NTNU
Norwegian University of Science and Technology
Department of Marine Technology

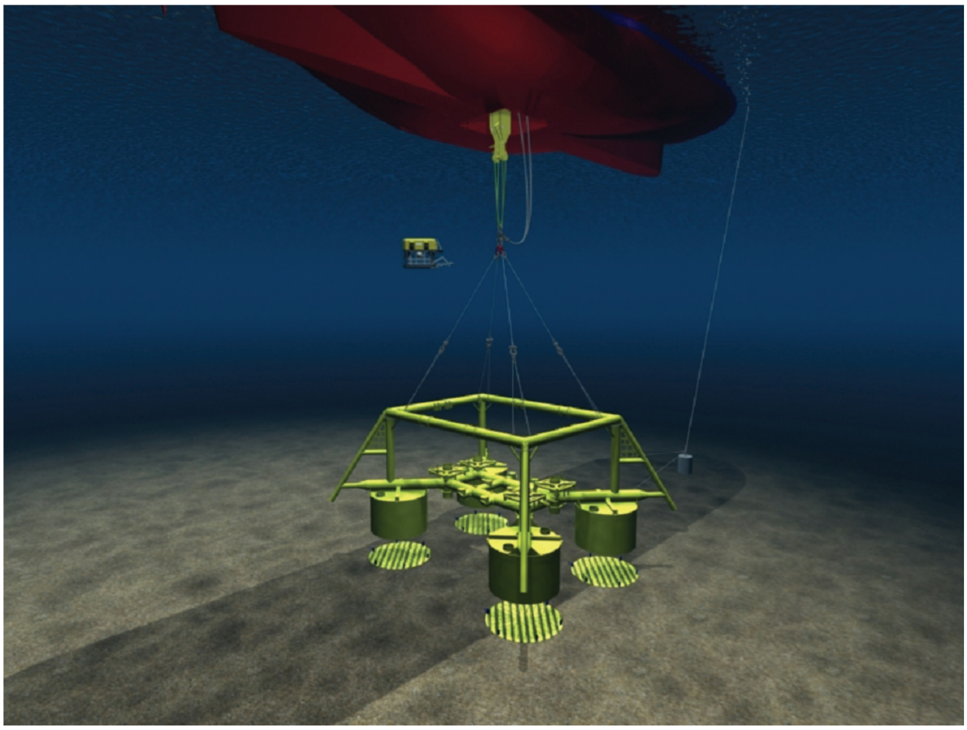
subsea 7

Master thesis, spring 2011

for

Stud. Tech. Torbjørn Aakerøy Olsen

Subsurface towing of heavy module



Keywords: **Advisors:**

Marine operations	**Delivered: 03.06.2011**	Bernt Johan Leira, NTNU
Single-degree-of-freedom system	**Number of pages: 242**	Daniel Karunakaran, Subsea 7
Multiple-degree-of-freedom system	**Availability: Closed**	
Dynamic response of template		

NTNU
Norwegian University of Science and Technology
Department of Marine Technology

NTNU
Norwegian University of Science and Technology
Department of Marine Technology

Master Thesis, Spring 2011

for

Stud. Tech. Torbjørn Olsen

Subsurface towing of heavy module

Undervanns tauing av tung modul

A heavy Integrated template structure (ITS) is to be towed by Subsea 7 by suspending it from the vessel MSV Botnica. This implies that the dynamic forces and response of the assembled system need to be analyzed.

The following subjects are to be examined in this thesis:

1. An overview of the different towing methods available in the market and a thorough description of the subsea 7 method used in the 1-dof and multiple dof analysis should be given.

2. The simple Matlab program developed in the project thesis is to be expanded and improved to make it versatile for different projects. The theory behind the Matlab program is to be described and an analysis of the transit phase of the Tyrihans project done by Subsea 7 is to be performed.

3. An overview of the multiple degree of freedom theory and how it is implemented in Orcaflex is to be given. A methodology for modelling and simulation of the towing operation done in the Tyrihans project is to be established. Possible simplifications that can be made are to be discussed, and response analysis is performed. The results from the calculation shall be discussed and compared to the 1-DOF of freedom mathematical model.

4. Two parametric studies are to be performed. A parametric study of the template motion with varying added mass and a parametric study of the offset angle with varying drag coefficient are to be performed and compared in both Matlab and Orcaflex.

5. The statistical properties of the extreme tension in the suspension line during the tow operation are to be considered. Samples of the extreme values for a sequence of different sea states are first to be obtained based on numerical simulation of the system. By combining the resulting estimated extreme value distributions for these sea-states with the relevant joint distribution of sea-state parameters, the unconditional extreme-value distribution is subsequently to be computed. Application of the resulting information for design purposes is to be discussed.

The work-scope may prove to be larger than initially anticipated. Subject to approval from the supervisor, topics may be deleted from the list above or reduced in extent.

In the thesis the candidate shall present his personal contribution to the resolution of problems within the scope of the thesis work. Theories and conclusions should be based on mathematical derivations and/or logic reasoning identifying the various steps in the deduction. The candidate should utilise the existing possibilities for obtaining relevant literature.

The thesis should be organised in a rational manner to give a clear exposition of results, assessments, and conclusions. The text should be brief and to the point, with a clear language. Telegraphic language should be avoided.

The thesis shall contain the following elements: A text defining the scope, preface, list of contents, summary, main body of thesis, conclusions with recommendations for further work, list of symbols and acronyms, references and (optional) appendices. All figures, tables and equations shall be numerated.

The supervisor may require that the candidate, at an early stage of the work, presents a written plan for the completion of the work. The plan should include a budget for the use of computer and laboratory resources which will be charged to the department. Overruns shall be reported to the supervisor.

The original contribution of the candidate and material taken from other sources shall be clearly defined. Work from other sources shall be properly referenced using an acknowledged referencing system.

The thesis shall be submitted in 2 copies:

- Signed by the candidate

- The text defining the scope included
- In bound volume(s)
- Drawings and/or computer prints which cannot be bound should be organised in a separate folder.

Supervisor: Professor Bernt J. Leira

Contact person at Subsea7: Daniel Karunakaran

Start: January 17th, 2011

Deadline: June 14th, 2011

Trondheim, 17 January 2011

Bernt J. Leira

NTNU
Norwegian University of Science and Technology
Department of Marine Technology

Preface

This master thesis is written by Torbjørn Aakerøy Olsen for and in cooperation with the Norwegian University of Science and Technology and Subsea 7. The thesis consists of a theoretical study, a programming assignment in Matlab and a modeling assignment in Orcaflex.

The study has been carried out under the supervision of my advisor at Subsea 7 Daniel Karunakaran, and my advisor at NTNU Bernt Johan Leira. Their advice, guidance and support are gratefully acknowledged.

I want to thank Jørgen Reine and Tommy Andresen, Subsea 7, for their guidance in the computer program Orcaflex and for providing the MSV Botnica data.

I am thankful to Kenneth Aarset, Subsea 7, for his guidance and valuable discussions regarding different towing methods and for providing papers and presentations of the different methods.

I would also like to express my gratitude to Subsea 7 for all the resources provided to me through this cooperation.

The Subsea 7 method described in this thesis, with all its illustrations, calculations and data is patent pending.

Torbjørn Aakerøy Olsen

03.06.2011

Abstract

The offshore industry is expanding, new contracts are offered all over the world and new oil fields are discovered deeper and further offshore. With this expansion follows great business opportunities for existing and new subsea companies. With this kind of competition it is crucial for the companies to be innovative and offer competitive solutions to a wide range of problems.

This thesis provides a description of different methods of transporting a subsea structure to its installation site. Traditionally the structure is transported on deck of a heavy lift vessel and lowered through the splash zone at the installation site. This is a weather sensitive method and provides dynamic challenges when lowering the structure through the splash zone. A different option is to perform a subsurface towing of the structure. This is still considered as a relatively new concept even though it has been conducted several years with success. A subsurface towing is safer for the personnel, more cost efficient and less weather sensitive. However transportation on deck is often preferred due to its short transportation time.

The subsea 7 method of performing a subsurface towing was used in the Tyrihans project and this method and project will form the basis of the analysis done in this thesis. A simple 1-DOF program was developed in the project thesis written by the author the fall of 2010. This program has been subject to extensive improvement to make it suitable to do an analysis of the Tyrihans project. The results provided from the 1-DOF program will be compared with the results provided from Orcaflex in terms of time histories and FFTs of the time histories. In addition to these comparisons; a parametric study of the templates heave motion with varying added mass coefficients as well as a parametric study of the offset angle with varying drag coefficients are conducted.

Table of Contents

List of figures

List of tables

Nomenclature

L	-	Wire length
w	-	Submerged weight per unit length
	-	Mass of the wire per unit length
W_0	-	Submerged weight at the end of the wire
m	-	Mass per unit length
ρ_w	-	Density of water
A	-	Crossection area
α	-	Offset angle of the wire
	-	Weibull scaling parameter
	-	Measure of the dispersion of the extreme variate X_n
F_{D0}	-	Drag force on the body connected to the end of the wire
q	-	Drag force per unit length
z	-	Vertical position
P_0	-	Pressure
v	-	Velocity of the current (vessel)
C_D	-	Drag coefficient
A	-	Refference area
T_E	-	Effective tension
EA	-	Stiffness per unit length
M	-	Mass at the bottom end of the wire
	-	Mass of the vessel
	-	Symmetric mass matrix
	-	Mass matrix
η_a	-	Oscillating amplitude
ω	-	Oscillating frequency
	-	Wave frequency
	-	Load frequency
w	-	Mass of the wire per unit length
V	-	Volume of the wire
A_L	-	Projected area of load
η	-	Dynamic elongation of the wire
m_c	-	Complex mass of the wire
M_c	-	Complex mass of the template
c_v	-	Estimated drag coefficient of the wire
C_v	-	Estimated drag coefficient of the template
C_e	-	Linear damping coefficient of the template
H_s	-	Significant wave height
T_P	-	Peak period

ω_P	-	Peak frequency
γ	-	Peak enhancement factor in the JONSWAP spectrum
	-	Accuracy parameter in the Newmark beta method
	-	Euler number
σ	-	Constant in the JONSWAP spectrum
T_{m0z}	-	Zero crossing period
T_{m0e}	-	Mean period
T_{m24}	-	Mean period between wave crests
H_{m0}	-	Calculated significant wave height
$H_{\max}$	-	Highest wave in a given period
$H_{\frac{1}{N}}$	-	Highest wave with a possibility of 1/N
T_z	-	Mean zero crossing period
ζ_{An}	-	Wave amplitude of wave component n
ω_n	-	Wave frequency of wave component n
k_n	-	Wave number of wave component n
ε_n	-	Random phase angle of wave component n
N	-	Number of wave components
	-	Interpolation polynomial
	-	Number of samples in the simulated time history
E	-	Energy
S	-	Wave spectrum
A_{33}	-	Added mass coefficient in heave for the vessel
B_{33}	-	Damping coefficient in heave for the vessel
A_{33}^{2D}	-	2 dimensional added mass coefficient in heave for the vessel
B_{33}^{2D}	-	2 dimensional damping coefficient in heave for the vessel
C_{33}	-	Stiffness coefficient in heave for the vessel
$\ddot{\eta}_3, \dot{\eta}_3, \eta_3$	-	Acceleration, velocity and displacement in heave of the vessel
F_3	-	External heave force on the vessel
A_w	-	Area of the water plane
$H(\omega)_3$	-	Transfer function in heave for the vessel
g	-	Gravitation constant
C	-	Symmetric viscous damping matrix
	-	Damping matrix
K	-	Symmetric stiffness matrix
	-	Stiffness matrix
$\ddot{d}, \dot{d}, d$	-	Acceleration, velocity and displacement vectors
d_n	-	Approximation of the displacement in the Newmark beta method
v_n	-	Approximation of the velocity in the Newmark beta method

a_n	-	Approximation of the acceleration in the Newmark beta method
$\tilde{d}_{n+1}$	-	Predictor of the displacement in step "n+1" in the Newmark beta method
$\tilde{v}_{n+1}$	-	Predictor of the velocity in step "n+1" in the Newmark beta method
β	-	Stability parameter in the Newmark beta method
$Q(t)$	-	External load matrix as a function of the time
p	-	Position
v	-	Velocity
a	-	Acceleration
	-	Wave amplitude
r	-	Position vector
	-	DOF of a structure
	-	Vector of nodal displacements
$\dot{r}$	-	Velocity vector
$\ddot{r}$	-	Acceleration vector
t	-	Simulation time
ρ	-	Density of a material
u	-	Displacement of a alement
	-	The characteristic largest value of the initial variate X
T	-	Kinetic energy
	-	Wave period
m_{ij}	-	Inertia of displacement parameter "i" because of a unit acceleration of displacement parameter "j"
P	-	Load vector
$(k_{jk})_i$	-	Stiffness at end "j" because of rotation in end "k". In element "i"
c_i	-	Damping matrix of one element
$c(x)$	-	Distributed damping force
ϕ	-	Vector of natural oscillation forms
ϕ_i	-	Natural oscillation form of global DOF "i"
y	-	Vector of displacement amplitudes
$Q_j(t)$	-	Component "j" of the harmonic load
Q_{0j}	-	Amplitude of component "j" of the harmonic load
α_j	-	Phase angle of component "j" of the harmonic load
X_j	-	Complex load vector
$\delta(t-\tau)$	-	Dirac delta function
I	-	Load impulse
h_{ij}	-	Response in DOF "i" because of a unit impulse in DOF "j"
$\ddot{u}_k, \dot{u}_k, u_k$	-	Acceleration, velocity and displacement at time step "k"
h	-	Size of the time step

$\ddot{u}(t)$	-	Assumed shape of the acceleration
u_n	-	The characteristic largest value of initial variate X
α_n	-	An inverse measure of dispersion of the extreme variate X_n
μ_y	-	Expected extreme value
μ	-	Weibull location parameter
α	-	Weibull scaling parameter
λ	-	Weibull shape factor
x_i	-	Sample values
$\tilde{m}_n$	-	N'th sample central moment
$\tilde{\mu}$	-	Sample mean
$\hat{\mu}$	-	Estimate of the expected value of a set of data
$\hat{\sigma}^2$	-	Estimate of the variance of a set of data
$C.O.V$	-	Coefficient of variance
u	-	The characteristic largest value of initial variate X
α	-	An inverse measure of dispersion of the extreme variate X_n
$\hat{\sigma}$	-	Estimate of the standard deviation of a set of data
$\hat{\mu}$	-	Estimate of the expected value of a set of data
F_D	-	Drag force
C_D	-	Drag coefficient
A_P	-	Projected area
C_{Deq}	-	Equivalent drag coefficient
A_{tot}	-	Total projected area
A_i	-	Projected area of one unit
C_{Di}	-	Drag coefficient of one unit
K_i	-	Axial stiffness of element "i"
K_{eq}	-	Equivalent axial stiffness
$U(Z)$	-	Current velocity at the point Z
U_{seabed}	-	Current velocity at the seabed
$\mathrm{U}_{surface}$	-	Current velocity at the surface
Z_{seabed}	-	Z-position at the seabed
$Z_{surface}$	-	Z-position at the surface
$Exponent$	-	Power law exponent
$S(f,\theta)$	-	Wave spectrum
$S_d(\theta)$	-	Directional spreading spectrum
$S_f(f)$	-	Frequency spectrum
$K(s)$	-	Normalising constant
$2s$	-	Spreading exponent
θ	-	Wave direction

θ_p	-	Principal wave direction
x	-	Vessel displacement
R	-	RAO amplitude
φ	-	RAO phase
t	-	Time
μ_{ex}	-	Average of the expected extreme values
$\hat{\alpha}$	-	Gumbel estimator of α in the Gumbel distribution
$\hat{u}$	-	Gumbel estimator of u in the Gumbel distribution
$\hat{\mu}_y$	-	Estimate for the expected value of a set of data
$\hat{s}_y$	-	Estimate for the variance of a set of data

Notation:

The plots of the time histories, FFTs of the time histories and the offset angles are given with legends ”Matlab” and “Orcaflex” (Chapter 7,8 and appendix 12.4 - 12.8). It should be noted that this means the results produced by the 1-DOF Matlab program and multiple-DOF results produced by Orcaflex respectively.

Abbreviations

CDF	Cumulative Density Function
CDTM	Controlled depth tow method
COG	Centre of gravity
C.O.V	Coefficient of variance
DNV	Det norske veritas
DOF	Degree of freedom
FFT	Fast Fourier Transformation
ID	Inner diameter
ITS	Integrated template structure
JONSWAP	Joint North Sea wave project spectrum
KG	Distance between keel and COG
LOA	Length over all
LPP	Length between perpendiculars
MBL	Minimum breaking load
OD	Outer diameter
PDF	Probability Density Function
PM	Pierson-Moskowitz
RAO	Response amplitude operator
ROV	Remotely operated vehicle
SWL	Safe work load
WL	Water line
WT	Wall thickness
MNPC	Matrix of Nodal Point Correspondence

Summary

The main purpose of this thesis was to extend and improve the 1-DOF Matlab program developed in the project thesis written the fall of 2010. This 1-DOF program was to analyze the towing performed in the Tyrihans project by Subsea 7 and compare the results with the ones produced by Orcaflex.

There are several alternatives for towing a structure from shore to its installation site. Some of these alternatives have been presented and a thorough description of the Subsea 7 method has been given. This method is the one Subsea 7 used for the Tyrihans project and is a wet tow method with the template suspended through the moonpool from the installation vessel.

To extend and improve the Matlab program and to create a model in Orcaflex, it is crucial to understand the 1-DOF and multiple DOF theory. This theory has been presented and used for all coding and modeling.

The results from the 1-DOF program is presented as graphs of the time histories of the surface elevation, vessel motion in heave and template motion in heave for both regular and irregular waves. The comparison between the 1-DOF and multiple-DOF regular analysis is easy as the vessel can experience exactly the same waves. With irregular waves the comparison gets more complex and the time histories are compared by calculating the standard deviation and performing a FFT to identify at which frequency the signal of the time history is at its strongest. This gives good results for both the surface elevation and vessel motion in heave. For the template motion there was an additional frequency peak after doing the FFT. This additional frequency peak was presumed to have its roots in the numerical integration in 1-DOF program. To investigate this phenomenon further, a parametric study of the template motion with varying added mass coefficients was performed. The results of this parametric study explained the deviation between the template motion in the 1-DOF and multiple-DOF analysis. It was concluded that the additional frequency peak occurred because of an insufficient damping and was solved by introducing an artificial damping in the numerical integration in the 1-DOF program.

The 1-DOF program calculates the offset angle as a function of the towing velocity. These results differ slightly from the ones produced in Orcaflex. The multiple-DOF analysis has an initial offset angle due to non symmetry in the modeling of the template. The towing velocity of interest is around 1,5 m/s and the offset angle in the two analyses in this area is close with the results from the 1-DOF program being a bit more conservative. To do an additional check of how the results match, a simple model with the template modeled as a lump mass and the hang off wire modeled as an equivalent wire is made in Orcaflex. Comparing the 1-DOF

program to these results confirms that the 1-DOF results are good but slightly more conservative.

The statistical properties of the extreme tensions in the suspension lines are of interest for design purposes. In this thesis the extreme tension of one of the suspension lines is calculated with the use of three different methods. It was found that the best method was to calculate the expected extreme tension in a suspension line from a Gumbel distributed set of maxima.

The results of the analysis and parametric studies led to the conclusion that the 1-DOF program gives good results for an initial simple check of a subsurface towing configuration according to the subsea 7 method. However the program needs to be tested for different projects and compared to different programs to check its reliability.

1 Introduction

The offshore industry is expanding, new contracts are offered all over the world and new oil fields are discovered deeper and further offshore. With this expansion follows great business opportunities for existing and new subsea companies. With this kind of competition it is crucial for the companies to be innovative and offer competitive solutions to a wide range of problems.

A new concept of subsurface towing of a heavy subsea structure has been developed and conducted with success by Subsea 7. This method involves suspending the structure from the moonpool of the vessel, allowing the hang-off point being as close to the vessels COG as possible. This method improves the weather criteria significantly and makes the operation safer to the personnel as all operations are done subsea by an ROV. In addition this method eliminates the need of a heavy lift installation vessel which means it is feasible with smaller installation vessels; hence it is a more cost efficient method. Even though there are lots of benefits of using this method for transporting the structure to its installation site, there is still a tendency of the clients preferring a transport with the structure on deck due to its shorter operation time.

The main objective in this thesis is to extend and improve the simple 1-DOF program made in the project thesis of the author and creating a model in Orcaflex. The 1-DOF program is to analyze the towing done in the Tyrihans project conducted by Subsea 7. Subsequently the results of the offset angle, surface elevation, heave motion of the vessel and template are to be compared with the results obtained from the Orcaflex model. These results are subjected to further comparison by means of a parametric study of the template motion with varying added mass coefficients and offset angle with varying drag coefficients. A second objective is to consider the statistical properties of the extreme tension in the suspension line during the tow operation. Numerous samples of the extreme values for a sequence of different sea states are first obtained based on numerical simulation of the system in Orcaflex. The expected extreme tension in one of the suspension lines is subsequently calculated with three different methods and compared.

The thesis is organized in three parts, theory, case study and results.

The first part is the theory and is presented in chapter 2 to 4. Chapter 2 considers different towing methods available and gives a thorough description of the subsea 7 subsurface towing method that forms the basis of this thesis. Chapter 3 gives an overview of the 1-DOF theory implemented in the 1-DOF program as well as the multiple DOF theory that Orcaflex is based on. Chapter 4 describes the extreme statistics theory that the calculation of the expected extreme tension in one suspension line is based on.

The second part is the case study and consists of chapter 5. This chapter describes the configuration of the system that is to be analyzed. It also describes how the modelling is done and which simplifications and assumptions that are made in the 1-DOF program and Orcaflex.

The third part is the results and consists of chapter 7 to 9. Chapter 7 compares the surface elevation, vessel motion and template motion in heave. These results are compared for both regular and irregular waves. Chapter 8 gives the results of two parametric studies. The first parametric study is to investigate the template motion with varying added mass coefficient in the 1-DOF and multiple-DOF analysis. The second parametric study is to investigate the offset angle with varying drag coefficients. Chapter 9 gives the results of three methods to calculate the expected extreme tension in one suspension line.

Chapter 10 gives a conclusion of the results and a recommendation of further work is given.

2 Towing methods

Transportation of a large offshore structure is complex and provides challenges to the industry. There are several towing techniques developed by different companies. Each has its advantages and challenges. Whether to choose a wet tow or transportation on deck is based on the time limit, weather conditions and the size of the structure. The tendency is that transportation on deck is the most popular method due to the short transportation time relative to the submerged towing techniques. However, the wet tow methods are still preferred in some cases and new methods are still being developed due to limited availability of heavy-lift vessels. In this chapter there will be given a short description of some transportation techniques, and a thorough description of the method used in this thesis, the subsea 7 method.

2.1 Transportation on deck

The subsea structure can be transported on the deck of a vessel and lowered down through the water at the desired location. This type of transportation is a fast, but also a weather sensitive method. During the operation the structure is exposed to wind and wave excitation forces, slamming forces, current forces and forces from the vessel. In this kind of transportation there is also a need for a significantly larger vessel and a crane with a much larger capacity compared to a wet tow operation.

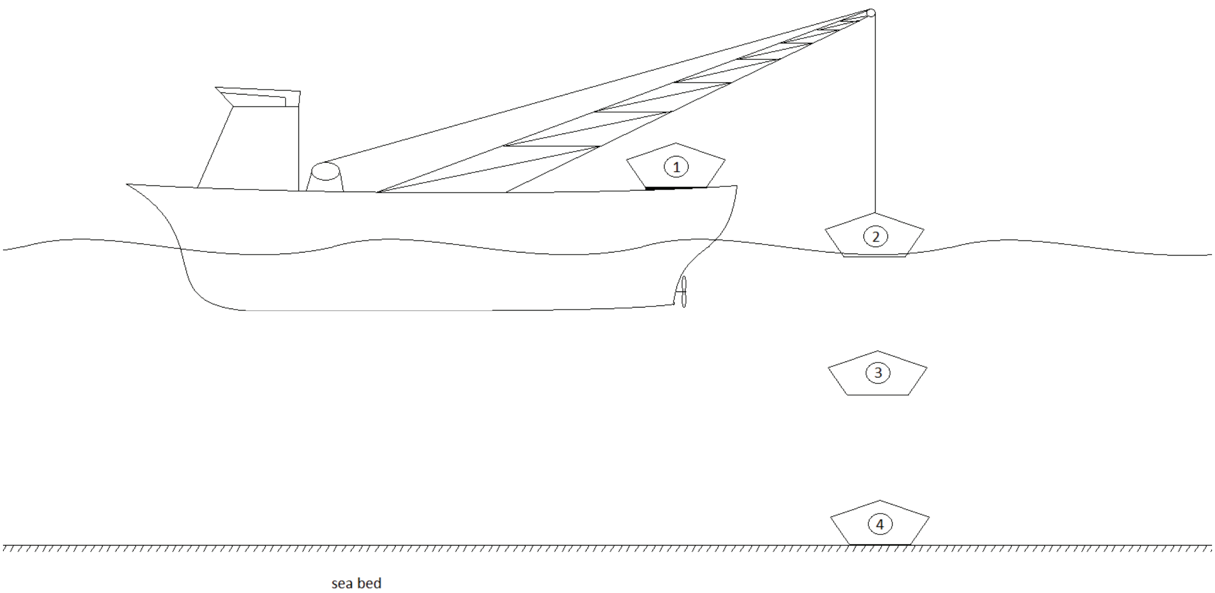


Figure 1 Phases in a lifting operation

The lifting operation can be divided into four phases:

1. Lifting the structure from the deck
2. Lowering the structure through the splash zone
3. Lowering the structure through the current zone
4. Placing the structure on the sea bed

The most critical part of this operation is the lowering through the splash zone. The structure will experience slamming loads from the waves in addition to the viscous forces due to the structures motion. A big concern is the effect of the buoyancy. The buoyancy of the module may cause a slack in the wire and subsequently large snag loads.

