose(fid2);

        %response amplitude
        FID2        = fopen...
        ('Case3_try2\nonlinear\response_amp_zero_damping_nonlin.txt','w');
        formatspec  = '%f \n';
        [maxVAL indx] = max(max(abs(ytrans(:,[ceil(0.9*N):N+1]))));

        for i = 1:length(ytrans(:,1))

            fprintf(FID2,formatspec,ytrans(i,indx + ceil(0.9*N)-1));

        end

        fclose(FID2);

    end

end

elseif Case == 4

if type == 1 %linear case


        %stress std
        fid1         = fopen...
        ('Case4\linear\stress_std_lin.txt','w');
        formatspec   = '%f \n';

        for i = 1:length(stress_std)

            fprintf(fid1,formatspec,stress_std(i));

        end

        fclose(fid1);

        %stress rms
        FID1         = fopen...
        ('Case4\linear\stress_rms_lin.txt','w');
        formatspec   = '%f \n';

        for i = 1:length(ytrans(:,1))

            fprintf(FID1,formatspec,stress_rms(i));

        end

        fclose(FID1);


    %% nonlinear
elseif type == 2 %nonlinear case

        %stress std
        fid1         = fopen...
        ('Case4\nonlinear\stress_std_nonlin.txt','w');
        formatspec   = '%f \n';

        for i = 1:length(stress_std)

            fprintf(fid1,formatspec,stress_std(i));

        end

        fclose(fid1);

        %stress rms
        FID1        = fopen...
        ('Case4\nonlinear\stress_rms_nonlin.txt','w');
        formatspec  = '%f \n';

        for i = 1:length(ytrans(:,1))

            fprintf(FID1,formatspec,stress_rms(i));

        end

        fclose(FID1);

end

elseif Case == 5

if type == 1 %linear case
    if c_s ≠ 0

        %stress amplitude
        fid1        = fopen...
        ('Case5\linear\stress_amp_nonzero_damping_lin.txt','w');
        formatspec  = '%f \n';

        for i = 1:length(stress_amp)

            fprintf(fid1,formatspec,stress_amp(i));

        end

        fclose(fid1);

        %response amplitude
        FID1        = fopen...
        ('Case5\linear\response_amp_nonzero_damping_lin.txt','w');
        formatspec  = '%f \n';

        [maxVAL indx] = max(max(abs(ytrans(:,[ceil(0.9*N):N+1]))));

        for i = 1:length(ytrans(:,1))

            fprintf(FID1,formatspec,ytrans(i,indx + ceil(0.9*N)-1));

        end

        fclose(FID1);

    else

        fid2        = fopen...
        ('Case5\linear\stress_amp_zero_damping_lin.txt','w');
        formatspec  = '%f \n';

        for i = 1:length(stress_amp)

            fprintf(fid2,formatspec,stress_amp(i));

        end

        fclose(fid2);

        %response amplitude
        FID2        = fopen...
        ('Case5\linear\response_amp_zero_damping_lin.txt','w');
        formatspec  = '%f \n';

        [maxVAL indx] = max(max(abs(ytrans(:,[ceil(0.9*N):N+1]))));

        for i = 1:length(ytrans(:,1))

            fprintf(FID2,formatspec,ytrans(i,indx + ceil(0.9*N)-1));

        end

        fclose(FID2);

    end

    %% nonlinear
elseif type == 2 %nonlinear case
    if c_s ≠ 0

        fid1        = fopen...
        ('Case5\nonlinear\stress_amp_nonzero_damping_nonlin.txt','w');
        formatspec  = '%f \n';

        for i = 1:length(stress_amp)

            fprintf(fid1,formatspec,stress_amp(i));

        end

        fclose(fid1);

        %response amplitude
        FID1        = fopen...
        ('Case5\nonlinear\response_amp_nonzero_damping_nonlin.txt','w');
        formatspec  = '%f \n';

        [maxVAL indx] = max(max(abs(ytrans(:,[ceil(0.9*N):N+1]))));

        for i = 1:length(ytrans(:,1))

            fprintf(FID1,formatspec,ytrans(i,indx + ceil(0.9*N)-1));

        end

        fclose(FID1);

    else

        fid2         = fopen...
        ('Case5\nonlinear\stress_amp_zero_damping_nonlin.txt','w');
        formatspec  = '%f \n';

        for i = 1:length(stress_amp)

            fprintf(fid2,formatspec,stress_amp(i));

        end

        fclose(fid2);