It is convenient to divide the crane operations in light lifts and heavy lifts.

Light lifts: When the load is assumed very small relative to the vessel it may be assumed that the motion characteristics of the vessel is unaffected by the presence of the load.

Heavy lifts: The load is assumed to affect the motion characteristics of the vessel and there will be a mutual interaction between the load and the vessel.

2.2 Wet tow methods

In wet tow methods the subsea structure is lowered through the splash zone at inshore sheltered areas. This requires less crane capacity and eliminates risk elements related to pendulum motion in air, and slamming/uplift loads during lowering through the splash zone.

Since the need for a large crane capacity and a large deck space for transportation are eliminated, a smaller vessel can be used for the transportation and installation. Also since the structure is submerged, the forces due to the weather conditions decrease significantly with the depth of the towing.

By using a wet tow method, all operations are done under water; this is a great advantage regarding safety of the personnel on deck.

2.2.1 Pencil Buoy method

The pencil buoy method is a technique developed by Aker, and is a method for transportation and installation of subsea structures. This method has also been successfully used for recovery of structures.

In this method the structure is transported on deck from the fabrication site to the load-out site. This will improve the operation time since the wet tow distance is kept at a minimum. At the load-out site the structure is lifted from the transportation barge and the structure rigging will be connected to the installation winch and the pencil buoy. The structure is lowered and the pencil buoy is launched from the installation vessel, which leads to the structure and rigging weight being completely carried by the pencil buoy.

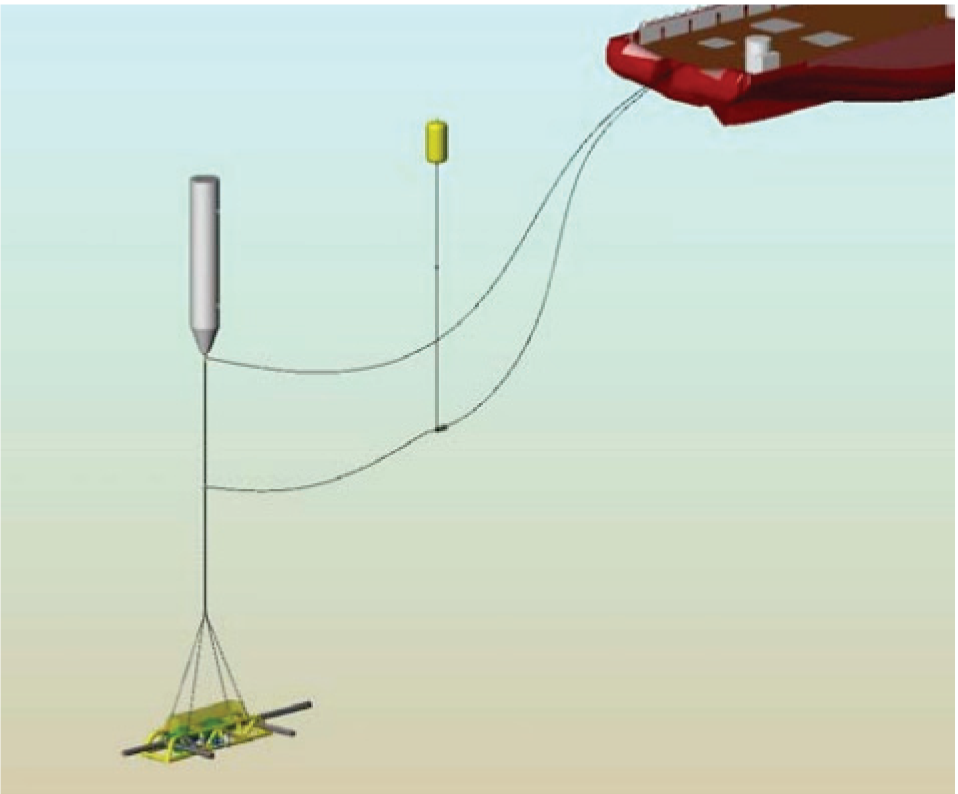


Figure 2 Pencil-buoy set-up, reference [1]

The wet tow is normally done at a speed of 3 – 3,5 knots from the load-out site to the installation site. When the structure is to be installed, the structure weight is transferred back to the towing winch wire and the buoy is disconnected. Now the structure can be lowered down and installed on the seabed. The lowering is done using a passive heave compensator.

A disadvantage using this method is that it requires more operations to lower the structure than for example the subsea 7 method. Aker also had problems with fatigue of the pad eye design. This has been improved and the method is now, according to Aker, working fine. Reference [1].

2.2.2 Wet tow of a bundle

Due to the strong demand of pipe lay vessels there is a great advantage with fabrication, welding and testing the bundle onshore. After the bundle is approved it will be wet towed to the installation site. This leads to a safe and controlled operation and a product which is fully tested onshore. There is no doubt that welding and testing out on the field will be more complicated, time consuming and will be a lot more expensive. The high pipe lay vessel rates make the tow concept more competitive.

The choice of tow method is dependent of the submerged weight and length of the towed system as well as the environment, seabed properties and existing pipelines along the towing route.

In the following concept three methods are used for the complete towing operation, the off-bottom-, controlled depth- and catenary tow method.

In the description of these methods an example of a combined usage of the three methods will be used. Reference [2] and [3].

2.2.2.1 Off-bottom tow method

The off-bottom tow method is used from the fabrication site to a predetermined point offshore, and is used where the conditions of the seabed is known. To control the submerged weight and stability of the bundle, buoyancy tanks and chains are mounted at frequent intervals. This makes it possible to control at which distance from the seabed the bundle is to be towed.

Since the buoyancy gets more expensive as the depth increases, off-bottom towing is only used to a certain water depth. Compared to the controlled depth tow method, the off-bottom tow method uses a lower towing speed, but still the fatigue damage is smaller since the bundle is located further away from the surface. Reference [2] and [3].

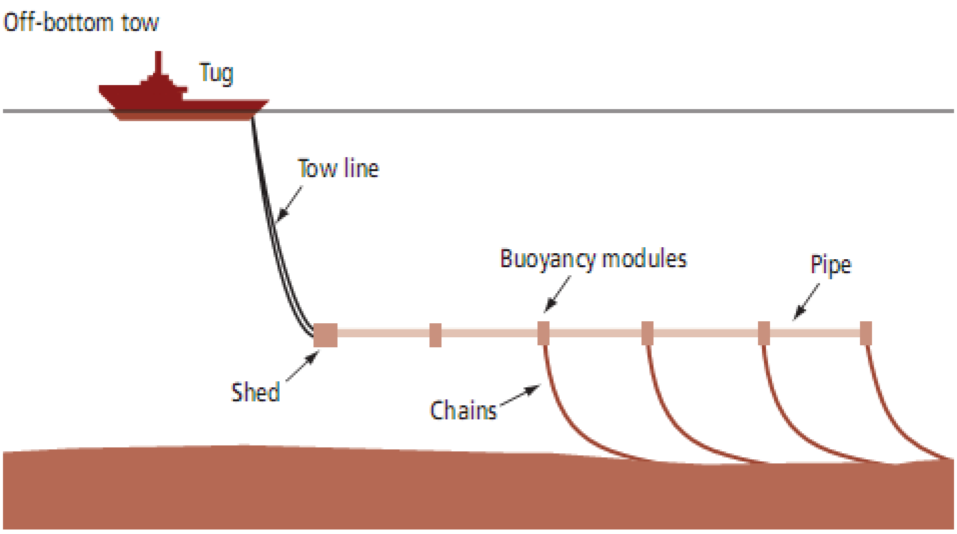


Figure 3 Off-bottom tow method, reference [2]

2.2.2.2 Controlled depth tow method (CDTM)

The controlled depth tow method, referred to as CDTM, is used from a predetermined point to a temporary location offshore. In this method the bundle is kept between the leading and trailing tug.

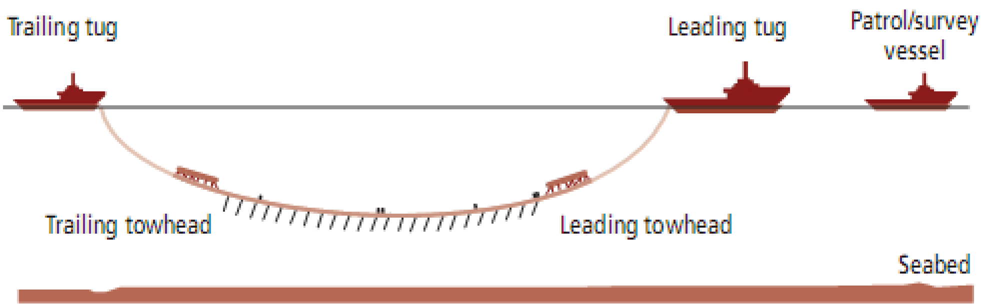


Figure 4 Controlled depth tow method (CDTM), reference [2]

The buoyancy tanks and chains are still used to control the buoyancy, which is now negative. In addition the drag on the chains will produce a lift which will affect the submerged weight. The lift produced by the chains will be dependent on the speed of the water, type of chains and the number of links.

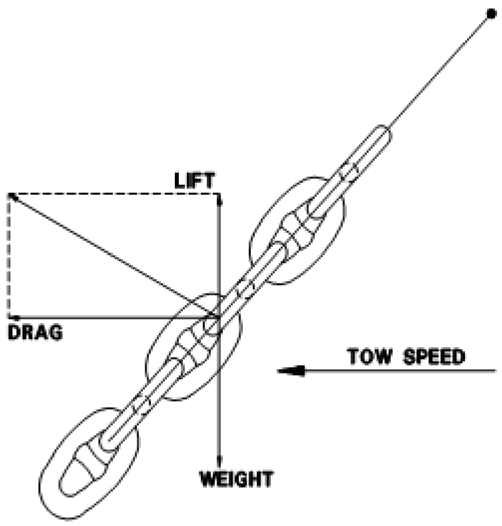


Figure 5 Forces on the tow chain during tow, reference [3]

There are some advantages using CDTM compared to the off-bottom tow method. CDTM allows a higher towing velocity, a maximum of about 6,8 knots. The CDTM never allows the bundle to be in contact with the sea bed; hence severe slopes and rocky conditions can be passed with ease. Reference [2] and [3].

2.2.2.3 Catenary tow

At the installation site the buoyancy tanks and chains are removed and a catenary tow is performed. Since the bundle is hanging between the two tugs and contact with the sea bed is to be avoided, this method cannot be used at shallow waters as the required horizontal bollard pull forces to keep the pipeline sag-bend of the seabed are too high for conventional tugs. Reference [2] and [3].

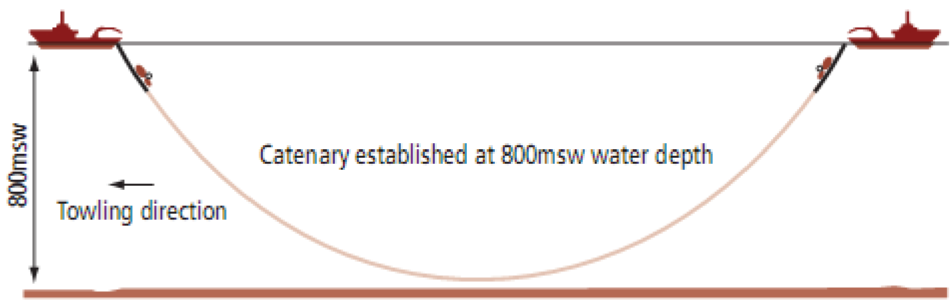


Figure 6 Catenary tow method, reference [2]

2.2.3 Subsea 7 method

As this thesis is written in cooperation with Subsea 7, the Subsea 7 method of submerged towing will be used. The operation of transporting four heavy templates to the Tyrihans field, done by Subsea 7 in cooperation with Statoil forms the basis of the analysis, therefore examples from this operation will be used in explaining the method.

The subsea 7 method allows the use of a small monohull construction vessel to wet tow and install massive subsea structures in harsh environmental conditions in a single operation. This method has proved to be safe and cost efficient compared to the traditional use of heavy lift vessels.

The concept has been used by subsea 7 before, but with lighter structures and the towing done from the vessel side using the installation crane. To improve the towing criteria and enable towing of heavy structures, the towing is done through the moonpool of the vessel. To reduce the effects of vessel-motions, the hang-off point is preferred to be as close to the vessels motion center as possible. This enables the operation to be performed in harsh environmental conditions and is solved by using a self designed hang-off tower. This hang-off tower is installed over the moonpool of the installation vessel.

Challenges in the subsea 7 method can be divided into geographical, operational and challenges due to the template properties;

Geographic

- Harsh environmental conditions
- Tow distance and fatigue
- Water depth for installation – Vertical resonance

Template properties

- Massive weight
- Large hydrodynamic loads due to suction anchors

Operational

- Heavy rigging
- Complex ROV operations
- Non heave compensated system

The method can be divided into the following operations. Some operations are illustrated in Figure 7:

1. Wet-store of the template
2. Pick up and hang-off
3. Tow to field
4. Transfer load to heavy lift winch system
5. Installation

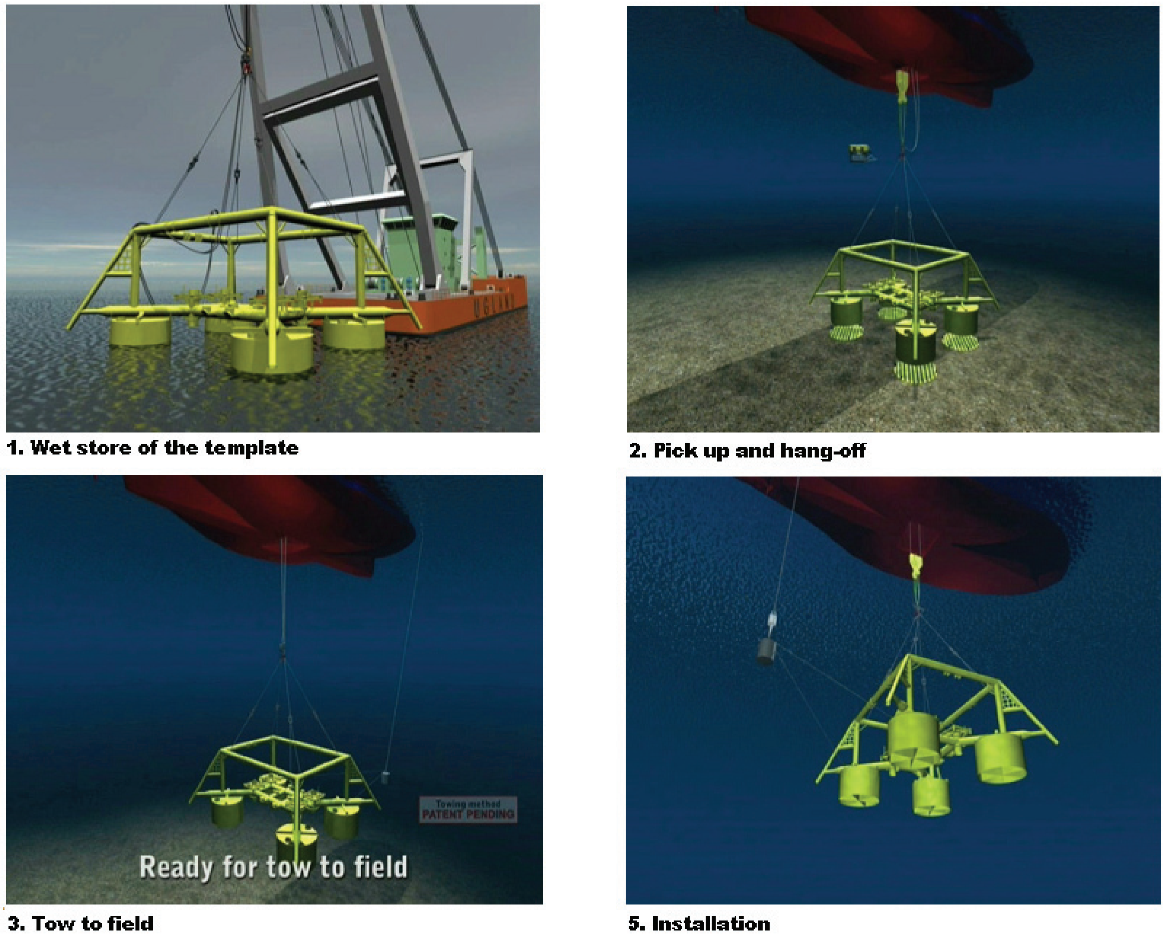


Figure 7 Illustration of four operation stages; wet-store, pick up and hang-off, tow to field and installation. Reference [4]

The following key conclusions were done by Subsea 7 after using this method to install four templates at the Tyrihans field, reference [4]:

- No manual handling of heavy rigging offshore
- All heavy lifts were performed inshore in sheltered waters
- Extremely limited exposure to personnel
- Cost-effective solution
- Ensures availability of vessels
- Limited use of “sophisticated” cranes and crane modes subject to higher risk of technical / software failures

2.2.3.1 Wet store

The structure to be towed offshore is first transported to a suitable location for a wet store. This is done by a heavy lift vessel. The structure can now be lowered through the splash zone in a sheltered location, and stored until it is ready to be towed offshore. Each structure is stored with a buoy holding all the rigging, preventing the rigging from damaging the structure. The buoys are marked and the coordinates are plotted to separate the structures that are stored.

A suitable location is chosen by the following criteria:

- Sheltered location within tow range of installation field
- Water depth corresponding to rigging length and lift height
- Nearby facilities, quay, storage capacity on quayside, availability of mobile crane, equipment etc.
- Sufficient area with flat seabed without obstructions
- Deep water tow route from location to field.

If the seabed on the chosen location is too soft, the structure will penetrate and if soft enough, sink through the soil. This was the case when Subsea 7 was to tow four templates to the Tyrihans field. Since the soil at the chosen location was too soft, Subsea 7 installed mud mat plates on each of the four suction anchors of the template. This prevented the template from penetrating the seabed. Reference [4], [6].

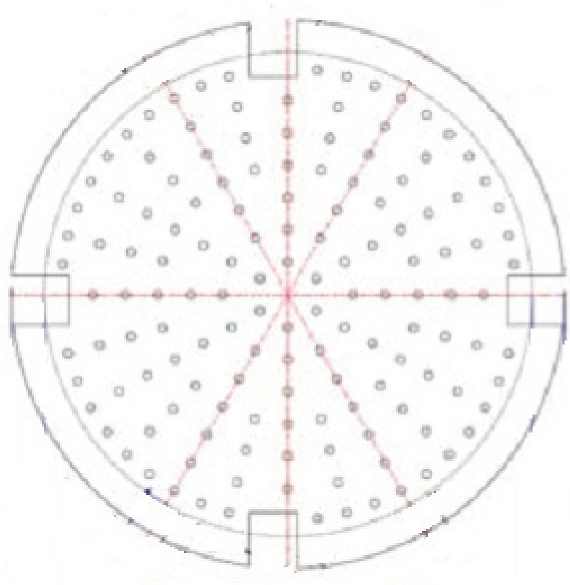
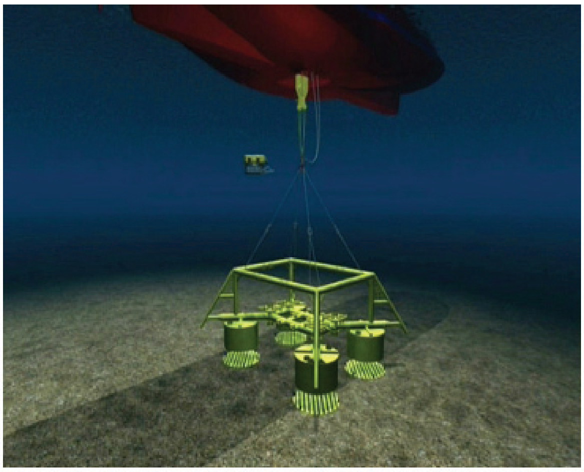


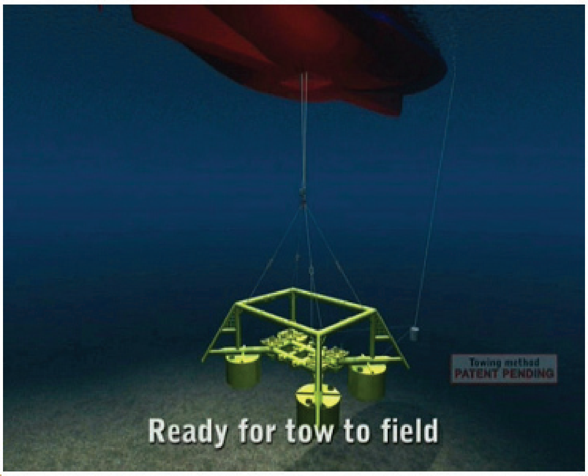
Figure 8 Illustration of the mud mats installed on each suction anchor. Reference [5].

2.2.3.2 Pick up and hang-off

The installation vessel that is to tow the structure to its destination picks up the structure at its storage location. The Installation crane is connected to the structures rigging with an ROV and lifted of the seabed before the hang-off wires can be attached to the vessel. During the tow the structures weight will be carried only by the two hang-off wires. These wires are fitted to the structure before it is placed on the wet store location, and will be attached to the installation vessel with the ROV while the weight of the structure is carried by the installation crane. When the hang-off wires are connected to the installation vessel, the installation crane lowers the structure until its weight is completely carried by the two hang-off wires. The ROV can now detach the installation crane from the structure and the structure is ready to be towed. Reference [5], [6].



1. The weight of the structure carried by the installation crane



2. The weight of the structure carried by the two hang-off wires

Figure 9 Illustration of the pick-up and hang-off of the structure. Reference [4]

2.2.3.3 Tow to field

The towing speed depends on waves and current, and a typical towing speed is about 3 knots. Before a towing can be executed, the weather forecast has to be analyzed so that the towing can be commenced when the weather is stable and showing a downward trend. The weather criteria have to be established according to the DNV rules for planning and execution of marine operations.

Figure 10 Configuration for transit to deeper waters. Reference [4]

The configuration for transit to deeper waters is as shown in Figure 10. The lump weight suspended from the bow of the installation vessel and connected to the front of the template is preventing the template from twisting during towing. Reference [5].

2.2.3.4 Transfer load to heavy lift winch system

Before installing the template on the seabed, the weight of the template has to be transferred from the hang-off wires to the heavy lift winch system that is used in the installation. This is done with the same procedure as described in chapter 2.2.3.2.

2.2.3.5 Installation

The first step to install the template on the seabed is to transfer the load to the heavy lift winch system. This is done as described in chapter 2.2.3.4. When the weight is carried by the installation winch and slack in the hang-off wires are obtained, the hang-off wires are cut by an ROV. This is the point of no return.

Figure 11 Configuration for installing the template. Reference [4].

As shown in Figure 11, a lump weight is suspended from the installation vessels crane and connected to the template. This is done to be able to control the orientation of the template. Now the template is ready to be lowered down and installed. After landing the template on the seabed, the ROV is connected to the template to control the suction anchors. The suction anchors are adjusted one by one so that the template remains leveled. When the template is fully anchored and leveled, the rigging is removed and the installation is complete. Reference [5], [6].

2.2.4 Wet tow over the side

Subsea 7 has done wet towing with the template hanging over the side. This procedure was used in the Heidrun project, and is very similar to the Subsea 7 method described in chapter 2.2.3. They are both wet tow methods that avoids the lowering through the splash zone. Risk of injuries to the personnel is reduced as all the operations are done subsea by ROVs. Both methods eliminate the need of heavy lift vessels which makes them both cost efficient relative to transporting the structure on deck.

By using a wet tow over the side:

- The structure can be wet stored or picked up directly from a lecter with the vessels crane.
- The structure is suspended over the side of the vessel from a hang-off beam during the transit to the installation site.
- The installation is done using a crane.