        %response amplitude
        FID2         = fopen...
        ('Case5\nonlinear\response_amp_zero_damping_nonlin.txt','w');
        formatspec  = '%f \n';

        [maxVAL indx] = max(max(abs(ytrans(:,[ceil(0.9*N):N+1]))));

        for i = 1:length(ytrans(:,1))

            fprintf(FID2,formatspec,ytrans(i,indx + ceil(0.9*N)-1));

        end

        fclose(FID2);

    end

end

elseif Case == 6

if type == 1 %linear case
    if c_s ≠ 0

        %stress amplitude
        fid1         = fopen...
        ('Case6\linear\stress_amp_nonzero_damping_lin.txt','w');
        formatspec  = '%f \n';

        for i = 1:length(stress_amp)

            fprintf(fid1,formatspec,stress_amp(i));

        end

        fclose(fid1);

        %response amplitude
        FID1        = fopen...
        ('Case6\linear\response_amp_nonzero_damping_lin.txt','w');
        formatspec  = '%f \n';

        [maxVAL indx] = max(max(abs(ytrans(:,[ceil(0.9*N):N+1]))));

        for i = 1:length(ytrans(:,1))

            fprintf(FID1,formatspec,ytrans(i,indx + ceil(0.9*N)-1));

        end

        fclose(FID1);

    else

        fid2        = fopen...
        ('Case6\linear\stress_amp_zero_damping_lin.txt','w');
        formatspec  = '%f \n';

        for i = 1:length(stress_amp)

            fprintf(fid2,formatspec,stress_amp(i));

        end

        fclose(fid2);

        %response amplitude
        FID2        = fopen...
        ('Case6\linear\response_amp_zero_damping_lin.txt','w');
        formatspec  = '%f \n';

        [maxVAL indx] = max(max(abs(ytrans(:,[ceil(0.9*N):N+1]))));

        for i = 1:length(ytrans(:,1))

            fprintf(FID2,formatspec,ytrans(i,indx + ceil(0.9*N)-1));

        end

        fclose(FID2);

    end
    %% nonlinear
elseif type == 2 %nonlinear case
    if c_s ≠ 0

        fid1        = fopen...
        ('Case6\nonlinear\stress_amp_nonzero_damping_nonlin.txt','w');
        formatspec  = '%f \n';

        for i = 1:length(stress_amp)

            fprintf(fid1,formatspec,stress_amp(i));

        end

        fclose(fid1);

        %response amplitude
        FID1        = fopen...
        ('Case6\nonlinear\response_amp_nonzero_damping_nonlin.txt','w');
        formatspec  = '%f \n';

        [maxVAL indx] = max(max(abs(ytrans(:,[ceil(0.9*N):N+1]))));

        for i = 1:length(ytrans(:,1))

            fprintf(FID1,formatspec,ytrans(i,indx + ceil(0.9*N)-1));

        end

        fclose(FID1);

    else

        fid2        = fopen...
        ('Case6\nonlinear\stress_amp_zero_damping_nonlin.txt','w');
        formatspec  = '%f \n';

        for i = 1:length(stress_amp)

            fprintf(fid2,formatspec,stress_amp(i));

        end

        fclose(fid2);

        %response amplitude
        FID2        = fopen...
        ('Case6\nonlinear\response_amp_zero_damping_nonlin.txt','w');
        formatspec  = '%f \n';

        [maxVAL indx] = max(max(abs(ytrans(:,[ceil(0.9*N):N+1]))));

        for i = 1:length(ytrans(:,1))

```

```
            fprintf(FID2,formatspec,ytrans(i,indx + ceil(0.9*N)-1));

        end

        fclose(FID2);

    end

end

elseif Case == 7

if type == 1 %linear case
    if c_s ≠ 0

        %stress amplitude
        fid1         = fopen...
        ('Case7\linear\stress_amp_nonzero_damping_lin.txt','w');
        formatspec   = '%f \n';

        for i = 1:length(stress_amp)

            fprintf(fid1,formatspec,stress_amp(i));

        end

        fclose(fid1);

        %response amplitude
        FID1         = fopen...
        ('Case7\linear\response_amp_nonzero_damping_lin.txt','w');
        formatspec   = '%f \n';