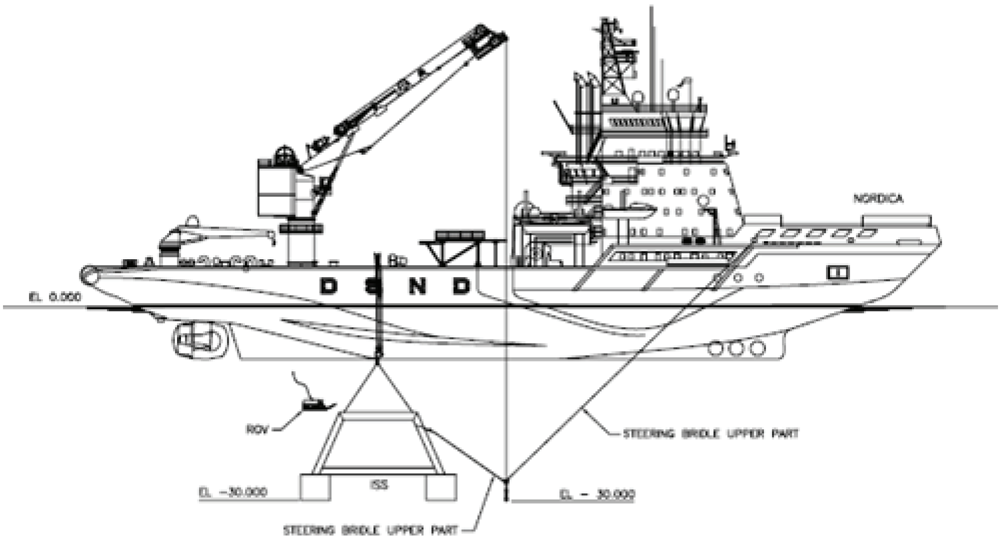


Figure 12 Configuration of the system in the transit phase. Reference [32]

The towing procedure is as follows:

1. Pick up the structure from a wet store or dry store with the vessels crane.
2. Transfer the weight of the structure to the hang-off beam.
3. Transit to installation site.
4. Transfer the weight of the structure to the crane.
5. Install the structure using the crane.

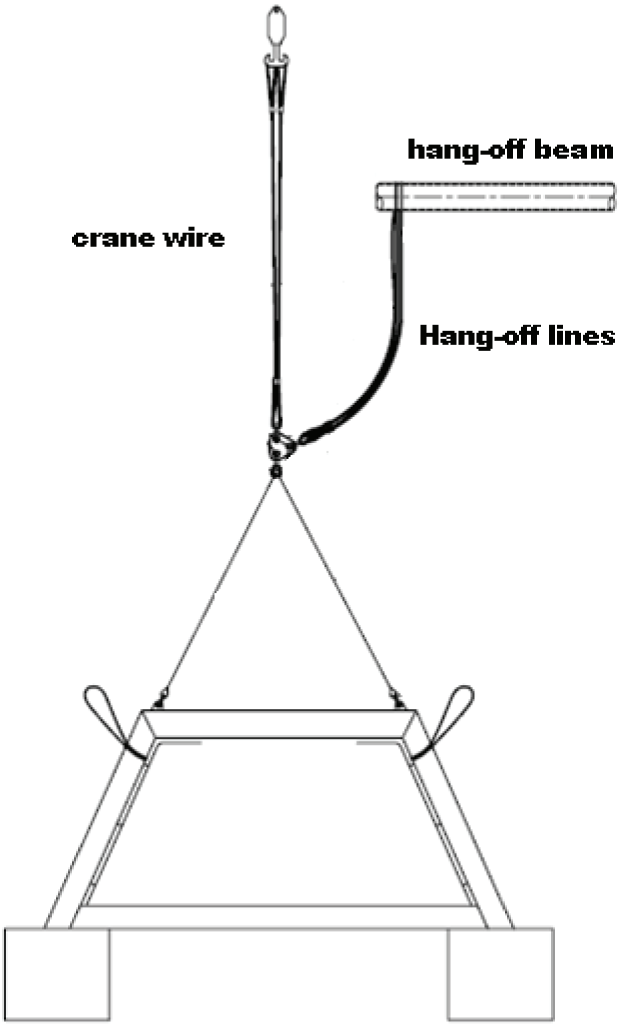


Figure 13 Illustration of structure suspended from the hang-off beam. Reference [32]

The obvious limiting factor of this method is the weight of the structure to be towed. Since the structure is suspended from a hang-off beam over the side of the vessel, the effects of the motions of the vessel will be significant as the hang-off point is far away from the vessels COG. This makes this method more sensitive to the weather than the Subsea 7 method which has a hang-off point close to the vessels COG.

As the installation is done with a crane, we have an active heave compensator which makes the installation criteria better than the Subsea 7 method.

3 Dynamic analysis method

3.1 1-DOF system

We want to analyze a subsurface towing of a module. In order to obtain correct results we need to be consequent with our choices throughout all the steps.

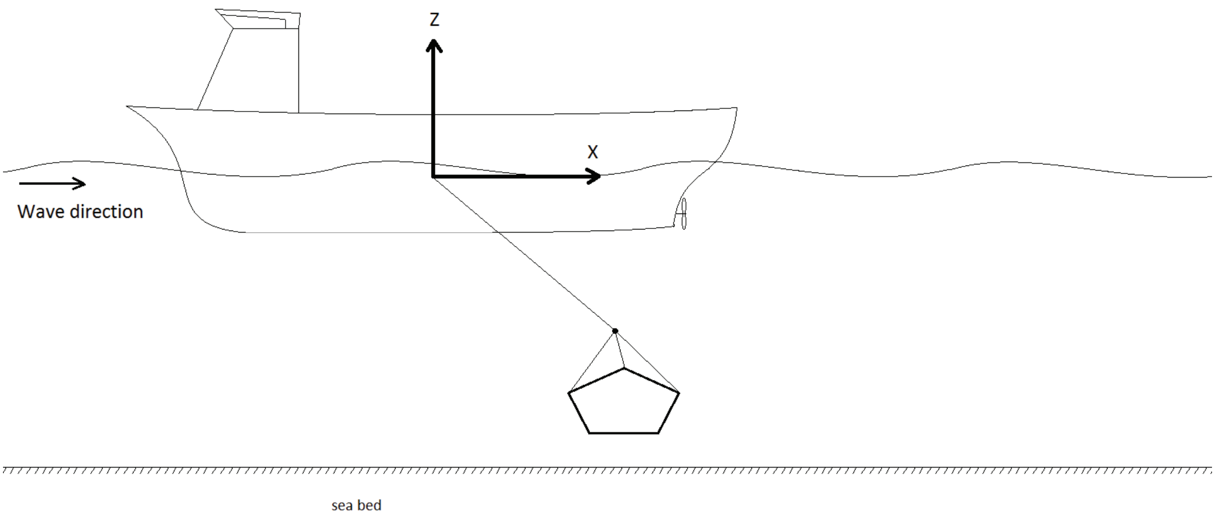


Figure 14 Definition of axes and direction of the waves

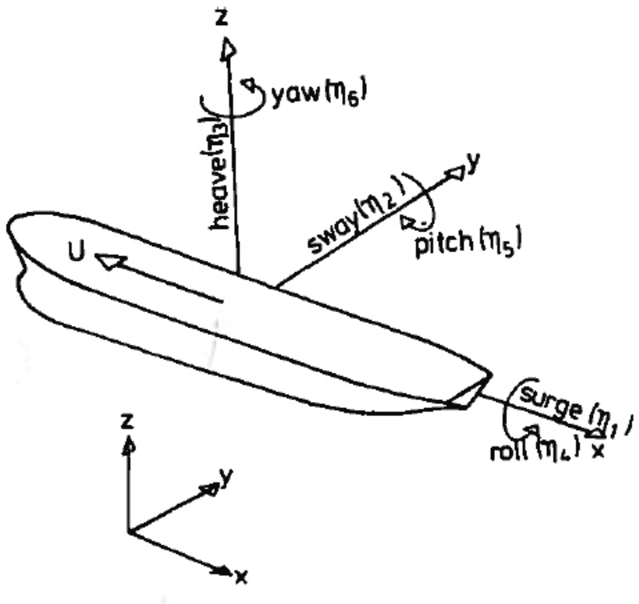


Figure 15 Definition of coordinate system and rigid-body motion modes. Reference [10]

The theory presented in this chapter is implemented in MATLAB to do a 1-DOF wet tow analysis. The simple program for a subsurface towing analysis that was developed in reference [7] has been expanded and improved to be able to do a proper analysis of the Tyrihans project.

3.1.1 Horizontal offset

We consider the static offset of a load hanging in a wire under the influence of current (velocity of the vessel). According to reference [11].

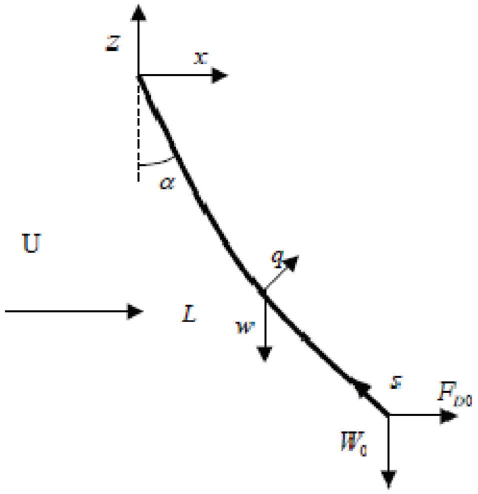


Figure 16 Force equilibrium in rope. Reference [11]

We assume infinite stiffness in axial direction, zero bending stiffness and ignore the elasticity. Using these assumptions, an expression for the horizontal offset and angle to the vertical plane can be derived according reference [11].

The top of the lifting cable is located at the surface (z=0), and the horizontal offset is measured from x=0.

At any point of the cable we have to derive an equation for vertical and horizontal equilibrium.

$$Vertical: T_w(z)\cos\alpha = W_0 + \rho_w g A z_b + mg \cdot s(z) - \int_0^{s(z)} q \sin\alpha ds$$

$$Horizontal: T_w(z)\sin\alpha = F_{D0} + \int_0^{s(z)} q \cos\alpha ds - p_0(z) A \sin\alpha$$

(3.1.1.1 ,3.1.1.2)

$L =$ wire length
$w = mg - \rho_w gA =$ Submerged weight per unit length
$W_0 =$ Submerged weight at the end of the wire
$m =$ Mass pr unit length of the wire
$\rho_w =$ Density of water
$A =$ Crossection area
$\alpha =$ Offset angle of the wire
$F_{D0} =$ Drag force on the body connected to the end of the wire
$q =$ drag force per unit length
$z =$ Vertical position
$P_0(z) =$ Pressure at position z

Since we are interested in the horizontal offset, we consider equation 3.1.1.2. The first component on the right hand side is the drag force. The second component is the integrated effect on the drag force and the third is the correction for missing pressure. Since the Integrated effect on the drag force is small relative to the drag force, this part is ignored in further calculations.

This gives the following equation:

$$T_w(z)\sin\alpha = F_{D0} - p_0(z)A\sin\alpha \quad (3.1.1.3)$$

Introducing the effective tension $T_E = T_w(z) + p_0(z)A$, insert this into equation 3.1.1.3, and rearrange, gives us:

$$(T_w(z) + p_0(z)A)\sin\alpha = F_{D0} = T_E(z)\sin\alpha \quad (3.1.1.4)$$

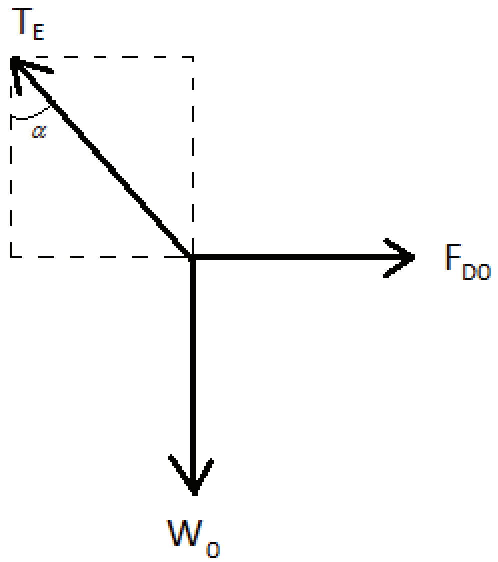


Figure 17 Effective tension

From the figure we see that:

$$T_E = \sqrt{W_0^2 + F_{D0}^2} \qquad (3.1.1.5)$$

We also need an expression for the drag force. Assuming horizontal current (velocity of the vessel) in the positive x-direction, the drag force is:

$$F_{D0} = \frac{1}{2}\rho_0 v^2 C_D A \qquad (3.1.1.6)$$

$v =$ Velocity of the current (vessel)
$C_D =$ Drag coefficient
$A =$ Refference area

Combining equation 3.1.1.4 and 3.1.1.5 gives us an expression for the horizontal offset angle:

$$\sin\alpha = \frac{F_{D0}}{T_E} = \frac{F_{D0}}{\sqrt{W_0^2 + F_{D0}^2}}$$
$$\Downarrow \qquad (3.1.1.7)$$
$$\alpha = \sin^{-1}\left[\frac{F_{D0}}{\sqrt{W_0^2 + F_{D0}^2}}\right]$$

3.1.2 Vertical oscillations of a wire including a mass

In the calculations of horizontal offset we made some simplifications regarding the wire. In this section we will assume no horizontal offset and evaluate the dynamic effect of the wire. Initially the damping is ignored as we will introduce it after obtaining the necessary equations. We simplify the system according to Figure 18.

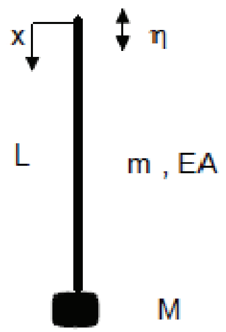


Figure 18 Simplified system. Reference [11]

$L =$ Wire length

$m =$ Homogenous mass distribution pr unit length

$EA =$ Stiffness pr unit length

$M =$ Mass at the bottom end of the wire

$\eta_a =$ Oscillating amplitude at the top end of the wire

$\omega =$ Oscillating frequency at the top end of the wire

$w =$ mass of the wire pr unit length

$V =$ Volume of the wire

$\rho_w =$ Density of water

$\mathrm{A}_L =$ Projected area of the load

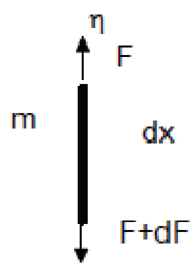


Figure 19 Equilibrium of an element

The deriving of dynamic displacement and force is done according reference [11].

Before we start to derive these equations, the reader should be aware of the following notation: $\dot{A}=\frac{dA}{dt}$ and $A'=\frac{dA}{dx}$

The static elongation of the wire is expressed as:

$$L-L_s=\eta(L)=\frac{L}{EA}\left[\frac{wL}{2}+Mg-\rho_w gV\right] \tag{3.1.2.1}$$

Considering Figure 19, we can obtain an expression for the dynamic equilibrium:

$$\begin{gathered} F+dF=F-wdx+mdx\frac{d^2\eta}{dt^2} \\ \frac{dF_d}{dx}=m\ddot{\eta} \end{gathered} \tag{3.1.2.2}$$

The stress strain relation for the element:

$$EA\frac{d\eta}{dx}=F \tag{3.1.2.3}$$

We insert the stress strain relation (3.1.2.3) into the expression for dynamic equilibrium (3.1.2.2) and get:

$$EA\frac{d^2\eta}{dx^2}=m\ddot{\eta} \tag{3.1.2.4}$$

Equation 3.1.2.4 is to be valid for all x and t.

The static elongation of the wire is now disregarded and η will denote the dynamic elongation.

To solve this equation we use separation of variables.

$$\eta(x,t) = X(x)T(t)$$
$$\Downarrow$$
$$\frac{\partial^2 \eta}{\partial x^2} = X''(x)T(t) \qquad (3.1.2.5)$$
$$\frac{\partial^2 \eta}{\partial t^2} = X(x)\ddot{T}(t)$$

The relations in 3.1.2.5 combined with 3.1.2.4 gives:

$$\frac{X''}{X} = \frac{m}{EA}\frac{\ddot{T}}{T} = \text{constant} \qquad (3.1.2.6)$$

At x=0 we assume top point excitation:

$$\eta(0,t) = \eta_a \cos(\omega t) \qquad (3.1.2.7)$$

The use of this boundary condition gives

$$\frac{X''}{X} = \frac{m}{EA}\frac{\ddot{T}}{T} = -\omega^2 \frac{m}{EA} = -k^2 \qquad (3.1.2.8)$$

We can now assume that X(x) and T(t) can be written as follows:

$$X(x) = A\cos(kx) + B\sin(kx)$$
$$T(t) = \cos(wt) \qquad (3.1.2.9\ ,\ 3.1.2.10)$$

To find both constants in the assumed equation for X(x), we need a second boundary condition:

The dynamic tension at the lower end of the wire must be balanced by the inertia force:

$$M\ddot{\eta}(L,t) = -EA\eta'(L,t) \qquad (3.1.2.11)$$

Using both boundary conditions 3.1.2.7 and 3.1.2.11, we get expressions for the constants A and B:

$$A = \eta_a$$
$$B = \eta_a \frac{1 + \frac{m}{kM}\tan(kL)}{\frac{m}{kM} - \tan(kL)} \quad (3.1.2.12)$$

Now the equations 3.1.2.9 and 3.1.2.10 is complete, and we insert them into equations 3.1.2.5 and 3.1.2.3 to get the dynamic displacement and the force on the wire respectively.

$$\eta(x,t) = \eta_a \left[\cos(kx) + \left(\frac{1 + \frac{m}{kM}\tan(kL)}{\frac{m}{kM} - \tan(kL)} \right) \sin(kx) \right] \cos(\omega t) \quad (3.1.2.13)$$

$$F(x,t) = EAk\eta_a \left[-\sin(kx) + \left(\frac{1 + \frac{m}{kM}\tan(kL)}{\frac{m}{kM} - \tan(kL)} \right) \cos(kx) \right] \cos(\omega t) \quad (3.1.2.14)$$

3.1.2.1 Inclusion of damping

Now that we have derived equations for the dynamic displacement and the force on the wire, we want to include the damping.

To do this we have to include a velocity dependent force, where C is the damping force per unit length.

After including the velocity dependent force, the equilibrium equation becomes:

$$F + dF = F - wdx + mdx\frac{d^2\eta}{dt^2} + c\frac{d\eta}{dt}dx$$
$$\frac{dF_d}{dx} = m\ddot{\eta} + c\dot{\eta} \quad (3.1.2.15)$$

Again we have to use the technique of separation of variables:

$$\frac{X''}{X} = \frac{m}{EA}\frac{\ddot{T}}{T} + \frac{c}{EA}\frac{\dot{T}}{T} = \text{constant} \quad (3.1.2.16)$$

Assume that T is harmonic and get the relations:

$$
\begin{aligned}
T &= T_a \cos(\omega t) \\
\dot{T} &= -\omega T_a \sin(\omega t) = -i\omega T \\
\ddot{T} &= -\omega^2 T
\end{aligned}
\tag{3.1.2.17}
$$

$$
\frac{X''}{X} = \frac{m}{EA}\frac{\ddot{T}}{T} + \frac{c}{EA}\frac{\dot{T}}{T} = \frac{m_c}{EA}\frac{\ddot{T}}{T} = \text{constant}
\tag{3.1.2.18}
$$

We have now introduced a linear damping by using an equivalent mass of the load and wire.

$$
\begin{aligned}
m_c &= m + i\frac{c_e}{\omega} \\
M_c &= M + i\frac{C_e}{\omega}
\end{aligned}
\tag{3.1.2.19}
$$

If the imaginary parts of the complex masses are small compared to masses, the force and displacement won't change significantly except at resonance.

Using the linear damping from the masses, we observe that the complex mass is reduced as the frequency increases.

Since the damping is in water, most of the damping will be of quadratic nature.

We can now express the damping force as:

$$
F_D = C_v |\dot{\eta}| \dot{\eta}
\tag{3.1.2.20}
$$

If η is harmonic with amplitude η_a, the linear damping coefficient is given by:

$$
C_e = \frac{8\omega}{3\pi} C_v \eta_a
\tag{3.1.2.21}
$$

Now we insert the damping coefficient into the expressions for the complex masses (3.2.2.19)

$$
\begin{aligned}
m_c &= m + i\frac{8}{3\pi} c_v \eta_a(x) \\
M_c &= M + i\frac{8}{3\pi} C_v \eta_a(L)
\end{aligned}
\tag{3.1.2.22}
$$

c_v and C_v are the estimated drag coefficients of the cable and template respectively, and is assumed to be:

$$c_v = \frac{1}{2}\rho C_{d\_cable} 2\pi r$$
$$C_v = \frac{1}{2}\rho C_{d\_template} A_L \qquad (3.1.2.23)$$

The complex masses (3.1.2.19) are inserted into the equations for dynamic displacement and the force on the wire (3.1.2.13, 3.1.2.14) and assume small damping.

This gives the equations for dynamic displacement ant the force on the wire with inclusion of a small damping:

$$\eta(x,t) = \eta_a \left[\cos(kx) + \left(\frac{1 + \frac{m}{kM}\tan(kL)}{\frac{\left(m + i\frac{\frac{8\omega}{3\pi}c_v\eta_a}{\omega}\right)}{kM}\left(1 - i\frac{\frac{8\omega}{3\pi}C_v\eta_a}{M}\right) - \tan(kL)} \right)\sin(kx) \right]\cos(\omega t) \qquad (3.1.2.24)$$

$$F(x,t) = EAk\eta_a \left[-\sin(kx) + \left(\frac{1 + \frac{m}{kM}\tan(kL)}{\frac{\left(m + i\frac{\frac{8\omega}{3\pi}c_v\eta_a}{\omega}\right)}{kM}\left(1 - i\frac{\frac{8\omega}{3\pi}C_v\eta_a}{\omega M}\right) - \tan(kL)} \right)\cos(kx) \right]\cos(\omega t) \qquad (3.1.2.25)$$

3.1.3 Wave spectrum

The wave spectrum describes mathematically the distribution of wave energy with frequency and direction. There are several wave spectrums, and each is based on the theory of stochastic processes.

In the Matlab program used for 1-DOF analysis, it is possible to use both the Pierson-Moskowitz spectrum and JONSWAP spectrum, as well as regular waves.

In the choice of a spectrum, several factors have to be considered.

Does the spectrum give a reasonable estimate for the spectrum in the frequency interval? Is the spectrum mathematically satisfying? And is the spectrum given by a reasonable number of parameters?

3.1.3.1 Pierson-Moskowitz spectrum

The Pierson-Moskowitz spectrum (PM-spectrum) assumes that if wind blows steadily for a long time over a large area, then the waves would eventually reach a point of equilibrium with the wind (fully developed sea).

This spectrum evolved from measurements in the north Atlantic from Moskowitz (1964).

The PM-spectrum can be represented by the following equation:

$$S(\omega) = H_S^3 T_P \frac{0,11}{2\pi}\left(\frac{\omega T_P}{2\pi}\right)^{-5} \exp\left[-0,44\left(\frac{\omega T_P}{2\pi}\right)^{-4}\right] \quad (3.1.3.1)$$

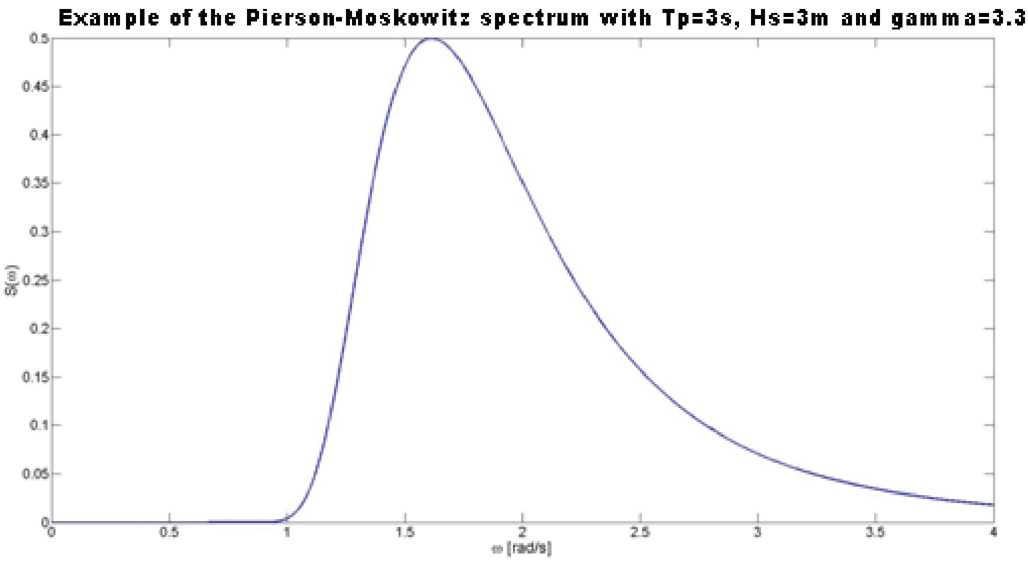


Figure 20 Example of the Pierson-Moskowitz spectrum

3.1.3.2 Joint North Sea Wave Project Spectrum

The Joint North Sea Wave Project Spectrum (JONSWAP) is a fetch-limited version of the Pierson-Moskowitz spectrum. However the JONSWAP spectrum is never fully developed and may continue to develop for a long time due to non-linear wave-interactions.

This spectrum evolved from measurements in the south-east parts of the northern sea in 1968-1969.

The JONSWAP spectrum can be represented by the following equation:

$$S(\omega)=\frac{5}{32\pi}H_S^2T_P\left(\frac{\omega_P}{\omega}\right)^{-5}\exp\left[-\frac{5}{4}\left(\frac{\omega_P}{\omega}\right)^4\right](1-0,287\ln\gamma)\,\gamma^{\exp\left[\frac{-\left(\frac{\omega}{\omega_P}-1\right)^2}{2\sigma^2}\right]}$$

$$\sigma=\begin{matrix}0,07 & if & \omega\le\omega_P\\ 0,09 & if & \omega\succ\omega_P\end{matrix} \qquad (3.1.3.2)$$

$$\omega_P=\frac{2\pi}{T_P}$$

Where ω_p is the peak frequency and γ is the peak enhancement factor.