        [maxVAL indx] = max(max(abs(ytrans(:,[ceil(0.9*N):N+1]))));

        for i = 1:length(ytrans(:,1))

            fprintf(FID1,formatspec,ytrans(i,indx + ceil(0.9*N)-1));

        end

        fclose(FID1);

    else

        fid2         = fopen...
        ('Case7\linear\stress_amp_zero_damping_lin.txt','w');
        formatspec   = '%f \n';

        for i = 1:length(stress_amp)

            fprintf(fid2,formatspec,stress_amp(i));

        end

        fclose(fid2);

        %response amplitude
        FID2        = fopen...
        ('Case7\linear\response_amp_zero_damping_lin.txt','w');
        formatspec  = '%f \n';

        [maxVAL indx] = max(max(abs(ytrans(:,[ceil(0.9*N):N+1]))));

        for i = 1:length(ytrans(:,1))

            fprintf(FID2,formatspec,ytrans(i,indx + ceil(0.9*N)-1));

        end

        fclose(FID2);

    end

    %% nonlinear
elseif type == 2 %nonlinear case
    if c_s ≠ 0

        fid1        = fopen...
        ('Case7\nonlinear\stress_amp_nonzero_damping_nonlin.txt','w');
        formatspec  = '%f \n';

        for i = 1:length(stress_amp)

            fprintf(fid1,formatspec,stress_amp(i));

        end

        fclose(fid1);

        %response amplitude
        FID1        = fopen...
        ('Case7\nonlinear\response_amp_nonzero_damping_nonlin.txt','w');
        formatspec  = '%f \n';

        [maxVAL indx] = max(max(abs(ytrans(:,[ceil(0.9*N):N+1]))));

        for i = 1:length(ytrans(:,1))

            fprintf(FID1,formatspec,ytrans(i,indx + ceil(0.9*N)-1));

        end

        fclose(FID1);

    else

        fid2        = fopen...
        ('Case7\nonlinear\stress_amp_zero_damping_nonlin.txt','w');
        formatspec  = '%f \n';

        for i = 1:length(stress_amp)

            fprintf(fid2,formatspec,stress_amp(i));

        end

        fclose(fid2);

        %response amplitude
        FID2        = fopen...
        ('Case7\nonlinear\response_amp_zero_damping_nonlin.txt','w');
        formatspec  = '%f \n';

        [maxVAL indx] = max(max(abs(ytrans(:,[ceil(0.9*N):N+1]))));

        for i = 1:length(ytrans(:,1))

            fprintf(FID2,formatspec,ytrans(i,indx + ceil(0.9*N)-1));

        end

        fclose(FID2);

    end

end

end

write = 1;

end
```

## A.25 plots

```
%post-processing creating plots
function post = plots(n,N,ytrans,stress_amp,stress_rms,stress_std,l,L,D,...
sea_bottom,seabed,static_config,c_s,Case,type,k_s)
%% plot/animation of response

%updating y_trans including zero at ends (for plotting)
y_plot              = zeros(n+1,N+1);
y_plot(1,:)         = 0;
y_plot(n+1,:)       = 0;
y_plot(2:n,:)       = ytrans(:,:);

%% animation of response shapes

% figure;
% for i = ceil(0.9*N):N
%     plot(y_plot(:,i),[0:l:L]);
%     xlim([-2*D 2*D]);
%     title('VIV Response');
%     xlabel('displacement [m]');
%     ylabel('length of structure [m]');
%     grid on
%     F(i) = getframe;
% end
%
% fps = 100;           %frames per second
% num = 2;             %number of times to play the animation

%% Snapshots of response curves

dN = 30; %number of time steps between each response

figure;

for i = ceil(0.9*N) : dN : (N+1)

    hold on;
    plot([0:l:L],y_plot(:,i));
    title('Snapshots of response amplitudes');
    xlabel('z (m)');
    ylabel('y (m)');
    grid on;

end

%% Snapshots of response curves + seabed

dN = 50; %number of time steps between each response

figure;
```

```
for i = ceil(0.9*N) : dN : (N+1)

    hold on;
    plot([0:l:L],y_plot(:,i) + static_config(:,2));
    title('Snapshots of response amplitudes relative to seabed');
    xlabel('z (m)');
    ylabel('y (m)');
    ylim([-1010 -990]);
    xlim([0 L]);
    grid on;

end

hold on;
%plot([0:l:L],seabed(:,2),'k','lineWidth',2);
area(seabed(:,2),-1010);
grid on;

%% plot of maximum response amplitude

k = 0;
for i = ceil(0.9*N):N+1

    k = k + 1;

    val(k) = max(abs(y_plot(:,i)));

    [val2 indx] = max(val);

end

tot_indx = indx + ceil(0.9*N)-1;

figure;
plot([0:l:L],abs(y_plot(:,tot_indx)));
xlabel('z (m)');
ylabel('y (m)');
title('Maximum response amplitude');
grid on;

%% plot of maximum response amplitude compared to VIVANA and to
%linear/nonlinear model

if Case == 1
    %OBS! works only for 180 elements
    Mat = load('project_thesis\response_amplitudes_CF_VIVANA.txt');
    L_VIVANA = Mat(:,1);
    amp_VIVANA = Mat(:,2);
    figure;
    plot([0:l:L],abs(y_plot(:,indx + ceil(0.9*N)-1)),L_VIVANA,amp_VIVANA);
    grid on
    title('VIV response amplitudes');
    legend('Ulveseter linear model','VIVANA');
    xlabel('z [m]');
    ylabel('y [m]');
elseif Case == 2
    %OBS! works only for 380 elements
    Mat = load('omae2004\response_amplitudes_CF_VIVANA_omae04.txt');
    L_VIVANA = Mat(:,1);
    amp_VIVANA = Mat(:,2);
    figure;
    plot([0:l:L],abs(y_plot(:,indx + ceil(0.9*N)-1)),L_VIVANA,amp_VIVANA);
    grid on
    title('VIV response amplitudes');
    legend('Ulveseter linear model','VIVANA');
    xlabel('z [m]');
    ylabel('y [m]');
end


%% plot of stress amplitude if c_s = 0
%%% LINEAR SOLUTION %%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
if c_s == 0 && type == 1
figure;
plot([0:l:L-2*l],10^-6*stress_amp);
grid on;
xlabel('length of structure [m]');
ylabel('\sigma [MPa]');
title('VIV Stress amplitdes');
end

%% plot of stress amplitude for Thorsen and VIVANA

if c_s == 0 && type == 1

    if Case == 1
        %OBS! only for 180 elements
        Mat = load('project_thesis\stress_amplitudes_VIVANA.txt');

        L_VIVANA = Mat(1:length(Mat(:,1))-2,1);
        stress_amp_VIVANA = Mat(1:length(Mat(:,1))-2,2);
        figure;
        plot([0:l:L-2*l],10^-6*...
        load('postproc\stress_amp_zero_damping_lin.txt'),L_VIVANA,...
        stress_amp_VIVANA);
        grid on;
        legend('Ulveseter linear model','VIVANA');
        title('VIV stress amplitudes');
        xlabel('z [m]');
        ylabel('\sigma [MPa]');
    elseif Case == 2
        %OBS! only for 381 elements
        Mat = load('omae2004\stress_amplitudes_VIVANA_omae04.txt');
        L_VIVANA = Mat(1:length(Mat(:,1))-2,1);
        stress_amp_VIVANA = Mat(1:length(Mat(:,1))-2,2);
        figure;
        plot([0:l:L-2*l],10^-6*...
        load('case2\linear\stress_amp_zero_damping_lin.txt'),L_VIVANA,...
        stress_amp_VIVANA*10^-3);
        grid on;
        legend('Ulveseter linear model','VIVANA');
        title('VIV stress amplitudes');
        xlabel('z [m]');
        ylabel('\sigma [MPa]');
    end

end




%% plot of stress amplitude if c_s = 0
%%% NONLINEAR SOLUTION %%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
if c_s == 0 && type == 2
figure;
plot([0:l:L-2*l],10^-6*stress_amp);
grid on;
xlabel('length of structure [m]');
ylabel('\sigma [MPa]');
title('VIV Stress amplitudes, nonlinear');
end