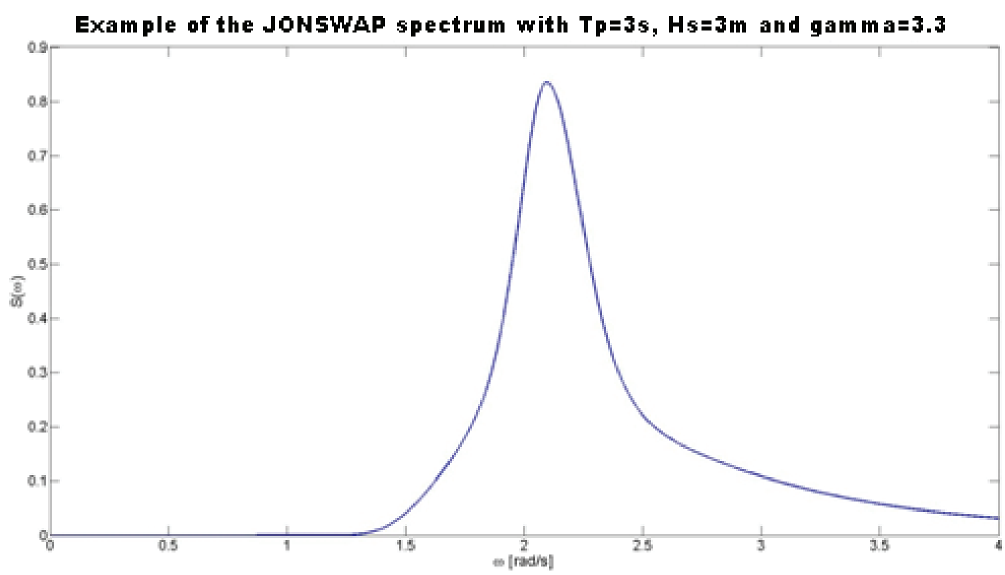


Figure 21 Example of the JONSWAP spectrum

When the wave spectrum is known, some statistical properties can be calculated:

Zero crossing period	$T_{m0z} = 2\pi\sqrt{\frac{m_0}{m_2}}$	
Mean period..	$T_{m0e} = 2\pi\left(\frac{m_0}{m_1}\right)$	
Mean period between wave crests............................	$T_{m24} = 2\pi\sqrt{\frac{m_2}{m_4}}$	
Calculated significant wave height	$H_{m0} = 4\sqrt{m_0}$	
Highest wave in a given period	$H_{\max} = H_{m0}\sqrt{\frac{\ln N}{2}}$	$N = \frac{\text{Duration of the sea state}}{T_z}$
Highest wave with a possibility of 1/N.....................	$H_{\frac{1}{N}} = H_{m0}\sqrt{\frac{\ln N}{2}}$	
Mean zero crossing period..	$T_z = \frac{\pi}{2}\frac{H_{m0}}{\sqrt{m_2}}$	

Table 1 Statistical properties of the wave spectrum

3.1.4 Surface elevation

We will consider a 2-dimensional problem with waves propagating in the positive x-direction. According to reference [12], the surface elevation can be written as:

$$\zeta(x,t) = \sum_{n=1}^{N} \zeta_{An} \cos(\omega_n t - k_n x + \varepsilon_n) \quad (3.1.4.1)$$

Where ζ_{An} is the wave amplitude, ω_n is the wave frequency, k_n is the wave number and ε_n is the random phase angle of wave component n. N is the total number of wave components.

We assume deep water and use the relation between the wave frequency and the wave number:

$$\omega_n^2 = k_n g \quad (3.1.4.2)$$

The energy for linear waves is given for each wave component n:

$$E_n = \frac{1}{2} \rho_w g \zeta_{An} \quad (3.1.4.3)$$

So the total energy is the sum of N harmonic waves:

$$\frac{E}{\rho g} = \sum_{n=1}^{N} \frac{1}{2} \zeta_{An}(\omega_n) \quad (3.1.4.4)$$

We now introduce the wave spectrum.

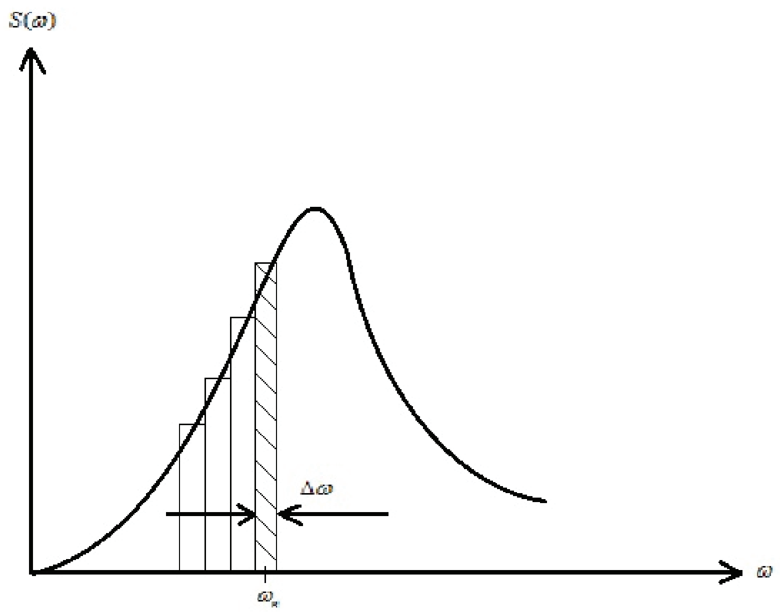


Figure 22 Sketch of a wave spectrum

The area inside a small frequency interval equals the energy to all components inside this interval.

$$\frac{1}{2}\zeta_{An}^2 = S(\omega_n)\Delta\omega \tag{3.1.4.5}$$

Combining equations 3.1.3.4 and 3.1.3.5 gives us a new expression for the total energy:

$$\frac{E}{\rho g} = \sum_{n=1}^{N}\frac{1}{2}\zeta_{An}(\omega_n) = \sum_{n=1}^{N} S(\omega_n)\Delta\omega \tag{3.1.4.6}$$

Combining equations 3.1.4.1 and equation 3.1.4.5 gives us an expression for the surface elevation:

$$\zeta_{An} = \sum_{n=1}^{N}\sqrt{2S(\omega_n)\Delta\omega}\cos(\omega_n t - k_n x + \varepsilon_n) \tag{3.1.4.7}$$

3.1.5 Added mass in heave for the vessel

The hydrodynamic mass (added mass) is the force that acts from the water on the vessel and that is in anti-phase with the acceleration when the vessel is forced to a sinus-oscillation. The transfer function of MSV Botnica is given and the effect of the added mass is included and.

3.1.6 Damping in heave for the vessel

The hydrodynamic damping is the force that acts from the water on the vessel and that is in anti-phase with the velocity when the vessel is forced to a sinus-oscillation.

Equally as the added mass, the damping is included in the transfer function that is given.

3.1.7 Transfer function

The transfer function is the response amplitude per unit wave amplitude and is a mathematical representation, in terms of spatial or temporal frequency of the relation between the input and output of a linear time-invariant system.

In this thesis the transfer function of the installation vessel MSV Botnica is given. To get a better understanding, the derivation of the transfer function of a barge is conducted.

We can use the transfer function to determine the time history of the barge. In order to derive an expression for the transfer function, we need the dimensions of the barge.

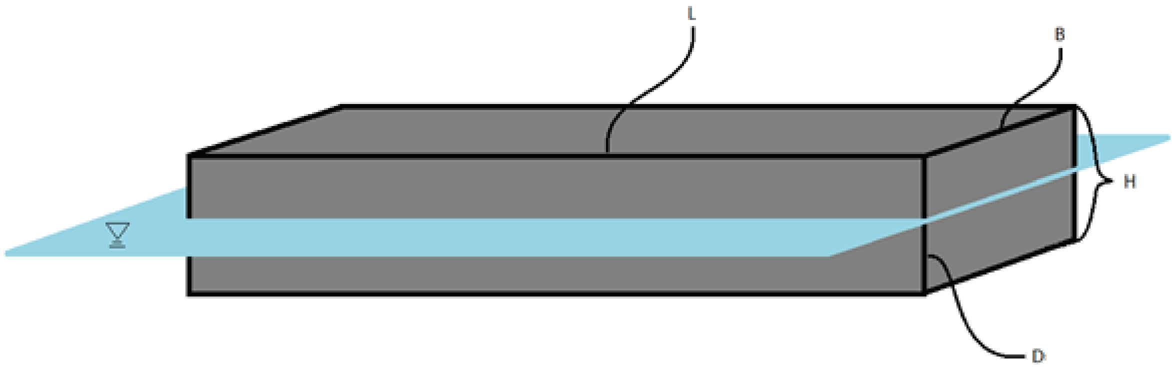


Figure 23 Dimensions of the barge

The heave equation of the center of gravity of the barge can be written as:

$$(M + A_{33})\ddot{\eta}_3 + B_{33}\dot{\eta}_3 + C_{33}\eta_3 = F_3 \quad (3.1.7.1)$$

We assume solution as follows:

$$\begin{aligned} \eta_3 &= \eta_{3A}\sin(\omega t) \\ \dot{\eta}_3 &= \eta_{3A}\omega\cos(\omega t) \\ \ddot{\eta}_3 &= -\eta_{3A}\omega^2\sin(\omega t) \end{aligned} \quad (3.1.7.2)$$

To get a solution for the heave response of the barge, we insert equation 3.1.7.2 into equation 3.1.7.1 and solve for η_3 :

$$\eta_3 = \frac{F_3}{\sqrt{(C_{33} - (M + A_{33})\omega^2)^2 + (B_{33}\omega)^2}} \quad (3.1.7.3)$$

An expression for the vertical excitation force F_3 is according to Reference [10]:

$$F_3 = \left(\rho g \zeta_a B \exp[-kD] - g A_{33}^{2D} k \zeta_a \exp\left[\frac{-kD}{2}\right] \right) \frac{2}{k} \sin\left(\frac{kL}{2}\right) \sin(\omega t) \tag{3.1.7.4}$$

Inserting equation 3.1.7.4 into equation 3.1.7.3 gives the equation for the heave response of the barge:

$$\eta_3 = \frac{\left(\rho g \zeta_a B \exp[-kD] - g A_{33}^{2D} k \zeta_a \exp\left[\frac{-kD}{2}\right] \right) \frac{2}{k} \sin\left(\frac{kL}{2}\right) \sin(\omega t)}{\sqrt{(C_{33} - (M + A_{33})\omega^2)^2 + (B_{33}\omega)^2}} \tag{3.1.7.5}$$

The formulas for A_{33}, B_{33}, and C_{33} is according to reference [10]:

$$\begin{aligned} C_{33} &= \rho g A_w \\ A_{33} &= \int_L A_{33}^{2D}(x) dx \\ B_{33} &= \int_L B_{33}^{2D}(x) dx \end{aligned} \tag{3.1.7.6}$$

Now we want an expression for the transfer function.

$$H(\omega)_3 = \left| \frac{\eta_3}{\zeta_a} \right| \tag{3.1.7.7}$$

Combining equation 3.1.7.5 and 3.1.7.7 gives the transfer function of the barge:

$$H(\omega)_3 = \left| \frac{\eta_3}{\zeta_a} \right| = \frac{\left(\rho g B \exp[-kD] - g A_{33}^{2D} k \exp\left[\frac{-kD}{2}\right] \right) \frac{2}{k} \sin\left(\frac{kL}{2}\right)}{\sqrt{(C_{33} - (M + A_{33})\omega^2)^2 + (B_{33}\omega)^2}} \tag{3.1.7.8}$$

3.1.8 Vessel response

To get the heave response of the vessel we need to combine the wave spectrum (chapter 3.1.3) with the transfer function (chapter 3.1.7). Reorganizing equation 3.1.7.7 gives:

$$\eta_{3An}^2 = |H(\omega_n)|^2 \zeta_{an}^2 \tag{3.1.8.1}$$

We recall the energy relationship from chapter 3.1.4, equation 3.1.4.5.

$$\frac{1}{2}\zeta_{An}^2 = S(\omega_n)\Delta\omega \rightarrow \zeta_{An}^2 = 2S(\omega_n)\Delta\omega \tag{3.1.8.2}$$

The heave response of the vessel is given by:

$$\eta_3 = \sum_{n=1}^{N} \eta_{3An}\cos(\omega_n t + \underbrace{\varepsilon_{ship}}_{\substack{\text{The phaseangel for the ship is ignored since only} \\ \text{the relative magnitude of heave motion is of interest.}}} + \varepsilon_{sea})$$

$$\Downarrow$$

$$\eta_3 = \sum_{n=1}^{N} \eta_{3An}\cos(\omega_n t + \varepsilon_{sea}) \tag{3.1.8.3}$$

Combining the equation for the heave response (3.1.8.3) and equations 3.1.8.1 and 3.1.8.2 gives a new equation for the heave response including the transfer function and the spectrum of choice:

$$\eta_3 = \sum_{n=1}^{N} \sqrt{|H(\omega_n)|^2 2S(\omega_n)\Delta\omega}\cos(\omega_n t + \varepsilon_{sea}) \tag{3.1.8.4}$$

3.1.9 Template response

If the vessel experiences irregular sea described by a wave spectrum, and the template experiences forced displacements, a semi discrete equation of motion has to be solved numerically according to reference [13]:

$$M\ddot{d} + C\dot{d} + Kd = F$$

$$\begin{aligned} &M - \text{ Symmetric mass matrix} \\ &C - \text{ Symmetric viscous damping matrix} \\ &K - \text{ Symmetric stiffness matrix} \\ &\ddot{d}, \dot{d}, d - \text{ Acceleration, velocity and displacement vectors} \end{aligned} \tag{3.1.9.1}$$

We want to find a displacement $d = d(t)$ that satisfies equation 3.1.9.1. This is done by using Newmark beta method.

The first step is to establish equations for approximations of the displacement, velocity and acceleration (d_n, v_n and a_n). The initial approximation values are assumed to be known, and we need equations for the next step of the iteration process (Newmark family):

$$\begin{aligned} &Ma_{n+1} + Cv_{n+1} + Kd_{n+1} = F_{n+1} \\ &d_{n+1} = d_n + \Delta t v_n + \frac{\Delta t^2}{2}\left[(1-2\beta)a_n + 2\beta a_{n+1}\right] \\ &v_{n+1} = v_n + \Delta t\left[(1-\gamma)a_n + \gamma a_{n+1}\right] \end{aligned} \tag{3.1.9.2}$$

The second step is to define predictors:

$$\begin{aligned} &\tilde{d}_{n+1} = d_n + \Delta t v_n + \frac{\Delta t^2}{2}\left[(1-2\beta)a_n\right] \\ &\tilde{v}_{n+1} = v_n + \Delta t(1-\gamma)a_n \end{aligned} \tag{3.1.9.3}$$

Combining the predictors (equation 3.1.9.3) and the equations from the Newmark family (3.1.9.2) yields:

$$\begin{aligned} &a_{n+1} = \frac{F_{n+1} - Cv_{n+1} - Kd_{n+1}}{M} \\ &d_{n+1} = \tilde{d}_{n+1} + \beta \Delta t^2 a_{n+1} \\ &v_{n+1} = \tilde{v}_{n+1} + \gamma \Delta t a_{n+1} \end{aligned} \tag{3.2.9.4}$$

The stiffness, damping and mass (including added mass) of the template is assumed to be known.

The values of β and γ is decided by the use of table 9.1.1 p. 493 Hughes 2008, reference [13].

Method	Type	β	γ	Stability condition(2)	Order of accuracy(3)
Average acceleration	Implicit	1/4	1/2	Unconditional	2
Linear acceleration	Implicit	1/6	1/2	$\Omega_{crit} = 2\sqrt{3} \cong 3.464$	2
Fox-Godwin (royal road)	Implicit	1/12	1/2	$\Omega_{crit} = \sqrt{6} \cong 2.449$	2
Central difference	Explicit(1)	0	1/2	$\Omega_{crit} = 2$	2

Table 2 Properties of well-known members of the Newmark family. Reference [13].

Notes to the table:

1: Strictly speaking, M and C need to be diagonal for the central difference method to be explicit.

2. Stability is based upon the undamped case, in which $\xi = 0$.

3. Second-order accuracy is achieved if and only if γ=1/2.

The parameters β and γ determines the stability and accuracy respectively of the method.

Determination of β is seen from Table 2.

The choice of γ decides if the method has an artificial damping.

$$\gamma > \frac{1}{2} \rightarrow \text{Positive artificial damping. When the time step increases, the amplitude decreases}$$

$$\gamma < \frac{1}{2} \rightarrow \text{Negative artificial damping. When the time step increases, the amplitude increases}$$

$$\gamma = \frac{1}{2} \rightarrow \text{No Artificial damping}$$

3.2 Multiple degree of freedom system

There are several methods to use in a dynamic analysis with multiple degrees of freedom. Some of the methods available will be discussed in this chapter.

The equation of motion for a system with multiple degrees of freedom is given by:

$$M(p,a)\ddot{r}+C(p,v)\dot{r}+K(p)r=Q(p,v,t)$$

$M =$ Mass matrix
$C =$ Damping matrix
$K =$ Stiffness matrix (3.2.1)
$Q(t) =$ External load matrix as a function of time
$p,v,a =$ Position-,velocity- and acceleration respectively
$r,\dot{r},\ddot{r} =$ Position-,velocity- and acceleration-vectors respectively
$t =$ simulation time

The solution of the equation of motion (equation 3.2.1) can be obtained in two different forms.

Solution in the time space

The solution is given as a function of time, which is natural if the load is given as a function of time.

Solution in the frequency space

The load can be given as an infinite sum of harmonic components which mathematically can be obtained by using Fourier-transformation. To solve the motion of equation in the frequency space means to solve equation 3.2.1 for a harmonic load with variation of the frequency.

If the solution method in the time- or frequency-space is given for the connected set of equations (equation 3.2.1) we have a direct method. It is also possible to transform equation 3.2.1 into an unconnected system and solve these unconnected equations separately (indirect method).

Reference [14].

3.2.1 Mass-matrix

In this chapter two different methods of establishing the mass-matrix will be discussed, the lumped- and the consistent mass-matrix.

3.2.1.1 Consistent mass-matrix

According to reference [14] the consistent mass-matrix for an element is given by:

$$m_i = \int_{V_i} \rho N^T N dv \tag{3.2.1.1}$$

$\rho =$ Density of the material
$N =$ Interpolation polynomial
$v =$ Dof of the element
$u =$ Displacement of the element
$r =$ Dof of the structure
$T =$ Kinetic energy

The connection between the displacement form of the element and the nodes of the element is given by the use of interpolation polynomials.

$$\begin{aligned} u &= N v_i \\ \dot{u} &= N \dot{v}_i \end{aligned} \tag{3.2.1.2}$$

This gives an expression for the mass-matrix for the whole structure:

$$M = \sum_i r v_i^T m_i v_i r^T \tag{3.2.1.3}$$

The elements of the mass-matrix can be interpreted as:

$m_{ij} =$ Inertia of displacement parameter "i" because of a unit acceleration of displacement parameter "j"

By looking at the expression for the kinetic energy we can find a connection with the mass matrix.

$$T = \frac{1}{2} mass \cdot velocity^2 = \frac{1}{2} \int_V \rho \dot{u}^T \dot{u} dV \tag{3.2.1.4}$$

$$T = \frac{1}{2}\sum_i \dot{v}_i^T \int_{V_i} \rho N^T N dV \dot{v}_i = \frac{1}{2}\sum_i \dot{v}_i{}^T m_i \dot{v}_i$$

$$T = \frac{1}{2}\dot{r}^T (\sum_i r v_i{}^T m_i v_i r^T)\dot{r} = \frac{1}{2}\dot{r}^T M \dot{r} \qquad (3.2.1.5)$$

The consistent mass-matrix connects the kinetic energy to the velocities of the nodes by using interpolation polynomials. It is called consistent when the interpolation polynomial is the same as the one used in the derivation of the stiffness-matrix. This means a correct use of the energy method since the kinetic energy is consistent with the potential energy. Reference [14].

3.2.1.2 Lumped mass-matrix

Orcaflex uses the lumped mass-matrix which is a simplified method where the mass is distributed in the structures nodes. The accuracy of the lumped mass-matrix is not as good as the consisted mass-matrix, but its simplicity gives great advantages regarding calculations. It uses fewer operations in the calculation of eigenvalues, reduction of DOFs and direct integration of dynamic response. This means that this method needs less storage and because of the reduction of number of operations needed, it will be more cost efficient than the consistent mass-matrix.

According to reference [14] we use virtual work to find the lumped mass-matrix for complicated elements.

$$\delta v^T P = \int_V \delta u^T \rho dv = \int_V (N\delta v)^T \rho dv$$
$$\delta v^T P = \delta v^T \int_V N^T \rho dv$$
$$\downarrow$$
$$P = \int_V N^T \rho dv \qquad (3.2.1.6)$$

$P =$ Load vector
$\rho =$ Density of the material
$N =$ Interpolation polynomial
$v =$ Dof of the element
$u =$ Displacement of the element

The diagonal of the mass matrix is given by the corresponding elements in the load vector P. Reference [14].

3.2.2 Stiffness-matrix

To establish a stiffness-matrix for the structure we want to analyze, we need to make a discretisation of the structure to connect the local degrees of freedom with the global degrees of freedom. To illustrate the procedure we look at two elements forming a structure with three degrees of freedom. The example provided is a simple structure; however the procedure is the same for more complex structures with the full six degrees of freedom.

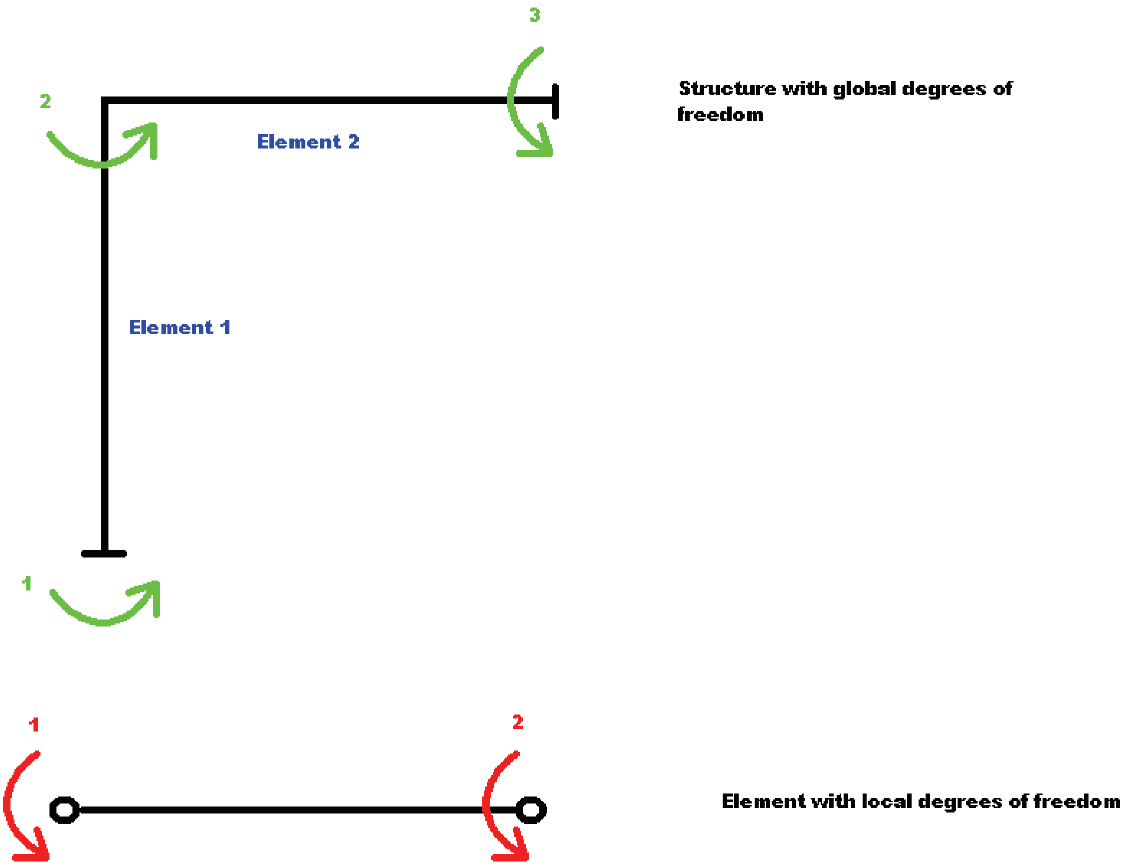


Figure 24 Discretisation of a structure

By creating a "Matrix of Nodal Point Correspondence" (MNPC) which gives a connection between the local and the global degrees of freedom, we can easily form the stiffness matrix.

The MNPC for the structure in Figure 24 is:

Element number	1	2
Local degree of freedom 1	1	2
Local degree of freedom 2	2	3

Table 3 Matrix of Nodal Point Correspondence for the structure in Figure 24.