%% plot of stress amplitude for Thorsen and VIVANA

if c_s == 0 && type == 2

    if Case == 1
        %OBS! only for 180 elements
        Mat = load('project_thesis\stress_amplitudes_VIVANA.txt');
        L_VIVANA = Mat(1:length(Mat(:,1))-2,1);
        stress_amp_VIVANA = Mat(1:length(Mat(:,1))-2,2);
        figure;
        plot([0:l:L-2*l],10^-6*load...
        ('postproc\stress_amp_zero_damping_nonlin.txt'),L_VIVANA,...
        stress_amp_VIVANA);
        grid on;
        legend('soil damping = 0, Thorsen','stress amplitudes VIVANA');
        title('VIV stress amplitudes, nonlinear');
        xlabel('z [m]');
        ylabel('\sigma [MPa]');
    elseif Case == 2
        %OBS! only for 381 elements
        Mat = load('omae2004\stress_amplitudes_VIVANA_omae04.txt');
        L_VIVANA = Mat(1:length(Mat(:,1))-2,1);
        stress_amp_VIVANA = Mat(1:length(Mat(:,1))-2,2);
        figure;
        plot([0:l:L-2*l],10^-6*load...
        ('case2\nonlinear\stress_amp_zero_damping_nonlin.txt'),L_VIVANA,...
        stress_amp_VIVANA);
        grid on;
        legend('soil damping = 0, Thorsen','stress amplitudes VIVANA');
        title('VIV stress amplitudes, nonlinear');
        xlabel('z [m]');
        ylabel('\sigma [MPa]');
    end

end

%% COMPARISON BETWEEN LINEAR AND NONLINER SOLUTION %%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

if Case == 1
figure;
plot([0:l:L-2*l],10^-6*load('postproc\stress_amp_nonzero_damping_lin.txt'),...
[0:l:L-2*l],10^-6*load('postproc\stress_amp_nonzero_damping_nonlin.txt'));
legend(['linear case, soil damping = ' num2str(c_s)],...
['nonlinear case, soil damping = ' num2str(c_s)]);
grid on;
title('VIV stress amplitudes');
xlabel('z [m]');
ylabel('\sigma [MPa]');
end


%% plot rms of stress-envelope

figure;
plot([0:l:L-2*l],10^-6*stress_rms,[0:l:L-2*l],10^-6*stress_std);
legend('stress rms','stress std');
grid on;
xlabel('length of structure [m]');
ylabel('stress amplitude [MPa]');
title('VIV Stress Response');


%% plot seabed

% figure;
% plot(sea_bottom(:,2),sea_bottom(:,1));
% grid on;
% xlabel('z [m]');
% ylabel('y [m]');
% title('Geometry of seabottom');
% ylim([-10 10]);

%% plot seabed from VIVANA-analysis in project work + static config

figure;
plot(seabed(:,1),seabed(:,2),static_config(:,1),static_config(:,2));
grid on;
legend('seabed','static configuration');
xlim([0 L]);
ylim([-1010 -990]);
xlabel('z [m]');
```

```
ylabel('y [m]');

%% POSTER%%
%plot seabed from VIVANA-analysis in project work + static config

figure;
%plot(seabed(:,1),seabed(:,2),'LineWidth',2,'Color',[0 0 1]);
%hold on;
area(seabed(:,2),-1010);
hold on;
%area()
plot(static_config(:,1),static_config(:,2),'LineWidth',3,'Color',[0 1 1]);
grid on;
legend('seabed','pipeline');
xlim([0 L]);
ylim([-1010 -990]);
xlabel('z [m]');
ylabel('y [m]');