We assume a known element stiffness matrix of:

$$k_i = \begin{bmatrix} (k_{11})_i & (k_{12})_i \\ (k_{21})_i & (k_{22})_i \end{bmatrix}$$

$(k_{11})_i =$ Stiffness at end "1" because of rotation in end "1" (element i)
$(k_{22})_i =$ Stiffness at end "2" because of rotation in end "2" (element i)
$(k_{12})_i =$ Stiffness at end "1" because of rotation in end "2" (element i)
$(k_{21})_i =$ Stiffness at end "2" because of rotation in end "1" (element i) (3.2.2.1)

By using the MNPC we can establish the stiffness matrix for the structure:

$$K = \begin{bmatrix} (k_{11})_1 & (k_{12})_1 & 0 \\ (k_{21})_1 & (k_{22})_1 + (k_{11})_2 & (k_{12})_2 \\ 0 & (k_{21})_2 & (k_{22})_2 \end{bmatrix} \qquad (3.2.2.2)$$

Reference [14].

3.2.3 Damping-matrix

Similar as for the mass-matrix we can get a diagonal damping matrix. For instance we can have concentrated viscous damping forces connected to degrees of freedom. This presupposes known damping coefficients.

If we have a distributed damping force, a consistent damping-matrix can be established with the same procedure as the consistent mass-matrix, chapter 3.2.1.1.

$$c_i = \int_{V_i} N^T c(x) N dv \tag{3.2.3.1}$$

c_i - Damping matrix of one element
$c(x)$ - Distributed damping force
N - Interpolation polynomial

Often there are uncertainties in deciding the elements in the damping matrix. In these cases the damping matrix can be expressed as a linear combination of the mass- and stiffness-matrix.

If we have a distributed damping force that is proportional with the mass in each point, we get a damping matrix which is proportional with the mass-matrix.

$$C = \alpha_1 M \tag{3.2.3.2}$$

Similarly if the damping is proportional with the stiffness we get proportionality with the stiffness-matrix.

$$C = \alpha_2 K \tag{3.2.3.3}$$

A combination of equation 3.2.3.2 and 3.2.3.3 gives a damping proportional with both the mass-matrix and the stiffness-matrix and is called Rayleigh-damping.

$$C = \alpha_1 M + \alpha_2 K \tag{3.2.3.4}$$

Reference [14].

3.2.4 Modal analysis

The essence of modal super positioning is to transform a set of connected equation to a set of independent differential equations. These solutions give the contribution from each natural oscillation form and the sum of these gives the total solution.

This is done according to reference [14].

Equation 3.2.1 is the equation of motion given with a vector of nodal displacements, r. By assuming a free and undamped oscillation we can solve the eigenvalue problem and obtain the natural oscillation shapes for the structure.

$$
\begin{aligned}
& M\ddot{r} + C\dot{r} + Kr = Q(t) \\
& \qquad \downarrow \qquad\qquad C = Q(t) = 0 \;\; \text{Free and undamped oscillation} \\
& M\ddot{r} + Kr = 0 \\
& \qquad \downarrow \qquad\qquad \text{Assume the solution } r = \phi \sin(\omega t) \\
& -M\phi\omega^2 \sin(\omega t) + K\phi \sin(\omega t) = 0 \\
& \qquad \downarrow \\
& \left[K - \omega^2 M\right]\phi = 0
\end{aligned}
\tag{3.2.4.1}
$$

M = Mass-matrix
C = Damping-matrix
K = Stiffness-matrix
$Q(t)$ = Load vector
r = Vector of nodal displacements
ϕ = Vector of natural oscillation form. ϕ_i = natural oscillation form of global DOF "i"
ω = Oscillating frequency

The natural oscillation shapes are linearly independent and we can therefore express any displacement of the structure as a linear combination of the natural oscillation shapes.

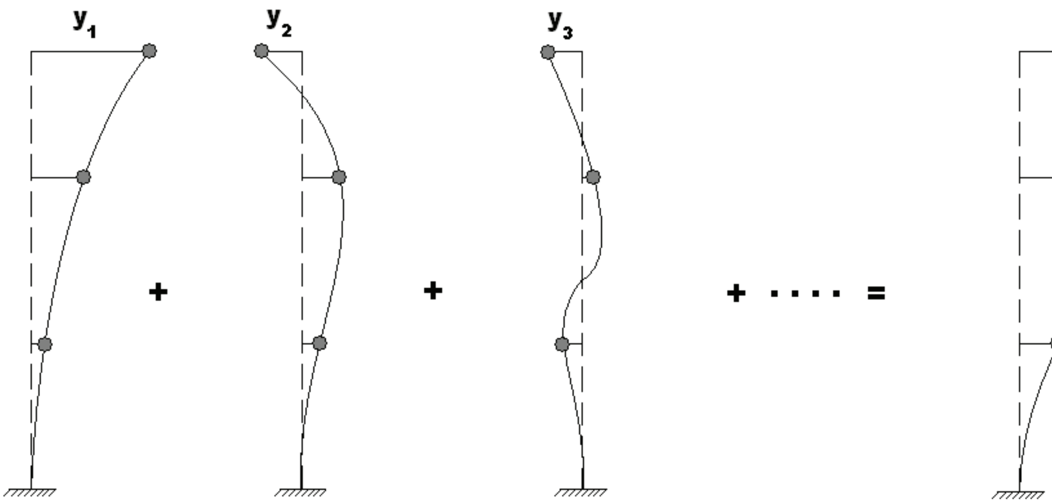


Figure 25 Illustration of a displacement as a linear combination of the natural oscillation shapes. Reference [14].

$$r = \sum_{i=1}^{n} \phi_i y_i(t) = \phi y$$

$$\phi = [\phi_1 \quad \phi_2 \quad \cdots \quad \phi_n]$$

$$y = \begin{bmatrix} y_1(t) \\ y_2(t) \\ \vdots \\ y_n(t) \end{bmatrix}$$ - Vector of displacement amplitudes

$n =$ Number of degrees of freedom (3.2.4.2)

From equation 3.2.4.2 we have the connection between y and r. This is inserted into the equation of motion (equation 3.2.1) and we get:

$$M\phi\ddot{y} + C\phi\dot{y} + K\phi y = Q(t) \qquad (3.2.4.3)$$

Equation 3.2.4.3 is now pre multiplied with ϕ_i^T. We use the orthogonality properties of the natural oscillation forms (equation 3.2.4.4) and also assume that the orthogonality property yields for the damping matrix. (This assumption presupposes for example proportional damping as described in chapter 3.2.3).

$$\phi_i^T M \phi_j = 0 \quad \text{for } i \neq j$$
$$\phi_i^T K \phi_j = 0 \quad \text{for } i \neq j$$
$$\phi_i^T C \phi_j = 0 \quad \text{for } i \neq j \quad \text{(Assuming proportional damping)} \tag{3.2.4.4}$$

This gives the equation:

$$\phi_i^T M \phi_i \ddot{y}_i + \phi_i^T C \phi_i \dot{y}_i + \phi_i^T K \phi_i y_i = \phi_i^T Q(t)$$
$$\downarrow$$
$$\overline{m_i} = \phi_i^T M \phi_i \quad , \quad \overline{c_i} = \phi_i^T C \phi_i \quad , \quad \overline{k_i} = \phi_i^T K \phi_i \quad , \quad \overline{Q_i} = \phi_i^T Q(t)$$
$$\downarrow$$
$$\overline{m_i} \ddot{y}_i + \overline{c_i} \dot{y}_i + \overline{k_i} y_i = \overline{Q_i} \tag{3.2.4.5}$$

This gives a set of “n” independent differential equations (equation 3.2.4.6). These can be solved separately either analytically or numerically for “y” and then with equation 3.2.4.2 we can find the nodal displacements “r”.

$$\begin{bmatrix} \overline{m_1} & & & 0 \\ & \overline{m_2} & & \\ & & \ddots & \\ 0 & & & \overline{m_n} \end{bmatrix} \begin{bmatrix} \ddot{y}_1 \\ \ddot{y}_2 \\ \vdots \\ \ddot{y}_n \end{bmatrix} + \begin{bmatrix} \overline{c_1} & & & 0 \\ & \overline{c_2} & & \\ & & \ddots & \\ 0 & & & \overline{c_n} \end{bmatrix} \begin{bmatrix} \dot{y}_1 \\ \dot{y}_2 \\ \vdots \\ \dot{y}_n \end{bmatrix} + \begin{bmatrix} \overline{k_1} & & & 0 \\ & \overline{k_2} & & \\ & & \ddots & \\ 0 & & & \overline{k_n} \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_3 \end{bmatrix} = \begin{bmatrix} \overline{Q_1} \\ \overline{Q_2} \\ \vdots \\ \overline{Q_n} \end{bmatrix} \tag{3.2.4.6}$$

3.2.5 Frequency-response method

The frequency-response method is based on a harmonic load that is represented as a complex vector.

According to reference [14] we choose a random component of the harmonic load:

$$Q_j(t) = Q_{0j} e^{i(\omega t + \alpha_j)} = Q_{0j}\left\{\cos(\alpha_j) + i\sin(\alpha_j)\right\} e^{i\omega t} = X_j e^{i\omega t}$$

$Q_j(t) =$ Component "j" of the harmonic load
Q_{0j} = Amplitude of component "j" of the harmonic load
ω = Load frequency
α_j = Phase angle for component "j" of the harmonic load
X_j = Complex load vector (3.2.5.1)

We recall the equation of motion (equation 3.2.1) and insert equation 3.2.5.1. The result is an equation of motion with a solution in the frequency space.

$$M\ddot{r} + C\dot{r} + Kr = Xe^{i\omega t} \quad (3.2.5.2)$$

Assuming the steady state solution $r = xe^{i\omega t}$ and inserting this into equation 3.2.5.2 yields:

$$-M\omega^2 xe^{i\omega t} + iC\omega xe^{i\omega t} + Kxe^{i\omega t} = Xe^{i\omega t}$$

$$\downarrow$$

$$\cancel{e^{i\omega t}}\left[-M\omega^2 + iC\omega + K\right]x = X\cancel{e^{i\omega t}}$$

$$\downarrow$$

$$\left[-M\omega^2 + iC\omega + K\right]x = X \quad (3.2.5.3)$$

This system of complex equations can be solved for given load frequencies. The solution of these equations will not be discussed in this thesis, but will result in a vector with a real and a complex component:

$$x_j = r_{0j}\left[\cos(\theta_j) + i\sin(\theta_j)\right]$$
$$x_j = r_{0j}\cos(\theta_j) + ir_{0j}\sin(\theta_j) = x_{Rj} + ix_{Ij}$$

$$x_{Rj} \text{ - Real part} \quad , \quad x_{Ij} \text{ - Complex part} \tag{3.2.5.4}$$

With amplitude and phase angle given by:

$$r_{0j} = \sqrt{x_{Rj}^2 + x_{Ij}^2} \qquad \text{- Amplitude}$$
$$\theta_j = \tan^{-1}\left(\frac{x_{Ij}}{x_{Rj}}\right) \quad \text{- Phase angle} \tag{3.2.5.5}$$

We now introduce the complex frequency response function:

$$H(\omega) = (-M\omega^2 + iC\omega + K)^{-1} \tag{3.2.5.6}$$

This is used to rewrite the solution provided in equation 3.2.5.3:

$$x(\omega) = H(\omega)X(\omega) \tag{3.2.5.7}$$

The elements in the complex frequency response function, $H_{ij}(\omega)$, can be interpreted as the response of degree of freedom “i” because of a harmonic load with a unit amplitude in degree of freedom “j”.

The procedure can be repeated for all frequencies of interest.

Reference [14].

3.2.6 Impulse-response method

The impulse response method divides a random load history into small load impulses. If the solution of these impulses is known, the response for a linear system can be found by super positioning the impulses. Load case "i" is a unit load impulse in degree of freedom "i".

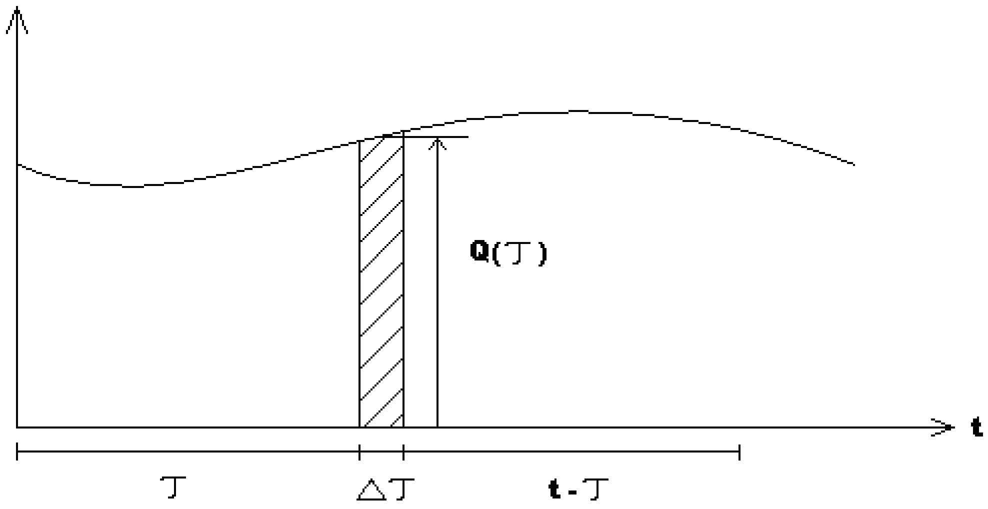


Figure 26 Illustration of a load impulse. Reference [14]

The unit impulse is defined by Dirac's delta function:

$$\delta(t-\tau)=\begin{cases} 0 \quad if \quad t\neq\tau \\ \rightarrow\infty \, if \quad t=\tau \end{cases} \tag{3.2.6.1}$$

The load function with a random load impulse can be written as:

$$Q(t)=I\delta(t-\tau)$$

$I =$ Load impulse
$Q(t) =$ Load function (3.2.6.2)

If the system is subject to a load equivalent to a unit impulse at the time $\tau=0$ we get a combination of the equation of motion (equation 3.2.1) and equation 3.2.6.2.

$$M\ddot{u}+C\dot{u}+Ku=Q(t)=I\delta(t) \tag{3.2.6.3}$$

The response of this system is called the impulse response function and is denoted $h(t)$.

$$u(t)=h(t) \tag{3.2.6.4}$$

This gives the equation of motion with the impulse response function as:

$$M\ddot{h} + C\dot{h} + Kh = I\delta(t) \tag{3.2.6.5}$$

The matrix $h(t)$ is symmetric and can be interpreted as:

h_{ij} = Response in degree of freedom "i" because of a unit impulse
in degree of freedom "j".

Expressing equation 3.2.6.4 with the Dirac delta function gives the response of a unit load impulse at the time “t”:

$$u_{ij}(t) = Ih_{ij}(t-\tau)$$
$$\downarrow \quad I = Q(\tau)\Delta\tau$$
$$u_{ij}(t) = Q(\tau)\Delta\tau h_{ij}(t-\tau) \tag{3.2.6.6}$$

By summing equation 3.2.6.6 for all impulses we can express the response of degree of freedom “i” as an integral with respect to the time.

$$r_i(t) = \int_0^t \sum_{j=1}^{n} h_{ij}(t-\tau)Q_j(\tau)d\tau$$
$$\downarrow$$
$$r(t) = \int_0^t h(t-\tau)Q(\tau)d\tau \tag{3.2.6.7}$$

Reference [14].

3.2.7 Numerical integration

The equation of motion (equation 3.2.1) is an initial value problem where the solution is decided by the initial displacement, velocity and acceleration of each degree of freedom. The period of interest is divided into "h" equal time steps. The initial conditions of a time step are known and the solution at the end of the time step can be decided by assuming the form of the motion. Since the solution is approximated the accuracy of this method increases as the size of the time steps decreases.

The methods based on numerical integration finds a velocity and a displacement for each new time step by integrating the acceleration two times.

$$\dot{u}_{k+1} = \dot{u}_k + \int_0^h \ddot{u}(t)dt$$

$$u_{k+1} = u_k + \int_0^h \dot{u}(t)dt$$

$$\ddot{u}(t) = M^T \left\{ Q(t) - C\dot{u}(t) - Ku(t) \right\} \quad \text{- Equation of motion solved for the acceleration}$$

$\ddot{u}_k, \dot{u}_k, u_k$ = Acceleration, velocity, displacement respectively at timestep k
h = Size of the time step
$\ddot{u}(t)$ = Assumed shape of the acceleration (3.2.7.1)

The difference between the methods is the assumption of how the acceleration is modeled. Orcaflex implements two complementary dynamic integration schemes, explicit and implicit. The explicit method is the forward Euler integration which assumes a constant acceleration. The implicit integration method used is the generalized-α Integration. As both these methods are generalizations of the Newmark beta method, detailed description will not be given. Reference [14].

3.2.7.1 Newmark-beta method

The equation of motion for the system is as described in equation 3.2.1:

$$M\ddot{r} + C\dot{r} + Kr = Q(t) \tag{3.2.7.2}$$

The equations from the Newmark-beta method (eq. 3.2.7.3, reference [14]) for velocity and displacement at time step "k+1" can be rearranged to give an expression for the acceleration and speed at the time step "k+1", expressed with the displacement.

$$\begin{aligned}
\dot{r}_{k+1} &= \dot{r}_k + (1-\gamma)h\ddot{r}_k + \gamma h\ddot{r}_{k+1} \\
r_{k+1} &= r_k + h\dot{r}_k + \left(\frac{1}{2}-\beta\right)h^2\ddot{r}_k + \beta h^2\ddot{r}_{k+1} \\
&\downarrow \\
\ddot{r}_{k+1} &= \frac{1}{\beta h^2}r_{k+1} - \frac{1}{\beta h^2}r_k - \frac{1}{\beta h}\dot{r}_k - \left(\frac{1}{2\beta}-1\right)\ddot{r}_k \\
\dot{r}_{k+1} &= \frac{\gamma}{\beta h}r_{k+1} - \frac{\gamma}{\beta h}r_k - \left(\frac{\gamma}{\beta}-1\right)\dot{r}_k - \left(\frac{\gamma}{2\beta}-1\right)h\ddot{r}_k
\end{aligned} \tag{3.2.7.3}$$

These equations (3.2.7.3) are inserted into the equation of motion (3.2.7.2):

$$M\ddot{r}_{k+1} + C\dot{r}_{k+1} + Kr_{k+1} = Q_{k+1} \tag{3.2.7.4}$$

To summarize we now have an equation of motion for time step "k+1". We have already assumed that that all information in time step "k" is known, and we have equations for the acceleration and velocity of time step "k+1" as a function of the displacement at time step "k+1". The load at time step "k+1" is also known.

$$\begin{aligned}
&\ddot{r}_k, \dot{r}_k, r_k, Q_k, Q_{k+1} = \text{ known} \\
&\ddot{r}_{k+1} = \text{ Unknown but expressed as a function of } r_{k+1} \\
&\dot{r}_{k+1} = \text{ Unknown but expressed as a function of } r_{k+1} \\
&r_{k+1} = \text{ Unknown}
\end{aligned}$$

When this information is applied, we have a set of equations with only one unknown for each degree of freedom. This equation is obtained by inserting equations 3.2.7.3 into equation 3.2.7.2 and solving for r_{k+1}. Now we can insert the calculated displacement of time step "k+1" into equations 3.2.7.3 to get the acceleration and velocity at time step "k+1". This completes the procedure for one time step and we have calculated the displacement, velocity and acceleration at the end of the time step.

This procedure has to be repeated for each time step until the solutions for the entire duration of the analysis are found.

The choice of the integration constants β and γ are described under chapter 3.1.9 and is the same whether the method is used for one or multiple degrees of freedom.

4 Extreme statistics

We are often interested in finding the extreme values of stochastic variables to determine a design load or investigating the worst case scenario. This is the case in this thesis where we want to find the expected highest effective tension in one of the suspension lines. According to Fisher-Tippett it is convenient to use three different extreme value distributions classified after the shape of the tail in the initial distribution

- Type I: Exponential distributions
- Type II: Distributions with infinite moments
- Type III: Limited distributions

Reference [23]

In this thesis we use two distributions; Weibull- and Gumbel-distribution which are both exponential distributions classified as type I by Fisher-Tippet.

According to Gumbel, the expected extreme value of a set of extremes is:

$$\mu_y = u_n + \frac{C_2}{\alpha_n} \tag{4.1}$$

Where u_n and α_n are the parameters in the Gumbel distribution:

$$F_X(x) = \exp\left[-\exp\left\{-\alpha\left(x-u\right)\right\}\right] \tag{4.2}$$

u_n - The characteristic largest value of the initial variate X
α_n - An inverse measure of dispersion of the extreme variate X_n

The parameter C2 is given by Gumbel:

N	C1	C2	N	C1	C2
10	0,9497	0,4952	**60**	1,17467	0,55208
15	1,02057	0,5128	**70**	1,18536	0,55477
20	1,06283	0,52355	**80**	1,19382	0,55688
25	1,09145	0,53086	**90**	1,20073	0,55860
30	1,11238	0,53622	**100**	1,20649	0,56002
35	1,12847	0,54034	**250**	1,24292	0,56878
40	1,14132	0,54362	**500**	1,25880	0,57240
45	1,15185	0,54630	**1000**	1,26851	0,57450
50	1,16066	0,54854	**inf**	1,28255	0,57722

Table 4 Constants C1 and C2 as a function of the size of the sample (N). Reference [23]

To use equation 4.1 to find the expected extreme value we need no express the parameters in the Gumbel distribution in terms of the parameters in the distribution of the data. If the set of data to be analyzed follows a Weibull distribution we want to express the parameters in the Gumbel distribution in terms of the parameters in the Weibull distribution. This allows us to find the expected extreme value of the Weibull distributed data from equation 4.1.

To express u_n and α_n in terms of the parameters in the initial distribution we use the following relations (Reference [23]):

$$F_X(u_n) = 1 - \frac{1}{n} \tag{4.3}$$

$$\alpha_n = n f_X(u_n) \tag{4.4}$$

The procedure can be summed up in three steps:

1. Assume an initial distribution of the data.
2. Find expressions for u_n and α_n in terms of the parameters in the initial distribution from equations 4.3 and 4.4.
3. Insert the equations for u_n and α_n in equation 4.1 to obtain the expected extreme value.

The procedure is shown for a Weibull distributed sample in chapter 4.1.

In order to obtain optimal results only the global peaks should be used in the calculations. A global peak is the maximum peak between each zero crossing.

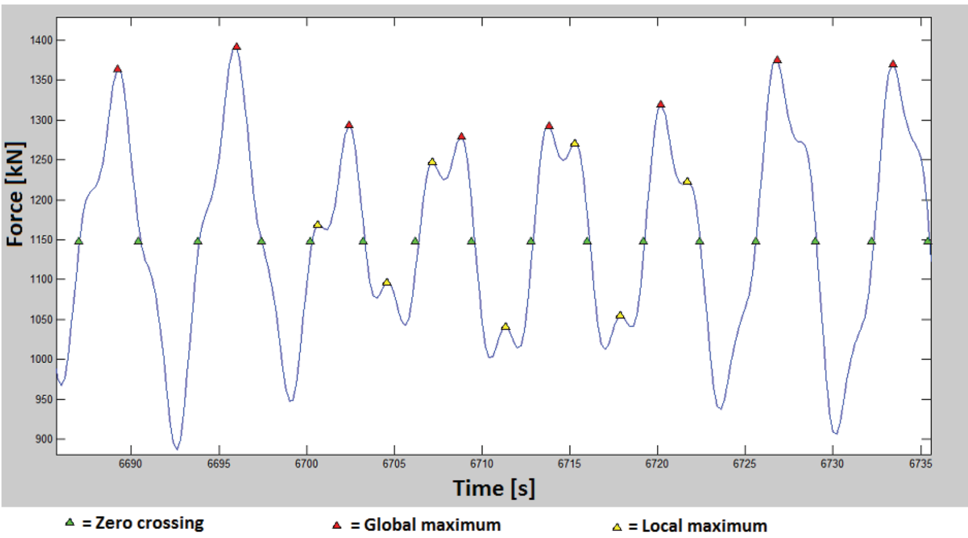


Figure 27 Definition of global maxima

4.1 Weibull distributed sample

If the sample follows a Weibull distribution we can find the expected extreme value as follows:

1. ***Assume an initial distribution of the sample***

The Weibull-distribution is an exponential distribution with a probability density function (pdf) and cumulative density function (cdf) given in equations 4.1.1 and 4.1.2 respectively

$$f_X(X) = \frac{\lambda}{\alpha}\left(\frac{x-\mu}{\alpha}\right)^{\lambda-1} \exp\left[-\left(\frac{x-\mu}{\alpha}\right)^{\lambda}\right] \tag{4.1.1}$$

$$F_X(X) = 1-\exp\left[-\left(\frac{x-\mu}{\alpha}\right)^{\lambda}\right] \tag{4.1.2}$$

$\mu =$ Weibull location parameter
$\alpha =$ Weibull scaling parameter
$\lambda =$ Weibull shape factor

2. ***Find expressions for*** u_n ***and*** α_n

Expression for Un

To find an expression for u_n we recall equation 4.3 and insert the CDF of the Gumbel distribution (equation 4.1.2)

$$\begin{gathered} F_X(u_n) = 1-\frac{1}{n} \\ \downarrow \\ 1-\exp\left[-\left(\frac{u_n-\mu}{\alpha}\right)^{\lambda}\right] = 1-\frac{1}{n} \end{gathered} \tag{4.1.3}$$

Taking the natural logarithm of each side of equation 4.1.3 yields:

$$\begin{gathered} -\left(\frac{u_n-\mu}{\alpha}\right)^{\lambda} = \ln\left(\frac{1}{n}\right) = \ln(1)-\ln(n) = -\ln(n) \\ \downarrow \\ u_n = \mu+\alpha\{\ln(n)\}^{\frac{1}{\lambda}} \end{gathered} \tag{4.1.4}$$

Expression for α_n

To find an expression for α_n we recall equation 4.4 and insert the PDF (equation 4.1.1)

$$\alpha_n = nf_X(u_n)$$

$$\alpha_n = n\frac{\lambda}{\alpha}\left(\frac{x_n-\mu}{\alpha}\right)^{\lambda-1}\exp\left[-\left(\frac{x_n-\mu}{\alpha}\right)^{\lambda}\right] \tag{4.1.5}$$

We insert the equation for u_n (equation 4.1.4) into equation 4.1.5.

$$\alpha_n = \frac{\lambda}{\alpha}\{\ln(n)\}^{\frac{\lambda-1}{\lambda}} \tag{4.1.6}$$

3. ***Obtain an equation for the expected extreme value.***

We have expressions for α_n and u_n.

Inserting equations 4.1.4 and 4.1.5 into equation 4.1 gives the expected extreme value for the Weibull fitted data.

$$\mu_y = \mu + \alpha\left[\{\ln(n)\}^{\frac{1}{\lambda}} + \frac{C_2}{\lambda}\{\ln(n)\}^{\frac{1-\lambda}{\lambda}}\right] \tag{4.1.7}$$

Estimating the parameters in the Weibull distribution

We have an expression for the expected extreme value of the Weibull distributed set of data. But we still need no estimate the parameters μ, α and λ. This is done according to reference [26]

The mean and the variance of the data can be estimated by the spectral moments as follows:

$$\tilde{m}_n = \frac{1}{N}\sum_{i=1}^{N}(x_i-\tilde{\mu})^n$$

$x_i =$ sample values
$\tilde{m}_n =$ n-th sample central moment
$\tilde{\mu} =$ sample mean
$N =$ Number of samples in the simulated time history (4.1.8)

$\hat{\mu} = \tilde{m}_1$ - estimate of the expected value of a set of data

$\hat{\sigma}^2 = \tilde{m}_2$ - estimate of the variance of a set of data

$c.o.v = \frac{\tilde{\sigma}}{\tilde{\mu}} = \frac{\sqrt{\tilde{m}_2}}{\tilde{m}_1}$ - estimate of the coefficient of variance (4.1.9)

An expression for the C.O.V as a function of c is given as:

$$C.O.\tilde{V} = \frac{\sqrt{\tilde{\mu}_2 - \tilde{\mu}_1^2}}{\tilde{\mu}_1 - x_1} = \frac{\sqrt{\Gamma\left(1+\frac{2}{\tilde{c}}\right) - \left[\Gamma^2\left(1+\frac{1}{\tilde{c}}\right)\right]}}{\Gamma\left(1+\frac{1}{\tilde{c}}\right)\left(1-\frac{1}{n^{\frac{1}{\tilde{c}}}}\right)} \tag{4.1.10}$$

By choosing different values for $\tilde{c}$, and comparing the C.O.V in equation 4.1.10 with C.O.V calculated from the data (equation 4.1.9), we can find $\tilde{c}$.

After finding an estimate for c we calculate an estimate for the parameters μ and λ with the following formulas:

$$\hat{\sigma}^2 = \alpha^2 \left\{ \Gamma\left(1+\frac{2}{\lambda}\right) - \left[\Gamma\left(1+\frac{1}{\lambda}\right)\right]^2 \right\} \tag{4.1.11}$$

$$\tilde{m}_1 = \mu + \alpha\Gamma\left(1+\frac{1}{\lambda}\right) \tag{4.1.12}$$

4.2 Gumbel distributed maxima

If we have a set of "n " maxima, one from each of our "n" samples, these maxima can be assumed to follow a Gumbel distribution.

The CDF of the Gumbel distribution is given in equation 4.2.1.

$$F_X(X) = \exp\left[-\exp\left\{-\alpha\left(x-u\right)\right\}\right] \tag{4.2.1}$$

u - The characteristic largest value of the initial variate X
α - Measure of the inverse dispersion of the extreme variate X_n

To find the expected extreme value from a set of maxima we recall equation 4.1:

$$\mu_{ex} = u_n + \frac{\gamma}{\alpha_n} \tag{4.2.2}$$

μ_{ex} - Expected extreme value
u_n - The characteristic largest value of the initial variate X
α_n - An inverse measure of dispersion of the extreme variate X_n
$\gamma \approx 0,57722$ - Euler number

Equation 4.2.2 differs from equation 4.1 by the constant C2 being replaced by the Euler number.

According to reference [36] the Gumbel estimators are given as:

$$\hat{\alpha} = \frac{C_1}{\hat{s}_y} \tag{4.2.3}$$

$$\hat{u} = \hat{\mu}_y - \frac{C_2}{C_1}\hat{s}_y \tag{4.2.4}$$

Where $\hat{s}_y$ is an estimate for the variance and $\hat{\mu}_y$ is an estimate for the expected value. These estimates can be found using moment estimators as in equation 4.1.8 and 4.1.9.The constants C1 and C2 in equations 4.2.3 and 4.2.4 can be found from Table 4 when we know the sample size. Inserting equations 4.2.3 and 4.2.4 into equation 4.2.2 gives the expected extreme value.

$$\mu_{ex} = \hat{\mu}_y - \frac{\hat{s}_y}{C_1}(C_2 + \gamma) \tag{4.2.5}$$

5 Case study

The 1-DOF analysis is done in a program developed in Matlab, and the multiple degree of freedom analysis is done in Orcaflex. The equipment used in the analysis is the same equipment as used in the Tyrihans operation. The complete 1-DOF program is found in appendix 12.1.

5.1 System description

In this chapter, a determination of the system used in the analysis will be done. All the properties and dimensions of the elements in the system can be found in appendix 12.2.

5.1.1 Tow configuration

The two hang-off wires (pennant wires) is the link between the installation vessel and the template. One end is attached to the hang-off frame above the moonpool of the vessel and the other end is connected to a four point bridle which is connected to the ITS. To maintain the desired orientation of the ITS a steering wire is attached between the vessel and the structure. Reference [9].

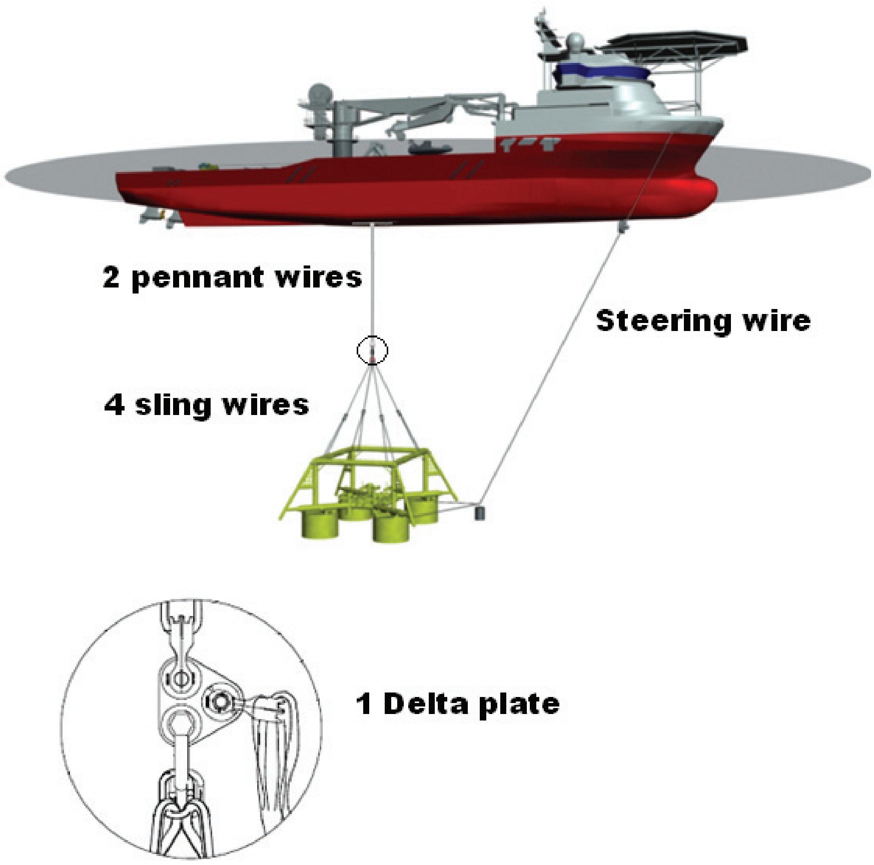


Figure 28 Tow configuration. Reference [4], [19].

5.1.2 Vessel

The vessel used for the analysis is the MSV Botnica. RAOs are defined for all six degrees of freedom and for a range of wave periods. The tow is to be performed at a towing speed of 1,5 m/s, which will be modeled as a uniform current. All data of the vessel is found in reference [8] and [9] and the main particulars are attached in appendix 12.2.2.

Figure 29 MSV Botnica. ©Arctia

5.1.3 Template

The template that is to be towed is a heavy ITS produced by FMC. Prior to the towing the roof hatches and guide posts will be removed. The mud mat hatches on top of the foundation buckets will be closed and all elements will be filled with water during towing. As the foundation buckets, legs and top frame has varying wall thickness, an equivalent wall thickness is used in the multiple degree of freedom analysis. The ITS parameters is found in appendix 12.2.3.

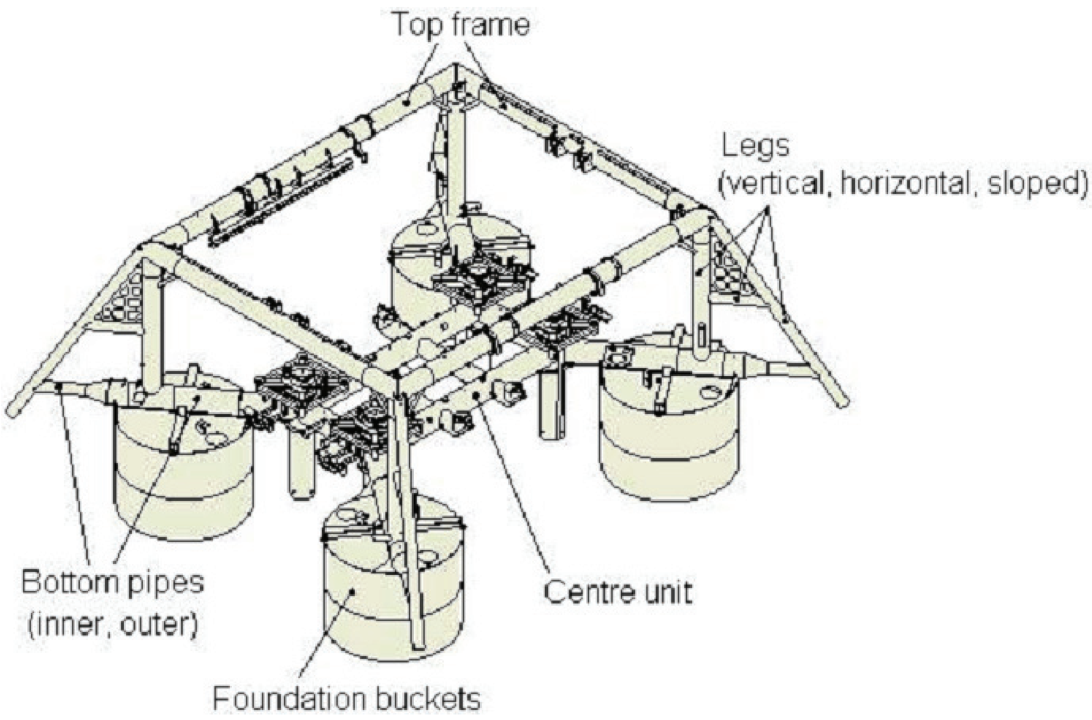


Figure 30 Sketch of the template. Reference [9]

5.1.4 Rigging

The rigging consists of two pennant wires suspended from the vessel and connected to a delta plate. The template is connected to the delta plate with four sling wires. There is also a steering wire with a clump weight which has the intension of keeping the template in the desired orientation. This is illustrated in Figure 28. The parameters of the rigging are found in appendix 12.2.4.

5.1.5 Environment

The towing is performed in irregular waves modeled from the JONSWAP spectrum. The environmental conditions are given in appendix 12.2.5.

5.2 Modeling

5.2.1 Modeling in Matlab

The modeling in Matlab is done according to the theory and formulas derived in chapter 3.1. Some further assumptions and simplifications have been done; this will be described in this chapter.

5.2.1.1 Vessel

The movement of the vessel is described by a transfer function from reference [8]. The hydrodynamic data of the MSV Botnica is imported in the Orcaflex analysis and the transfer function of the hang-off point of the wires is exported to a text file. This text file is used to import the transfer function to the 1-DOF program.

The transfer function is given as a function of the period with only 36 values. We want the transfer function to be a function of the frequency and we need an approximated equation to be able to use it in the 1-DOF program. To transform the period to frequency, we use the following formula:

$$\omega = \frac{2\pi}{T} \tag{5.2.1.1}$$

$\omega =$ Wave frequency
$T =$ Wave period

The RAO extracted from Orcaflex is found in appendix 12.2.2, Table 14.

To calculate an approximate equation for the transfer function we use the curve fitting tool in Matlab. We plot the given transfer function (Table 14) and approximate it with a rational fit with a 5th degree polynomial as the numerator and a 5th degree polynomial as the denominator.

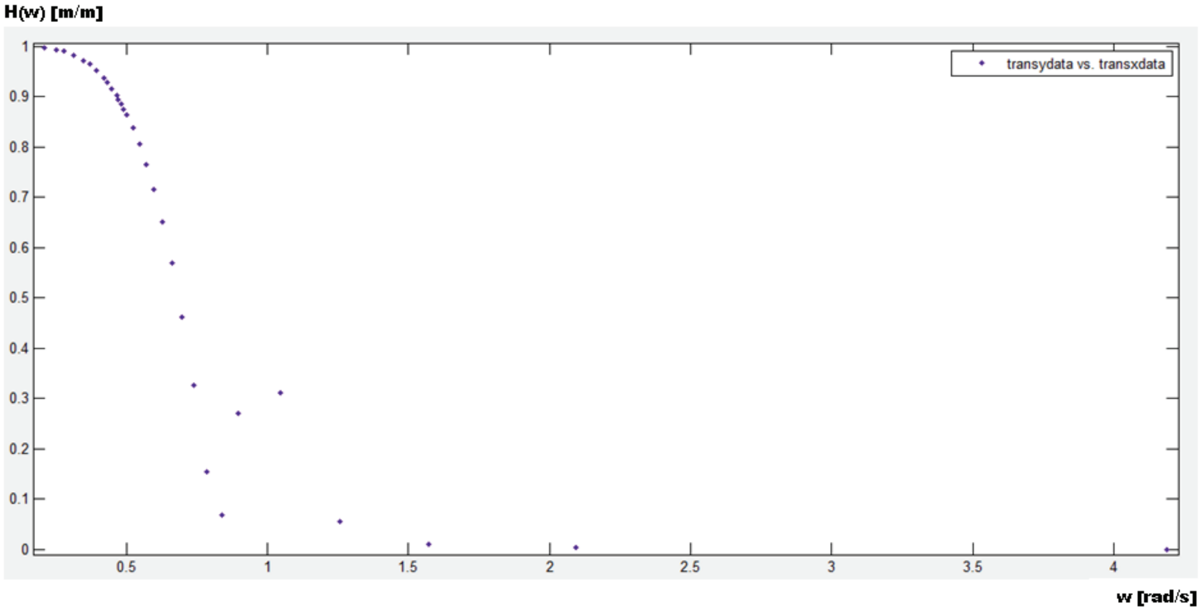


Figure 31 Plot of the transfer function extracted from Orcaflex

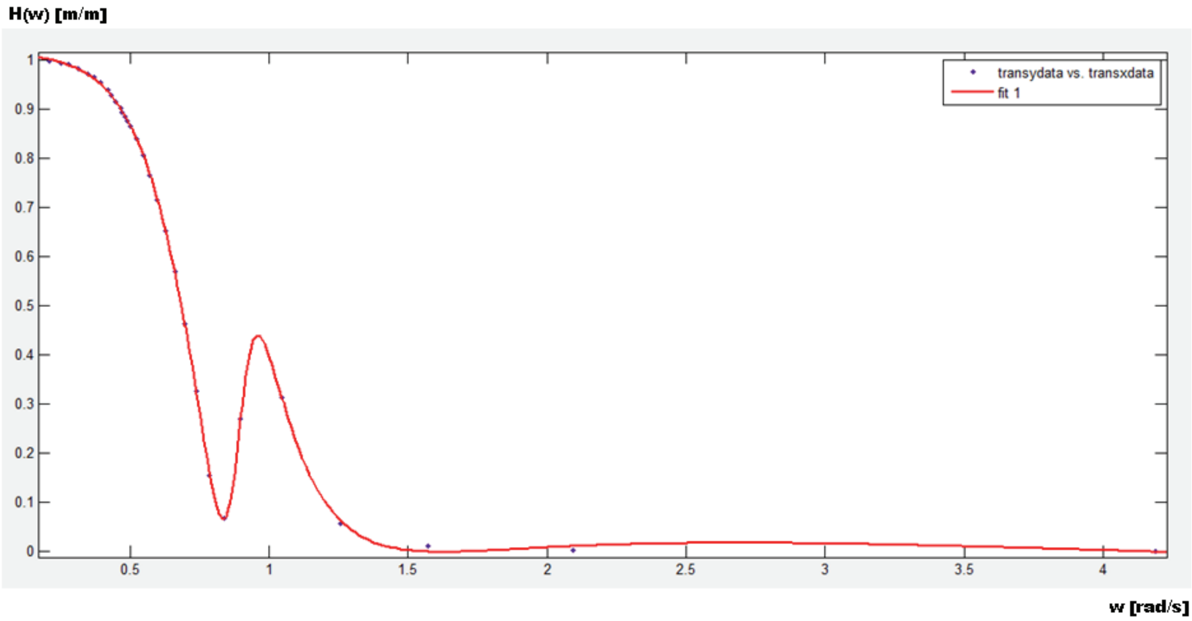


Figure 32 Plot of the transfer function (Figure 31) after curve fitting in Matlab.

H(w) = (p1*w^5 + p2*w^4 + p3*w^3 + p4*w^2 + p5*w + p6) / (w^5 + q1*w^4 + q2*w^3 + q3*w^2 + q4*w + q5)

Coefficients (with 95% confidence bounds):

p1 = -0.194 (-0.3746, -0.01333)

p2 = 1.76 (0.1575, 3.362)

p3 = -5.679 (-10.76, -0.5975)

p4 = 8.408 (1, 15.82)

p5 = -5.793 (-10.83, -0.7572)

p6 = 1.505 (0.2119, 2.798)

q1 = -1.07 (-2.901, 0.7618)

q2 = -3.239 (-9.42, 2.942)

q3 = 7.347 (-0.9763, 15.67)

q4 = -5.493 (-10.77, -0.2185)

q5 = 1.471 (0.1563, 2.785)

Figure 33 Approximated equation of the transfer function in Figure 32.

5.2.1.2 Template

The template is modeled as a lump mass with given mass, added mass and drag. The added mass in z-direction (definition of axes in Figure 14) is given in reference [9] and a summary is given in appendix 12.2.3, Table 21.

The drag in x-and z-direction is calculated by superimposing each member of the ITS. The formula for calculating the drag force is:

$$F_D = C_D \cdot A_P \qquad (5.2.1.2)$$

F_D=Drag force
C_D=Drag coefficient
A_P= Projected area

To be able to calculate the total drag force of the template we have to calculate the total drag area and an equivalent drag coefficient. This is done by using the following formulas:

$$A_{tot} = \sum A_i \qquad (5.2.1.3)$$

$$C_{Deq} \cdot A_{tot} = \sum A_i \cdot C_{Di} \quad \Rightarrow \quad C_{Deq} = \frac{\sum A_i \cdot C_{Di}}{A_{tot}} \qquad (5.2.1.4)$$

C_{Deq} = Equivalent drag coefficient
A_{tot} = Total projected area
A_i = Projected area of one unit
C_{Di} = Drag coefficient of one unit

The dimensions and drag coefficients in x- and z-direction of the ITS are found in appendix 12.2.3, and the calculations are found in appendix 12.2.3, Table 22 and Table 23.

5.2.1.3 Rigging

The rigging of the system to be analyzed consists of two hang-off wires, a delta plate and a four sling bridle (described in chapter 5.1.4). The 1-DOF program assumes one wire suspended from the vessel and attached to the template. This means that the steering wire has been neglected and we need to calculate an equivalent wire for all the rigging.

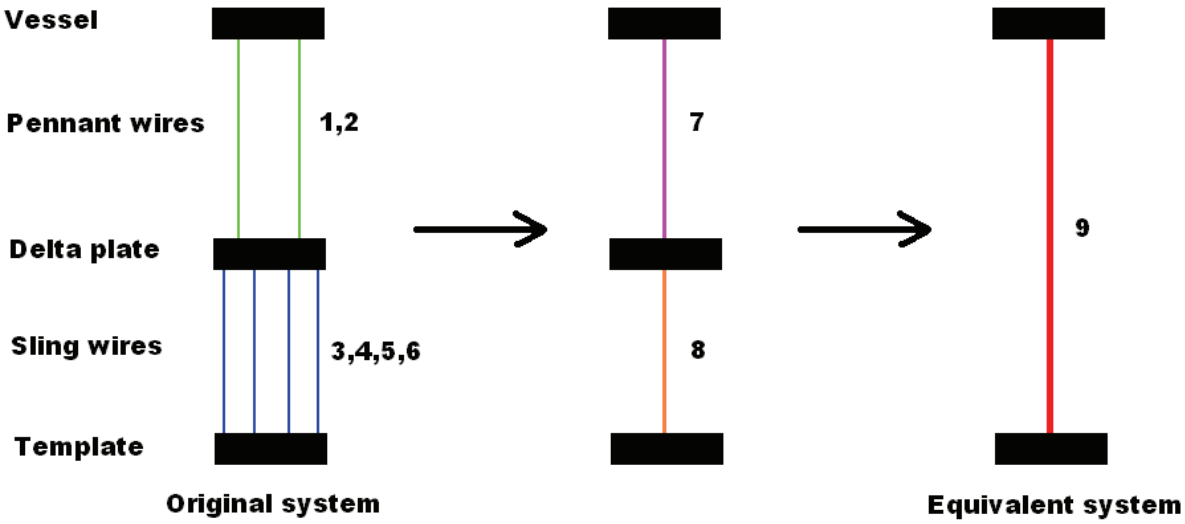


Figure 34 Equivalent wire for the system

The axial stiffness of each wire is known. To find an equivalent axial stiffness we use the following formula:

$$K_7 = K_1 + K_2$$
$$K_8 = K_3 + K_4 + K_5 + K_6$$
$$\frac{1}{K_9} = \frac{1}{K_7} + \frac{1}{K_8}$$
$$K_{eq} = K_9 = \left[\frac{1}{\left(\frac{1}{K_7}\right) + \left(\frac{1}{K_8}\right)}\right] = \left[\frac{1}{\left(\frac{1}{K_1 + K_2}\right) + \left(\frac{1}{K_3 + K_4 + K_5 + K_6}\right)}\right] \tag{5.2.1.5}$$

K_i = Axial stiffness of element "i"
K_{eq} =Equivalent axial stiffness

To find the equivalent drag coefficient we use the same procedure as for the template in chapter 5.2.1.2.

The parameters of the equivalent wire and a description of how they are calculated is found in appendix 12.2.4, Table 27.

5.2.1.4 Environment

The wave spectrum is modeled as described in chapter 3.1.3.2, and the environmental conditions are given in appendix 12.2.5.

5.2.2 Modeling in Orcaflex

In order to do an analysis in Orcaflex it is essential to understand the theory. This chapter will describe how an analysis is done in Orcaflex and a short description of the elements that are used in the analysis.

5.2.2.1 Static analysis

Before a dynamic simulation is done it is useful to do a static analysis to obtain the starting configuration of the system. This determines the equilibrium configuration of the system under weight, buoyancy, hydrodynamic drag etc. The calculation of the equilibrium is determined in a series of iterative stages. The initial positions of vessels and buoys, hence also the initial positions of the lines attached are decided by the initial data. The external loads are applied stepwise and new equilibrium points are calculated. This is repeated until the sum of all forces equals zero and we have reached equilibrium. Reference [17].

5.2.2.2 Dynamic analysis

After the static analysis is performed and the starting configuration of the system is obtained, a dynamic analysis is done. To get a gentle start to the simulation there is a build-up stage called ramping. This will be described in chapter 5.2.2.4.

The dynamic analysis in Orcaflex implements two complementary dynamic integration schemes, explicit (forward Euler) and implicit (Generalized alpha integration). Both methods are generalizations of the Newmark beta method which is described in chapter 3.1.9 and 3.2.7 .

Reference [17].

5.2.2.3 Interpolation methods

Orcaflex uses three different methods for interpolating data.