%% plot
post = 1;
end
```

# Appendix B

# Algorithm for nonlinear Newmark-$\beta$ time integration

The inspiration for the time integration procedure used in Ulveseter's nonlinear model is based on the following algorithm. It is presented in (Langen and Sigbjörnsson, 1979).

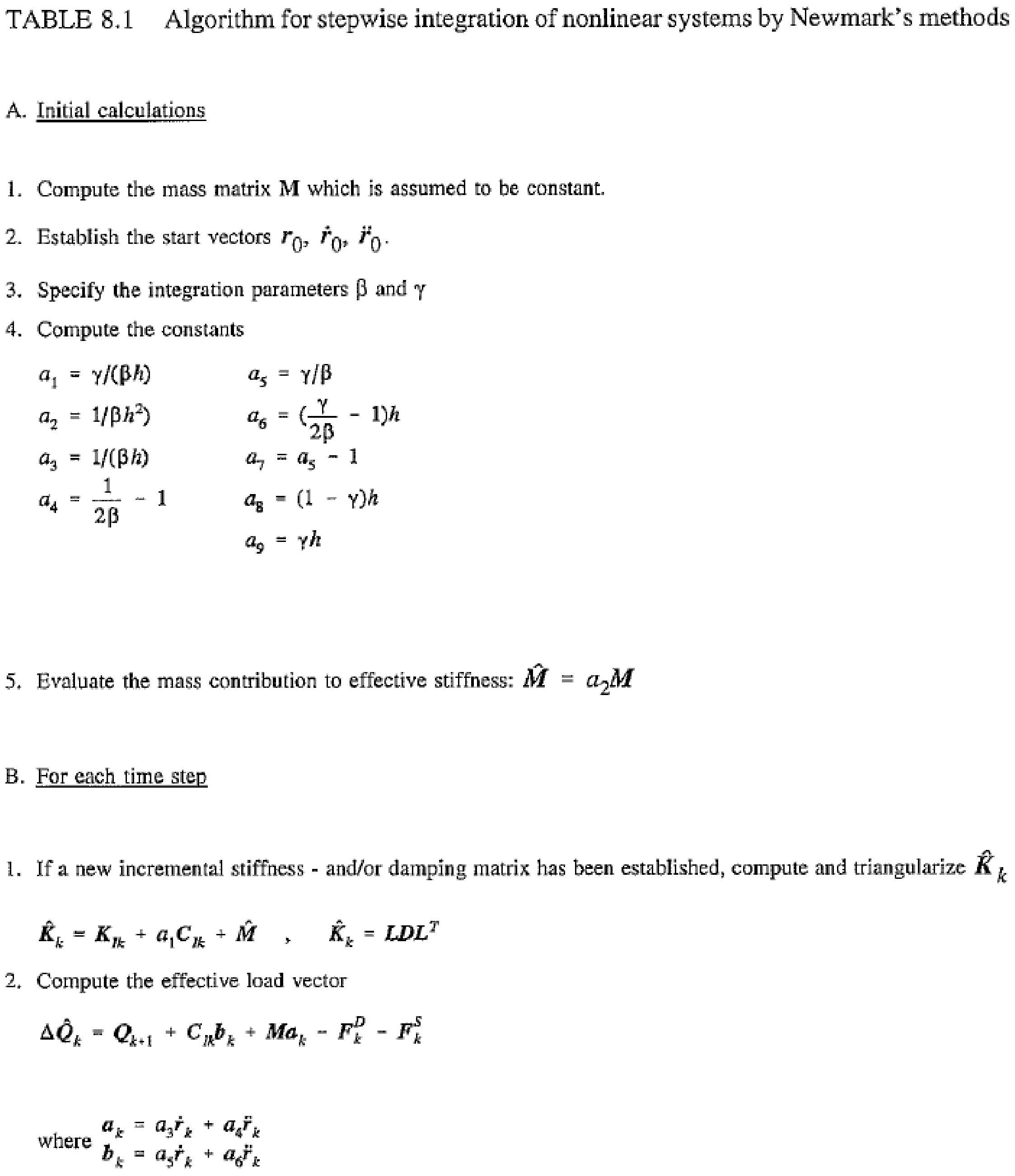


TABLE 8.1 Algorithm for stepwise integration of nonlinear systems by Newmark's methods

A. Initial calculations

1. Compute the mass matrix **M** which is assumed to be constant.
2. Establish the start vectors $\boldsymbol{r}_0$, $\dot{\boldsymbol{r}}_0$, $\ddot{\boldsymbol{r}}_0$.
3. Specify the integration parameters β and γ
4. Compute the constants

$$a_1 = \gamma/(\beta h) \qquad a_5 = \gamma/\beta$$
$$a_2 = 1/\beta h^2) \qquad a_6 = (\frac{\gamma}{2\beta} - 1)h$$
$$a_3 = 1/(\beta h) \qquad a_7 = a_5 - 1$$
$$a_4 = \frac{1}{2\beta} - 1 \qquad a_8 = (1 - \gamma)h$$
$$a_9 = \gamma h$$

5. Evaluate the mass contribution to effective stiffness: $\hat{\boldsymbol{M}} = a_2\boldsymbol{M}$

B. For each time step

1. If a new incremental stiffness - and/or damping matrix has been established, compute and triangularize $\hat{\boldsymbol{K}}_k$

$$\hat{\boldsymbol{K}}_k = \boldsymbol{K}_{Ik} + a_1\boldsymbol{C}_{Ik} + \hat{\boldsymbol{M}} \quad , \qquad \hat{\boldsymbol{K}}_k = \boldsymbol{LDL}^T$$

2. Compute the effective load vector

$$\Delta\hat{\boldsymbol{Q}}_k = \boldsymbol{Q}_{k+1} + \boldsymbol{C}_{Ik}\boldsymbol{b}_k + \boldsymbol{M}\boldsymbol{a}_k - \boldsymbol{F}_k^D - \boldsymbol{F}_k^S$$

where $\begin{matrix} \boldsymbol{a}_k = a_3\dot{\boldsymbol{r}}_k + a_4\ddot{\boldsymbol{r}}_k \\ \boldsymbol{b}_k = a_5\dot{\boldsymbol{r}}_k + a_6\ddot{\boldsymbol{r}}_k \end{matrix}$

3.Solve with respect to the displacement increment. The values of the **L**, **D** factors which were computed last are employed, i.e.

$$\boldsymbol{LDL}^T\Delta\boldsymbol{r}_k = \Delta\hat{\boldsymbol{Q}}_k$$

4. If required, iteration is performed to achieve dynamic equilibrium:

a) Set: $${}^0\Delta\boldsymbol{r}_k = \Delta\boldsymbol{r}_k \text{ , } i = 1$$ $$\boldsymbol{d}_k = a_7\dot{\boldsymbol{r}}_k + a_6\ddot{\boldsymbol{r}}_k$$

b) Compute approximations to acceleration, velocity and displacement

$${}^{i-1}\ddot{\boldsymbol{r}}_{k+1} = a_2\,{}^{i-1}\Delta\boldsymbol{r}_k - \boldsymbol{a}_k$$

$${}^{i-1}\dot{\boldsymbol{r}}_{k+1} = a_1\,{}^{i-1}\Delta\boldsymbol{r}_k - \boldsymbol{d}_k$$

$${}^{i-1}\boldsymbol{r}_{k+1} = \boldsymbol{r}_k + {}^{i-1}\Delta\boldsymbol{r}_k$$

c) Compute the effective residual forces:

$${}^{i-1}\Delta\boldsymbol{F}_k = \boldsymbol{Q}_{k+1} - \boldsymbol{M}^{i-1}\ddot{\boldsymbol{r}}_{k+1} - {}^{i-1}\boldsymbol{F}^D_{k+1} - {}^{i-1}\boldsymbol{F}^S_{k+1}$$

d) Solve with respect to the correction of the displacement increment