1. Linear – The data follows a straight line between each data point. Continuous but discontinuous first derivative.
2. Cubic spline – Fits a cubic polynomial over each interval of the data. Both first and second derivative is continuous. A change of one data point affects the the interpolated curve over the whole range.
3. Cubic Bessel – Fits a cubic polynomial over each interval of the data. Only the first derivative is continuous. A change of one data point only affects the interpolated curve over the intervals near that point.

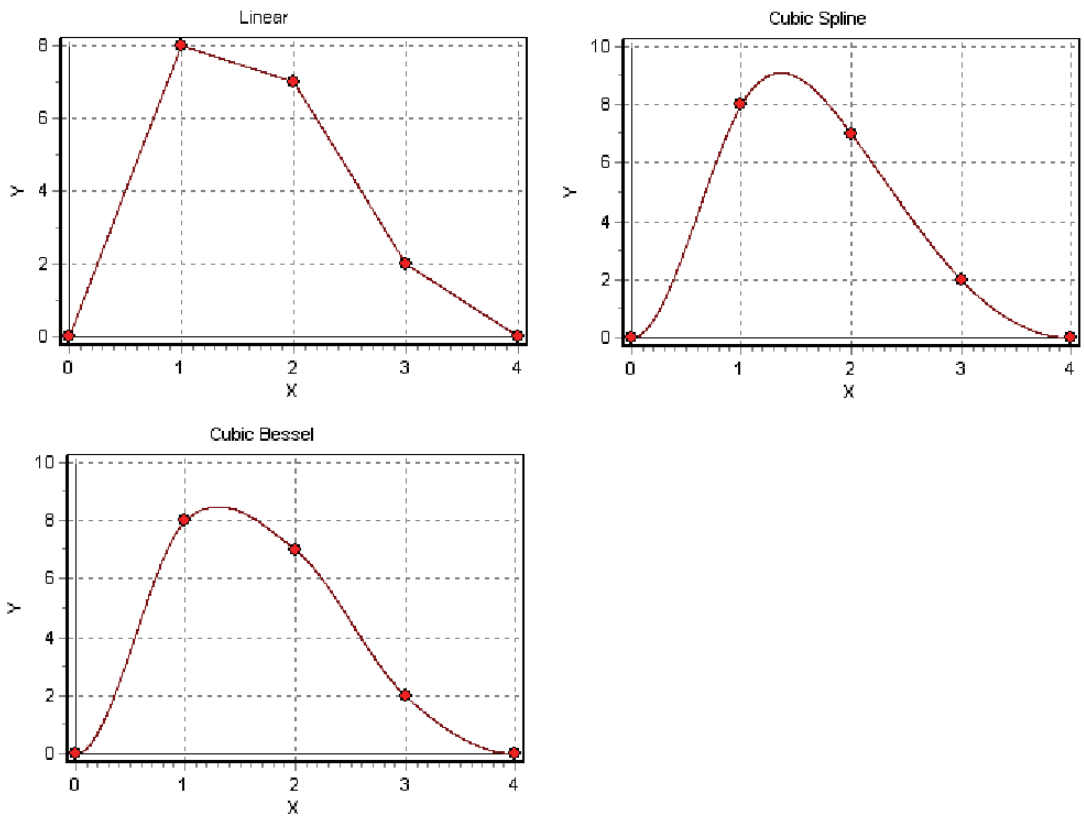


Figure 35 Illustration of three interpolation methods. Reference [17]

The choice of method depends on what the interpolated data is used for and the different properties of the interpolation methods. Interpolation with cubic –spline or –bessel often overshoots the curve if you do not have enough data points. This can cause a problem if you for instance want to model the transfer function of a vessel. You should also consider what continuity is needed. Linear interpolation is appropriate if continuity of the first derivative is not required. If continuity of the first derivative is required then cubic –spline or –bessel will be appropriate. Cubic spline is the appropriate interpolation method if both the first and second derivative needs to be continuous. Reference [17].

5.2.2.4 Ramping

The build-up stage is performed to get a gentle start of the simulation and reduce the need of a long simulation. This procedure is called ramping. During the build-up stage of minimum one wave period, the wave dynamics, vessel motions and the current are built up smoothly from zero to their full level. This is done by using the ramping factor:

$$\text{Ramping factor} = r^3(6r^2 - 15r + 20)$$
$$r = \frac{(\text{Time + length of build-up stage})}{(\text{length of build-up stage})} \qquad (5.2.2.1)$$

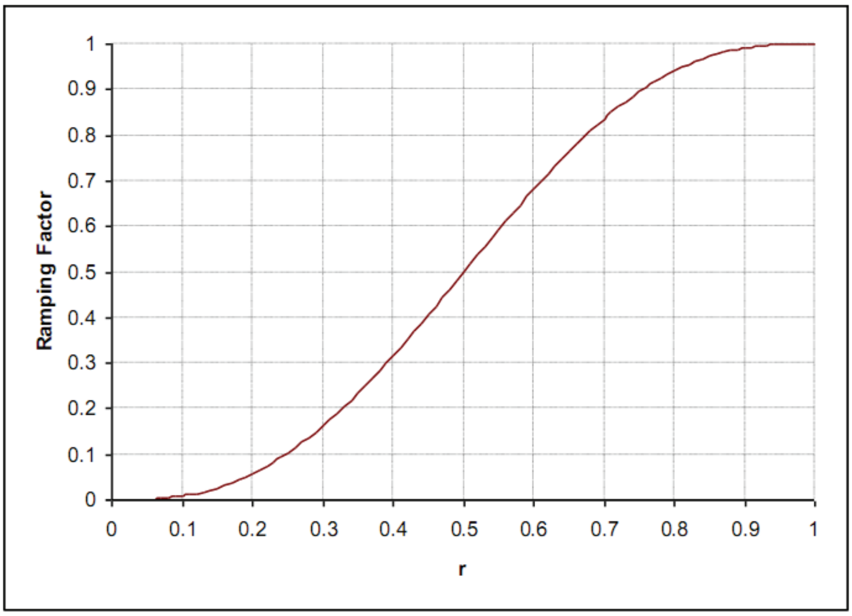


Figure 36 Illustration of the ramping factor. Reference [17]

Reference [17].

5.2.2.5 Environment

The JONSWAP wave spectrum is chosen to model the irregular waves. The wave direction, significant wave height and zero crossing period are inserted and the other parameters in the JONSWAP spectrum are calculated by Orcaflex. A description of the wave spectrum is given in chapter 3.1.3. All the environmental properties are given in appendix 12.2.5.

Current:

Orcaflex uses three methods to model the current.

1. Extrapolation
2. Interpolation
3. Power law

Extrapolation:

The current is defined at the still water level. However at the presence of waves the current has to be extrapolated above the still water level. Orcaflex uses the surface current for all points above the still water plane. The same thing applies if we have a sloped seabed. The current at the greatest depth specified is applied to all greater depths.

Interpolated method:

For intermediate depth interpolation of the current is used and the current is specified from the still water surface to the seabed.

Power law:

The direction of the current is specified by the user and does not change, whilst the velocity varies with the position according to the formula:

$$U(Z) = U_{seabed} + (U_{surface} - U_{seabed}) \cdot \left[\frac{(Z - Z_{seabed})}{(Z_{surface} - Z_{seabed})} \right]^{\left(\frac{1}{exponent}\right)}$$

$U(Z) =$ Current velocity at the point Z

$\mathrm{U}_{seabed} =$ Current velocity at the seabed

$\mathrm{U}_{surface} =$ Current velocity at the surface

$Z =$ Position of interest

$Z_{seabed} =$ z-position at the seabed

$Z_{surface} =$ z-position at the surface

$Exponent =$ power law exponent

if $\mathrm{Z} \leq \mathrm{Z}_{seabed} \rightarrow Z = Z_{seabed}$

if $\mathrm{Z} \geq \mathrm{Z}_{surface} \rightarrow Z = Z_{surface}$ (5.2.2.2)

Reference [17].

Waves:

The waves in Orcaflex can be modeled as regular, random or specified by a time history.

Five standard frequency spectra are offered in Orcaflex: JONSWAP, ISSC, Pierson-Moskowitz, Ochi-Hubble, Torsethaugen and Gaussian Swell. The JONSWAP and Pierson-Moskowitz spectra are described in detail in chapter 3.1.3.2 and 3.1.3.1 respectively. The number of wave components is determined by the user and chosen using an equal energy approach. This means that the frequency spectrum is divided into n parts with equal amount of spectral energy. This approach results in a finer discretisation used around the spectral peak. To get the same level of discretisation using equal frequency spacing would result in many more components being used. So the equal energy approach used in Orcaflex is more efficient compared to the equal frequency spacing used in the Matlab program developed in this thesis.

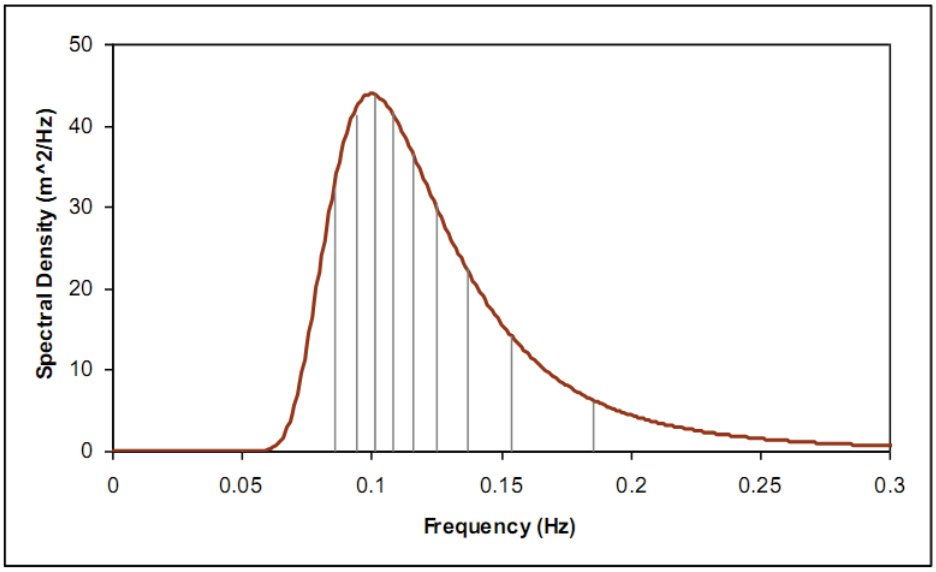


Figure 37 Equal energy approach to choosing wave components. Reference [17].

Orcaflex enables the user to implement the direction of the waves using the following formula:

$$S(f,\theta) = S_f(f) \cdot S_d(\theta) \quad = \text{ total wave spectrum}$$

$$S_d(\theta) = K(s)\cos^{2s}(\theta - \theta_p) \quad \text{for} \quad \frac{-\pi}{2} \le \theta - \theta_p \le \frac{\pi}{2} \quad = \text{directional spreading spectrum}$$

$$S_f(f) = \text{ frequency spectrum}$$

$$K(s) = \pi^{\frac{-\pi}{2}} \frac{\Gamma(s+1)}{\Gamma(s+\frac{1}{2})} \quad = \text{normalising constant}$$

$$2s = \text{ spreading exponent}$$

$$\theta = \text{ wave direction}$$

$$\theta_p = \text{ principal wave direction} \tag{5.2.2.3}$$

Reference [17].

5.2.2.6 Vessel

The vessel element in Orcaflex is a rigid body with motions prescribed by the user.

The movement of the vessel is defined by RAOs. Each RAO is imported from model tests or specialist computer programs and defines the vessels response amplitude for one particular degree of freedom for one particular wave direction and period. To get a complete motion characteristic of the vessel the RAO must give information of the vessels movement in each of the six degrees of freedom for each wave direction and period.

The Orcaflex way of defining the RAO is to use the amplitude of response per unit wave amplitude and the phase lag from the time the wave crest passes the RAO origin until the maximum positive excursion is reached. This can be formulated mathematically as:

$$x = R \cdot a \cdot \cos(\omega t - \varphi)$$

$x =$ vessel displacement in length units

$a, \omega =$ wave amplitude in length unit and frequency in radians/second

$R, \varphi =$ RAO amplitude and phase

$t =$ time in seconds (5.2.2.4)

When the RAO is imported Orcaflex also needs to know:

- Coordinates of the RAO origin and of the phase origin
- The wave direction (zero degrees=waves approaching from astern, 90 degrees = waves approaching from the starboard side and 180 degrees=waves approaching from the bow.

It is also important to be consequent regarding the axes. Orcaflex uses a right-handed system with positive movements defined as:

Surge – positive forward

Sway – positive to port

Heave – positive upwards

Roll – Positive starboard down

Pitch – Positive bow down

Yaw – Positive bow to port

Reference [17].

To model the movement of the MSV Botnica, the hydrodynamic properties are imported and related to the COG of the vessel. The RAO that is used is calculated at the moonpool position and without a stabilizing tank. As the movement of the vessel is specified by RAOs, it is not affected by the tension in the wires. Since the transfer function of the ship is obtained by a linear interpolation between the hydrodynamic data given from test results, the resulting transfer function is different from the smooth function derived in Matlab. In order to compare the results from the 1-DOF analysis with the ones obtained in the multiple-DOF analysis, we need to use the exact same transfer function. This is solved by importing the data for the transfer function from Orcaflex to Matlab, doing the curve fitting in Matlab (as described in chapter 5.2.1.1) and exporting the smooth transfer function from Matlab back to Orcaflex. Reference [8].

5.2.2.7 Template

The template is modeled by connecting lines to a 6d buoy. The 6d buoy models the center unit of the template and includes the mass and hydrodynamic properties. We use the 6d buoy to get a template that can move in all six degrees of freedom. The line facility is used to model the top frame, legs, foundation buckets and bottom beams which are set to be fixed according to axes of the center unit. This enables all the parts to follow the movement of the center unit. All cylindrical elements modeled by the line facility are water filled and all the hydrodynamic properties are included. All the parameters and drawings needed to make a model of the template are included in appendix 12.2.3.

5.2.2.8 Rigging

The rigging consists of two pennant wires, four sling wires, one delta plate and one clump weight. All the wires are modeled using the line facility, and the delta plate is modeled using a 6d buoy where the hydrodynamic properties and mass are neglected. The clump mass is modeled using a 3d buoy where the mass is included and the hydrodynamic properties are neglected. All the parameters used to model the rigging are found in appendix 12.2.4.

5.2.2.9 Lines

The line facility in Orcaflex is very versatile. It can be used to represent pipes, flexible hoses, cables, mooring lines etc. Line properties may vary over the length for example to vary the diameter of a pipe.

Orcaflex discretisises the lines as illustrated in Figure 38 and uses a finite element model to calculate the forces and moments on each node.

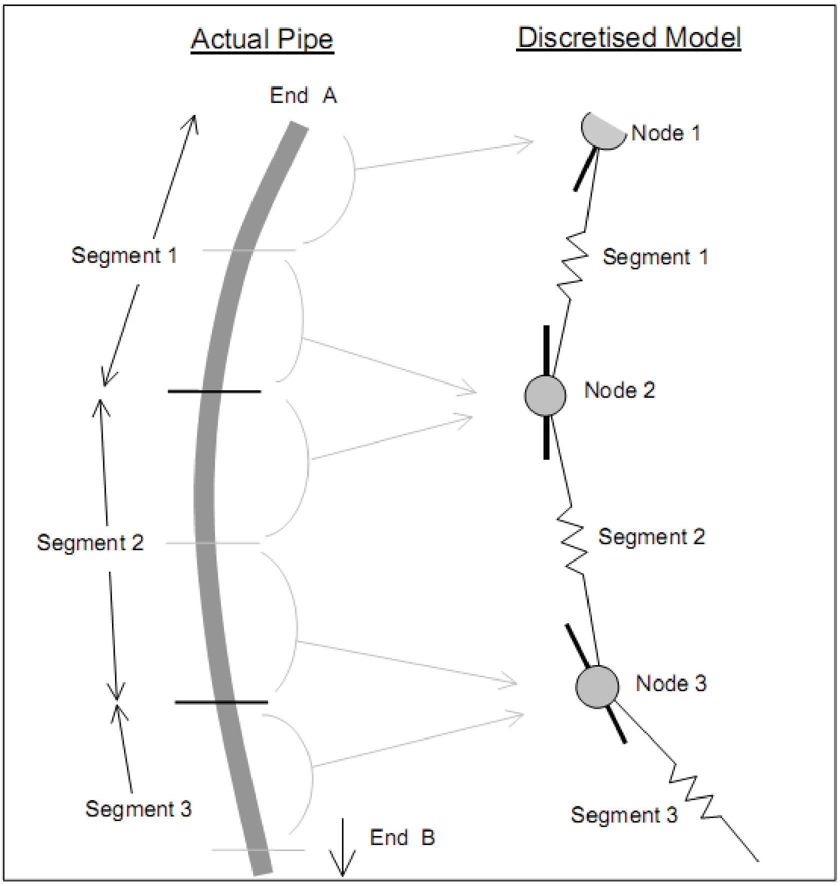


Figure 38 Orcaflex line model. Reference [17].

The line is divided in a series of mass less line segments which models the axial and torsion properties of the line. The properties like mass, weight, buoyancy etc. are lumped to the nodes which connects the line segments to each other.

The forces and moments on each node are divided into five stages:

1. Tension forces
2. Bend moments
3. Shear forces
4. Torsion moments
5. Total load

Description of the calculation in each stage will not be given, but is described in detail in reference [17].

5.2.2.103D- and 6D-buoys

The 3D buoys motion in the translational degrees of freedom is calculated by Orcaflex. They can model all rigid bodies and does not have to be buoyant even though they are called buoys. The 3D buoy does not rotate and is therefore used to model small bodies where rotations are not of interest. The properties like weight, buoyancy, drag and added mass are specified by the user to model the desired body.

The 6D buoys are more complex than the 3D buoys since they are rigid bodies with the full six degrees of freedom. This gives a different range of use, but the properties needed from the user are the same as for the 3D buoys.

Reference [17].

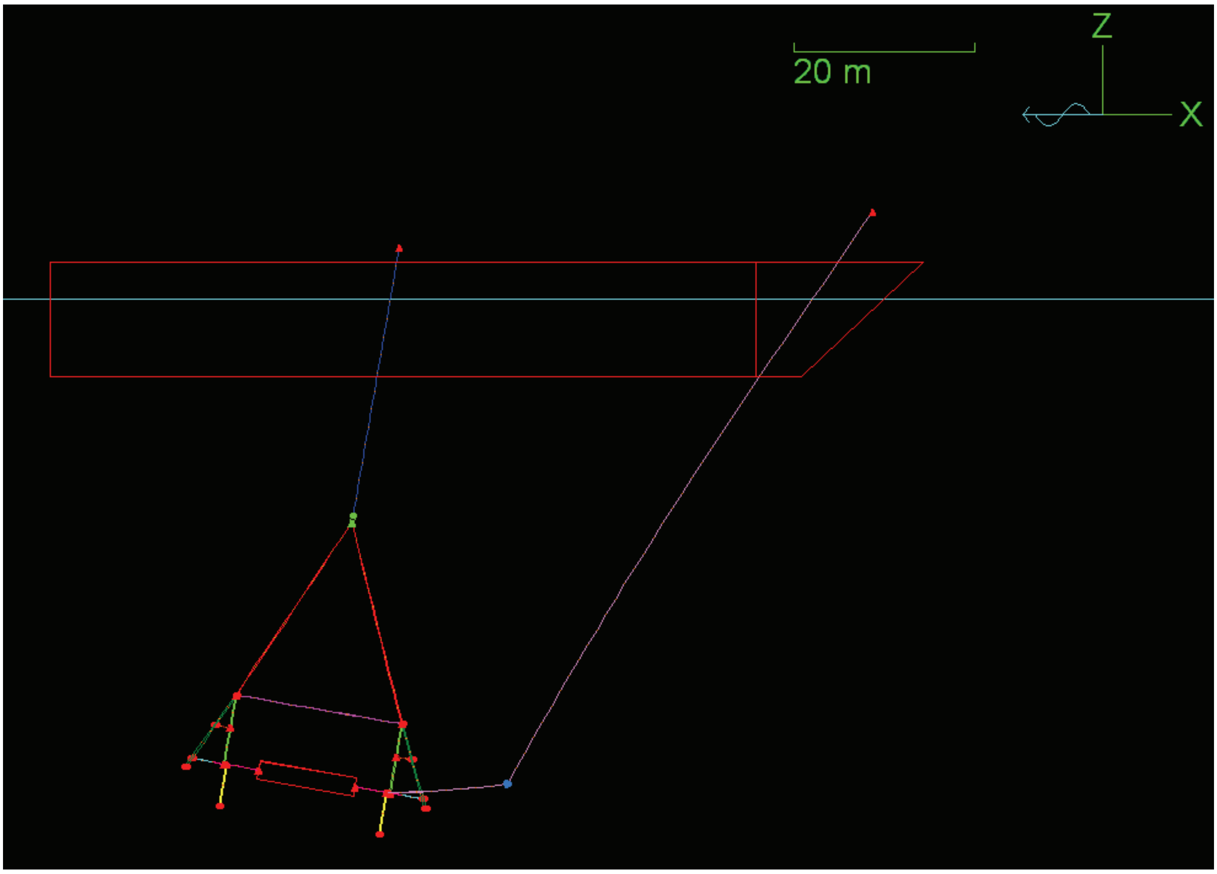


Figure 39 Orcaflex model

6 Results

The 1-DOF and multiple-DOF analysis of the system described in chapter 5.1 are performed with a length of 3 hour and with 1000 wave components. Description of how the modeling is done in Matlab and Orcaflex is described in chapter 5.2.1 and 5.2.2 respectively. The results of interest is the wave elevation, vessel movement in heave, template motion in heave and the offset angle of the template relative to the hang-off point. In addition to this, the 1-DOF program produces the following results:

- Drag force in Newton as a function of towing velocity in meters per second.
- Dynamic displacement of the wire as a function of the frequency.
- Dynamic force in the wire as a function of the frequency.
- 3D plot of the absolute part of the motion response with varying wire distance.
- 3D plot of the real part of the motion response with varying wire distance.
- 3D plot of the absolute force in the wire with varying wire distance.

These 1-DOF results are not relevant for the comparison with the multiple-DOF results and will therefore not be discussed in this thesis. The plot of the results can be found in appendix 12.3 and a detailed description can be found in reference [7].

7 Comparison of the results

To compare the 1-DOF program developed in this thesis with Orcaflex, two different analyses are performed. The comparison is done in Matlab.

1. 18 second analysis with regular waves described by a period of 6 seconds and a wave height from trough to crest of 2,9 meters.
2. 3 hour analysis with 1000 wave components, 0,2 second time step in irregular sea. The irregular waves are described by the JONSWAP spectrum, with significant wave height of 2,9 meters, and a peak period of 6 seconds.

7.1 Regular waves

The comparison of the results from the 1-DOF and multiple-DOF analysis with regular waves is done by inspection of the graphs given in this chapter.

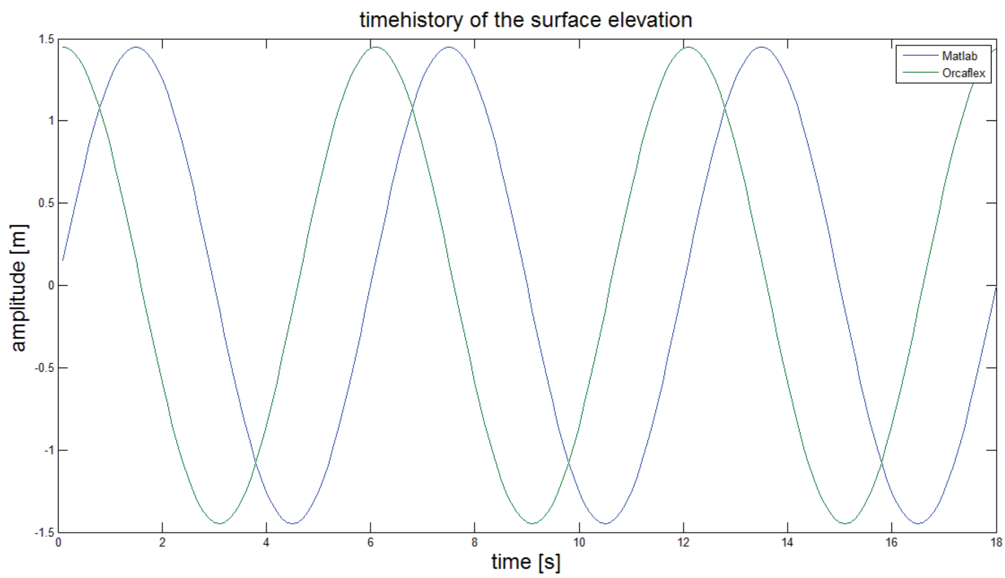


Figure 40 Time history of the regular waves. Regular analysis.

As we can see from Figure 40 the waves modeled in the 1-DOF program are identical to the ones modeled in Orcaflex.

The phase difference of the 1-DOF and multiple-DOF results is due to the 1-DOF program modeling sine waves whilst Orcaflex models cosine waves. This difference has no effect on the results.