$$\boldsymbol{LDL}^T\;{}^i\Delta = {}^{i-1}\Delta\boldsymbol{F}_k$$

e) Compute the corresponding new displacement increment

$${}^i\Delta\boldsymbol{r}_k = {}^{i-1}\Delta\boldsymbol{r}_k + {}^i\Delta$$

f) Test for convergence

$$\|{}^i\Delta\|_{M2} < tol$$

If the convergence test is satisfied, go to step 5. If convergence is not achieved and if i < MAXIT, set i = i + 1 and go to (b). If i > MAXIT, stop the procedure for a restart.

5. Compute acceleration, velocity and displacement at time $t_{k+1}$

$$\ddot{\boldsymbol{r}}_{k+1} = a_2\Delta\boldsymbol{r}_k - \boldsymbol{a}_k$$

$$\dot{\boldsymbol{r}}_{k+1} = \dot{\boldsymbol{r}}_k + a_8\ddot{\boldsymbol{r}}_k + a_9\ddot{\boldsymbol{r}}_{k+1}$$

$$\boldsymbol{r}_{k+1} = \boldsymbol{r}_k + \Delta\boldsymbol{r}_k$$

# Bibliography

Aronsen, K. (2006). *Hydrodynamic coefficients for in-line Vortex Induced VIbrations*. PhD thesis, NTNU.

Blevins, R. (1990). Flow-induced vibration. In *2nd Edition, Van Nostrand Reinhold*.

Facchinetti, M. L., de Langre, E., and Billey, F. (2004). Coupling of structure and wake oscillators in vortex-induced vibrations. *Jornal of Fluids and Structures*.

Faltinsen, O. M. (1990). *Sea Loads on Ships and Offshore Structures*. Cambridge University.

Finn, L., Lambrakos, K., and Maher, J. (1999). Time domain prediction of riser viv. In *4th International Conference on Advances in Riser Technologies*.

Halse, K. H. (1997). *On vortex shedding and prediction of vortex-induced vibrations of circular cylinders*. PhD thesis, NTNU.

Hansen, N.-E. O. (1982). Vibrations of pipe arrays in waves. In *Proc. Third Int. Conf. on Behaviour of Offshore Structures*, pages 641–650.

Langen, I. and Sigbjörnsson, R. (1979). *Dynamic Analysis of Constructions (Dynamisk Analyse av Konstruksjoner)*. Tapir.

Larsen, C. M. Vortex induced vibrations (viv) part ii. Power point presentation on advanced topics of VIV.

Larsen, C. M. (2011). *Vortex Induced Vibrations VIV*. Department of Marine Technology.

Larsen, C. M. (2014). Handbook on design and operation of flexible pipes. Joint Industry Project: "Safe and Cost Effective Operation of Flexible Pipes" Volume 1, MARINTEK, 4subsea, NTNU.

Larsen, C. M., Baarholm, G. S., Passano, E., and Koushan, K. (2004). Non-linear time domain analysis of vortex induced vibrations for free spanning pipelines. In *Proceedings of OMAE'04: 23rd International Conference on Offshore Mechanics and Arctic Engineering*.

Larsen, C. M. and Halse, K. H. (1997). Comparison of models for vortex induced vibrations of slender marine structures. *Marine Structures*.

Larsen, C. M. and Koushan, K. (2005). Empirical model for the analysis of vortex induced vibration for free spanning pipelines. In *European Conference on Structural Dynamics (Eurodyn)*. Centre for Ship and Ocean Structures.

Larsen, C. M. and Passano, E. (2006). Time and frequency domain analysis of catenary risers subjected to vortex induced vibrations. In *OMAE2006*.

Lie, H. (1995). A time domain model for simulation of vortex induced vibratins on a cable. In *Proceedings of Sixth International Conference on Flow Induced Vibrations*.

Lie, H., Larsen, C. M., and Tveit, Ø. (2001). Vortex induced vibration analysis of catenary risers. In *Offshore Technology Conference*.

Lie, H., Larsen, C. M., Yttervik, R., Passano, E., and Baarholm, G. S. (2008). *VIVANA - Theory Manual Version 3.6*.

Mainçon, P. and Larsen, C. M. (2011). Towards a time-domain finite element analysis of vortex induced vibrations. In *OMAE2011*.

Morse, T. and Williamson, C. (2009). Prediction of vortex- induced vibration response by employing controlled motion. *Jornal of Fluid Mechanics*.

Passano, E., Larsen, C. M., Lie, H., and Wu, J. (2014). *VIVANA Theory Manual Release 4.2*.

Pettersen, B. (2007). *Marine Technology 3 Hydrodynamics*. Department of Marine Technology.

Richardson, P. L. and Reverdin, G. (1987). Seasonal cycle of velocity in the atlantic north equatorial countercurrent as measured by surface drifters, current meters and ship drifts. *Journal of Geophysical Research 92*.

Sarpkaya, T. (1978). Fluid forces on oscillating cylinders. *Jornal of the Waterway, Port, Coastal and Ocean Division*.

Steen, S. and Minsaas, K. (2013). *Ship Resistance*. Department of Marine Technology.

Sumer, B. M. and Fredsøe, J. (2006). *Hydrodynamics around cylindrical structures*, volume 26. World Scientific Publishing Co. Pte. Ltd.

Sunden, B. Tubes, crossflow over.

Thorsen, M., Sævik, S., and Larsen, C. (2014). A simplified method for time domain simulation of cross-flow vortex-induced vibrations. *Jornal of Fluids and Structures*.

Thorsen, M. J., Sævik, S., and Larsen, C. M. (2015a). Fatigue damage from time domain simulation of combined in-line and cross-flow vortex-induced vibrations. *Marine Structures*.

Thorsen, M. J., Sævik, S., and Larsen, C. M. (2015b). Time domain simulation of vortex-induced vibrations based on phase-coupled oscillator synchronization. In *OMAE2015*.

Ulveseter, J. V. (2014). The effect from damping on amplitudes of vortex induced vibrations. *Project Thesis*.

Venugopal, M. (1996). *Damping and Response Prediction of a Flexible Cylinder in a Current*. PhD thesis, Massachusetts Institule of Technology.

Veritas, D. N. (2006). *Free Spanning Pipelines, DNV-RP-F105.*

Vikestad, K., Larsen, C. M., and Vandiver, J. (2000). Norwegian deepwater program: damping of vortex- induced vibrations. In *Offshore Technology Conference.*

Yttervik, R. (2004). *Ocean current variability in relation to offshore engineering.* PhD thesis, NTNU.

Zarantonello, E. H. and Brikhoff, G. (1957). *Jets, Wakes, and Cavities.* Adademic Press Inc. New York.