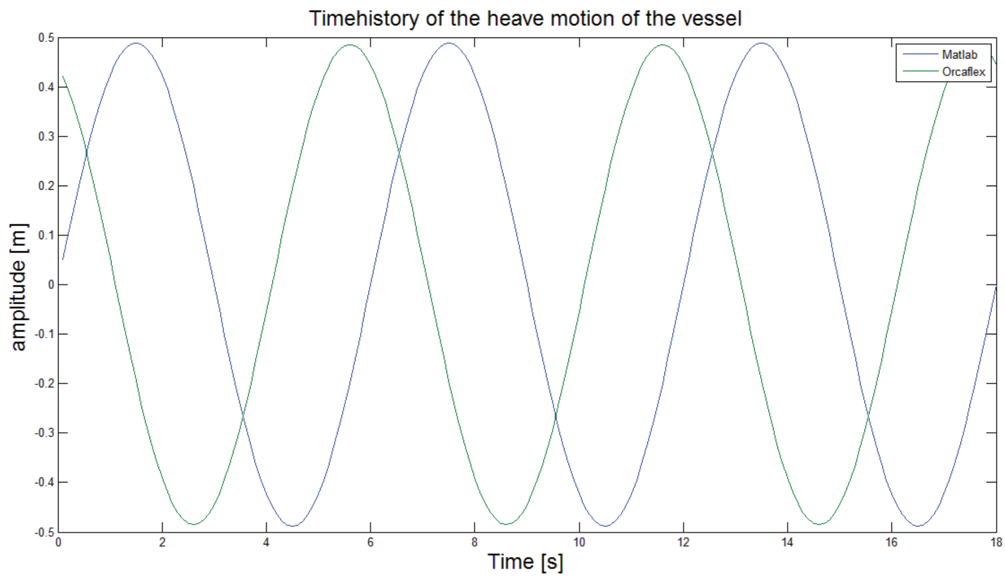


Figure 41 Time history of the heave motion of the vessel. Regular analysis.

From Figure 41 it shows that the vessel motions are identical, but it should be noted that there can be a difference because of how the value of the transfer function is chosen in 1-DOF program. This is because the frequency used in modeling the waves are calculated from the period. The period is 6 seconds, hence the frequency is given by the formula:

$$\omega = \frac{2\pi}{T} \tag{7.1.1}$$

ω = Wave frequency
T =Wave period

In the 1-DOF program the transfer function of the vessel is given as a function of the frequency and the intervals between the values of the transfer function is not small enough to cover all periods. If the chosen period gives a frequency between two points of the transfer function, the 1-DOF analysis is automatically done using the frequency closest to the one calculated from the period chosen by the user.

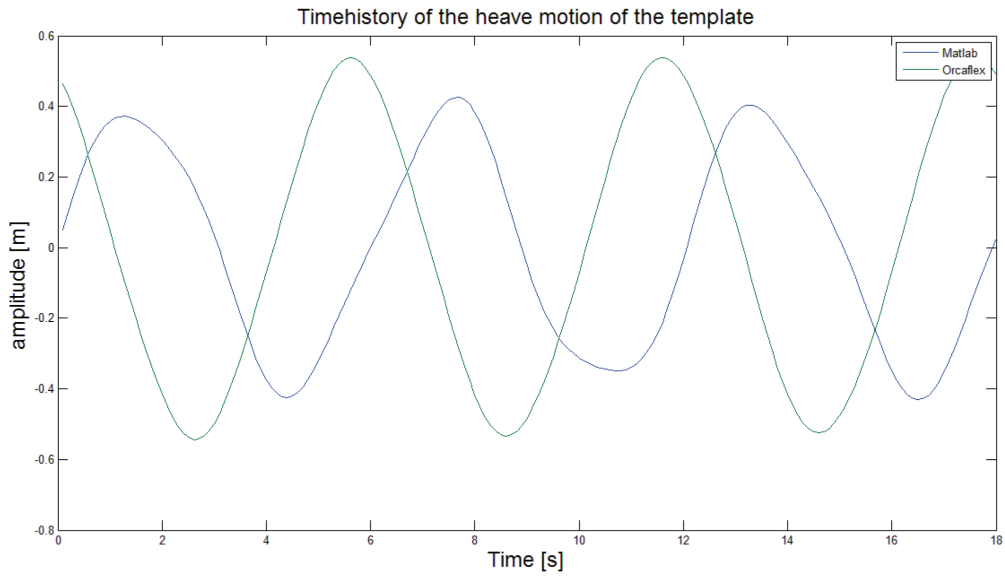


Figure 42 Time history of the heave motion of the template. Regular analysis.

As we can see from Figure 42 the heave motion of the template in the 1-DOF and multiple-DOF analysis have different behaviors.

To investigate further the motion of the template we look at the vessel and the template motion in the same plot.

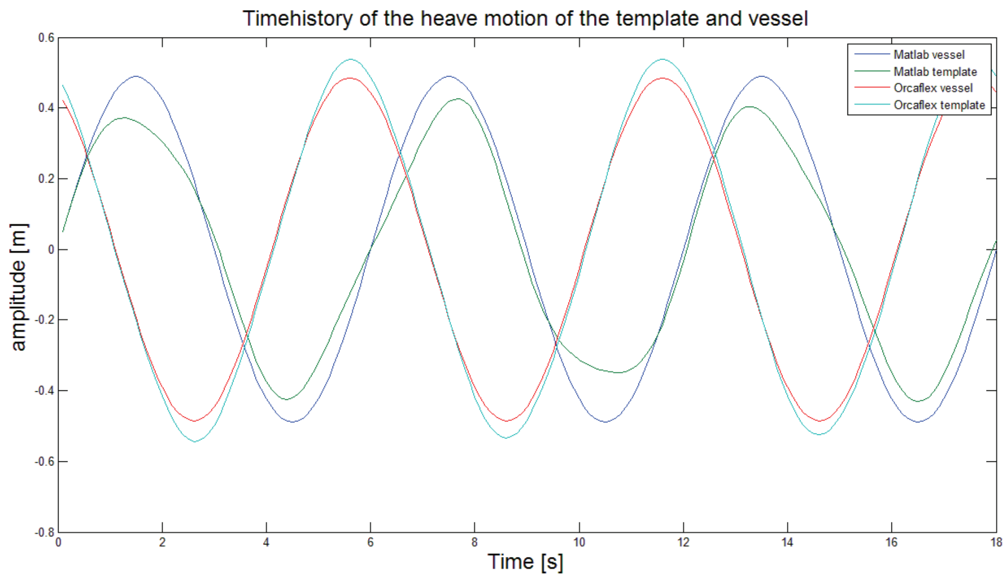


Figure 43 Time history of the heave motion of the vessel and the template. Regular analysis.

Figure 43 clearly shows a difference in the calculation of the template motion as the template in the multiple-DOF analysis has a motion closer to the vessel than in the 1-DOF analysis. If we look at the motion difference between the vessel and the template, this is confirmed.

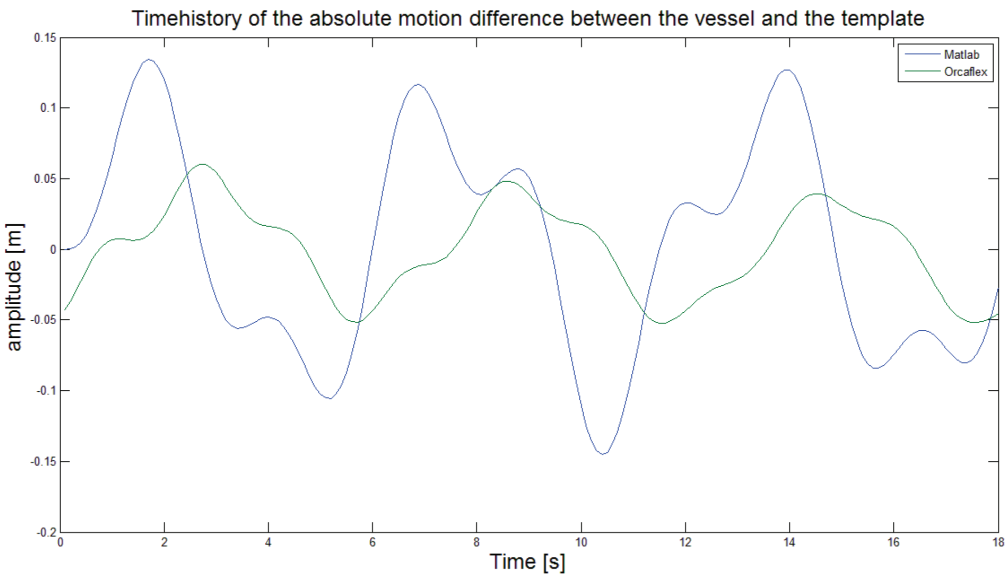


Figure 44 Motion difference between the vessel and template. Regular analysis.

From Figure 44 we can observe that the multiple-DOF analysis gives a motion difference that is a lot smoother than the 1-DOF analysis. In addition the motion difference tends to oscillate throughout the whole movement and the maximum value is a lot higher in the 1-DOF analysis.

The plots of the template motion and motion difference of the vessel and template (Figure 42 and Figure 44 respectively) clearly shows a weakness in how the template motion is modeled in the 1-DOF program. This will be discussed further in the parametric study of the template motion in irregular sea, Chapter 8.1.

7.2 Irregular waves

The comparison of the results from the analysis with irregular waves from the 1-DOF program and Orcaflex is done in Matlab according to the following three steps:

1. Plot the time history of the wave elevation, heave motion of the vessel and heave motion of the template.
2. Calculate the standard deviation of the time histories produced in step 1.
3. Do a FFT for each of the time histories to determine at which frequency the signal is at its strongest.

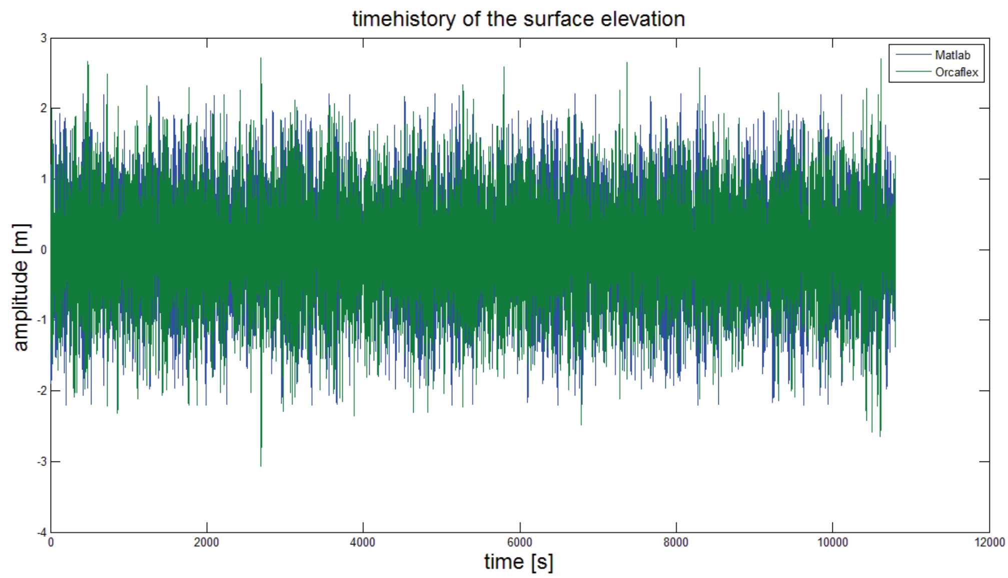


Figure 45 Time history of the surface elevation. 3 hour analysis.

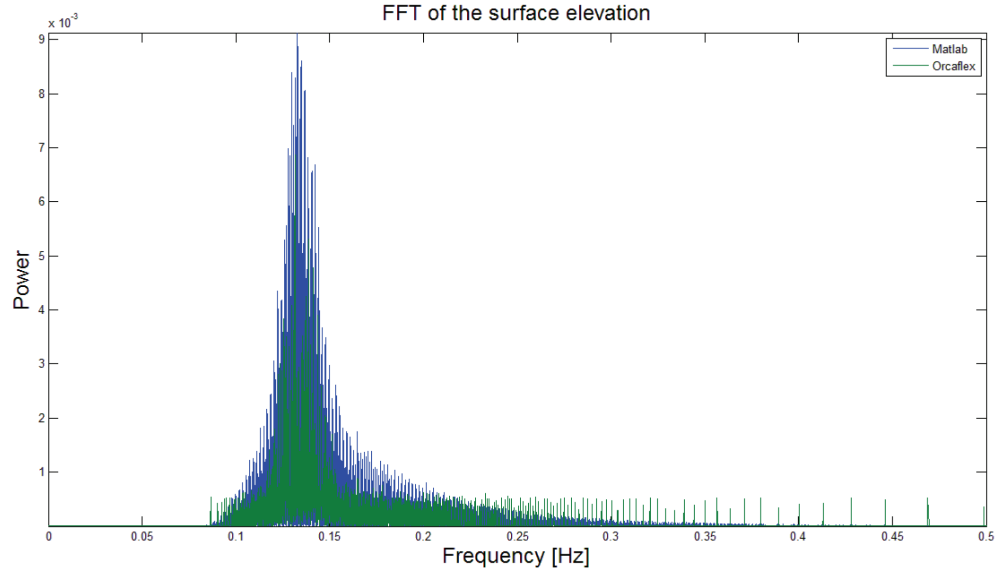


Figure 46 FFT of the surface elevation. 3 hour analysis.

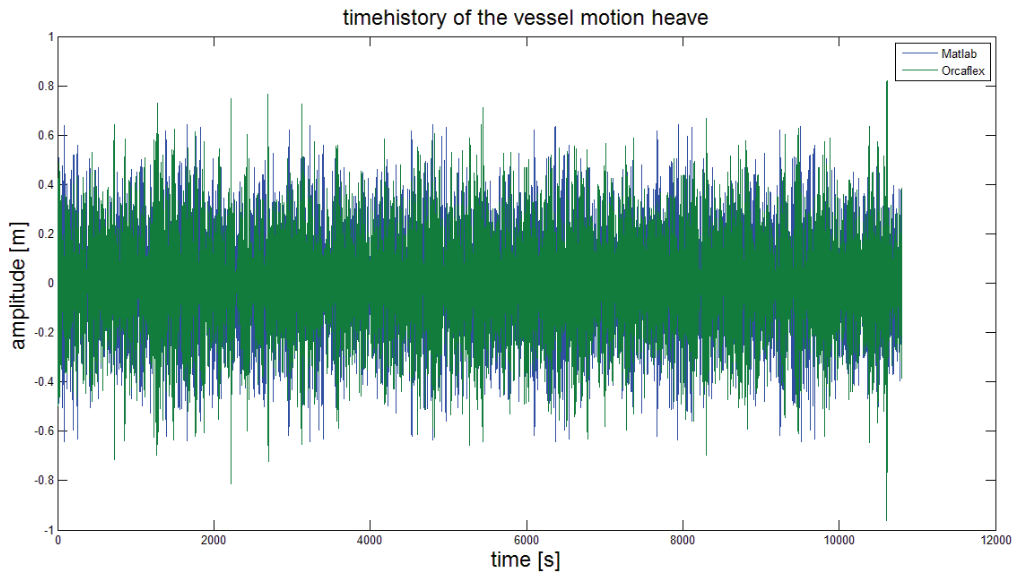


Figure 47 Time history of the vessel motion in heave. 3 hour analysis.

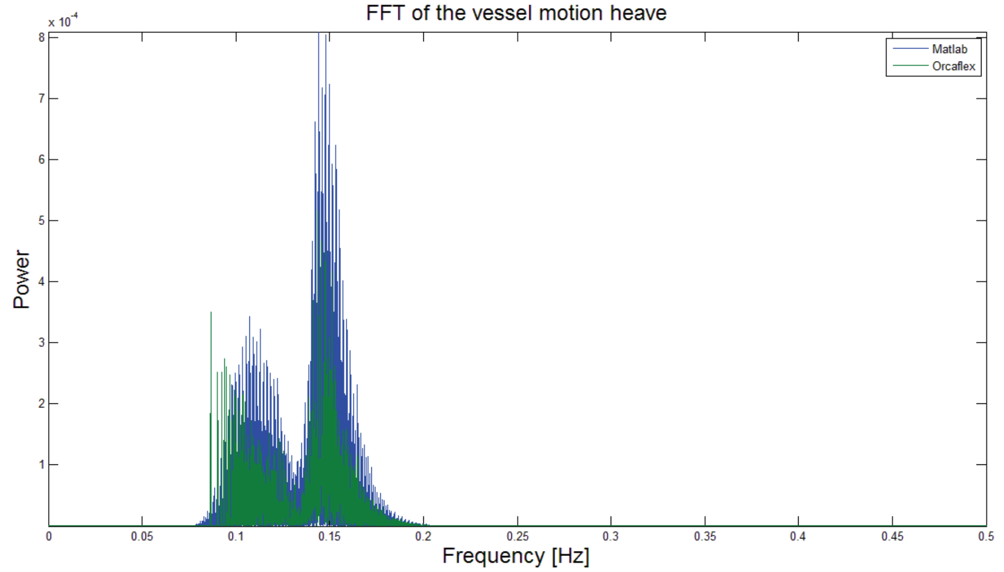


Figure 48 FFT of the vessel motion in heave. 3 hour analysis.

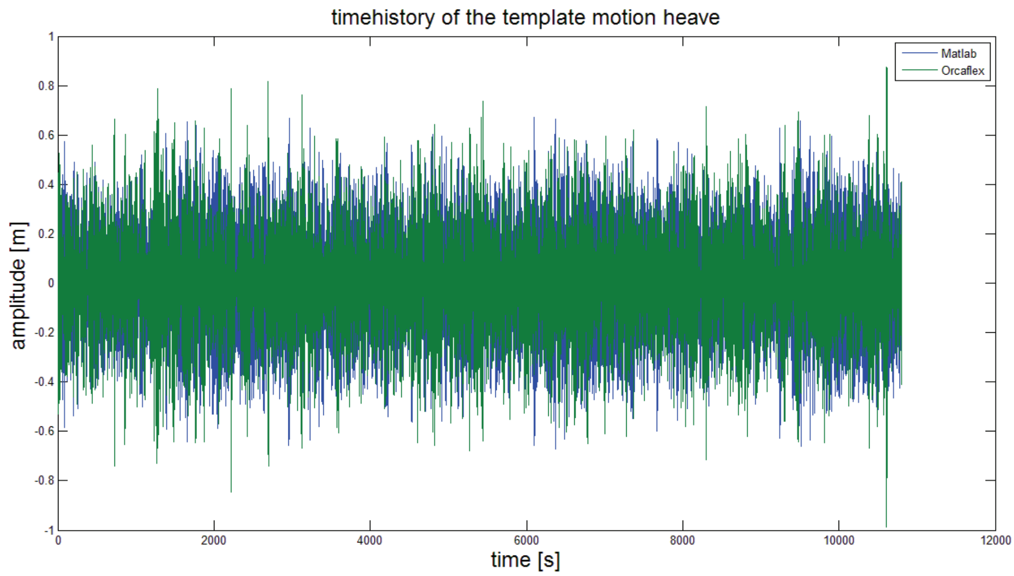


Figure 49 Time history of the template motion in heave. 3 hour analysis.

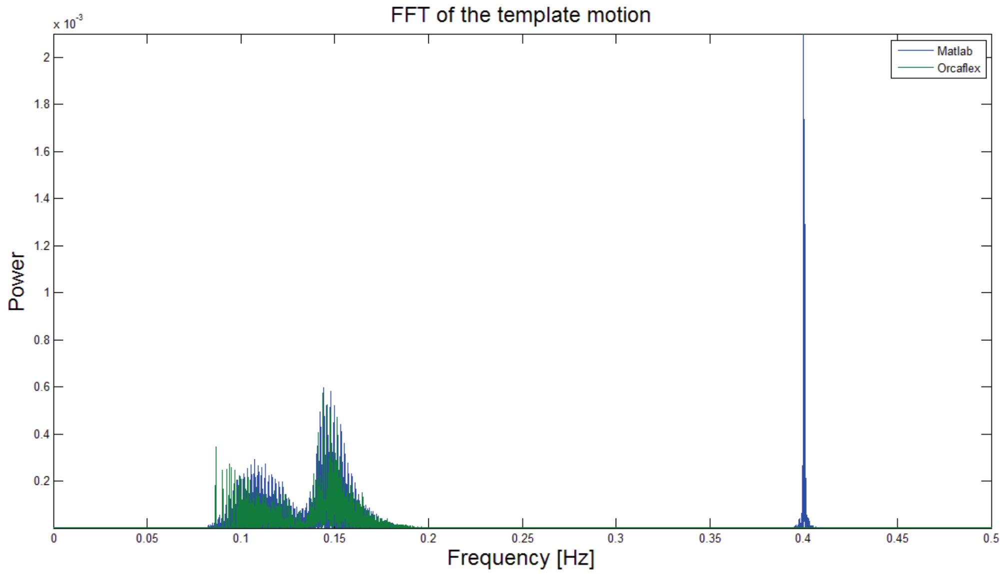


Figure 50 FFT of the template motion in heave. 3 hour analysis.

To examine the peaks of the FFT for the surface elevation, vessel motion and template motion (Figure 46, Figure 48 and Figure 50 respectively) it is convenient to plot these in separate graphs. These plots can be seen in appendix 12.4. The standard deviations of the time histories and the peaks of the FFT plots are calculated in Matlab.

	Standard deviation 1-DOF	Frequency peak 1-DOF(FFT)	Standard deviation multiple-DOF	Frequency peak multiple-DOF (FFT)
Wave elevation	0,7237	0,1326 [Hz]	0,7184	0,1326 [Hz]
Vessel motion	0,2048	0,1446 [Hz]	0,2046	0,1446 [Hz]
Template motion	0,1982	0,1446 [Hz]	0,2136	0,1446 [Hz]

Table 5 Comparison of the 1-DOF and multiple-DOF analysis in irregular

Step one in doing a towing analysis is to model the waves. The irregular waves are created by summing 1000 random wave components. The time histories for the vessel and template are then calculated with reference to the waves. Since the irregular waves in the 1-DOF and multiple-DOF analysis never will be exactly the same, some differences in the results are inevitable. This taken in mind, the standard deviations of the time histories and the frequency peaks from the FFTs of the time histories in Table 5, gives satisfying results.

As mentioned in chapter 7.1, there is a tendency of an oscillation motion of the template in the 1-DOF analysis. This shows in the FFT of the template motion, Figure 50. The shape and frequency peaks are the same, but in addition the 1-DOF analysis give a peak frequency about 0,4 Hz. This peak describes the oscillations and will be investigated and discussed in chapter 8.1 Parametric study of the template motion.

8 Parametric study

Two individual parametric studies are done in Matlab.

1. Parametric study of the template motion with varying added mass coefficients, from 80% to 120% of the initial value.

2. Parametric study of the horizontal offset with varying drag coefficients, from 80% to 120% of the initial value.

8.1 Parametric study of the template motion

To do a parametric study of the template motion with respect to the added mass, the added mass coefficient is varied from 80% to 120% of the initial calculated value. The time histories of the template with 80,90,100,110 and 120% of the added mass coefficient, in the 1-DOF and multiple-DOF analysis, are then compared according to the following:

1. Calculate the standard deviation of the time histories produced by the 1-DOF and multiple-DOF analysis.
2. Find the frequency peak of the FFT of the time histories produced by the 1-DOF and multiple-DOF analysis.
3. Investigate the additional signal that occur in the FFT of the 1-DOF time histories

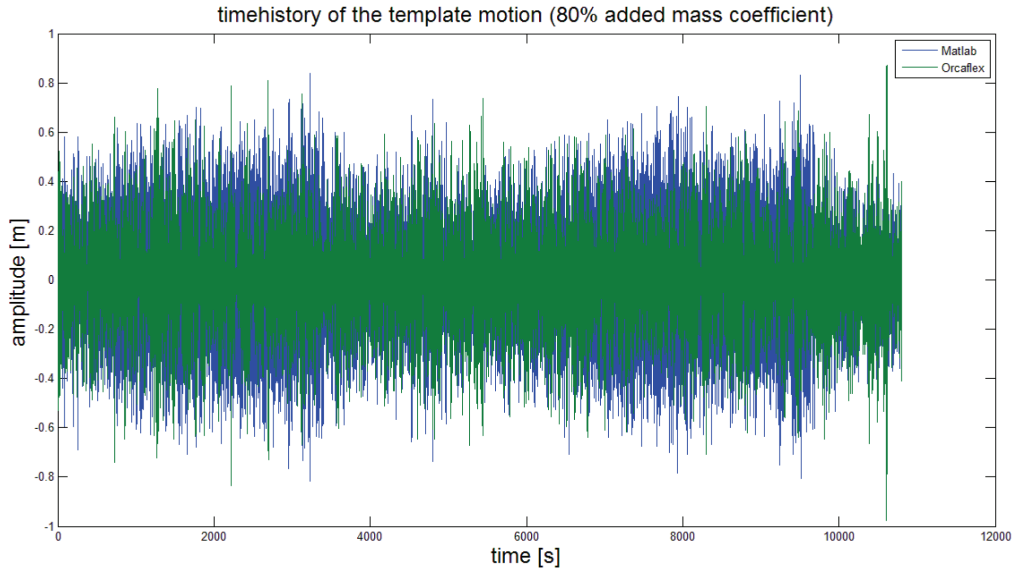


Figure 51 Time history of the template motion. 80% of the added mass coefficient.

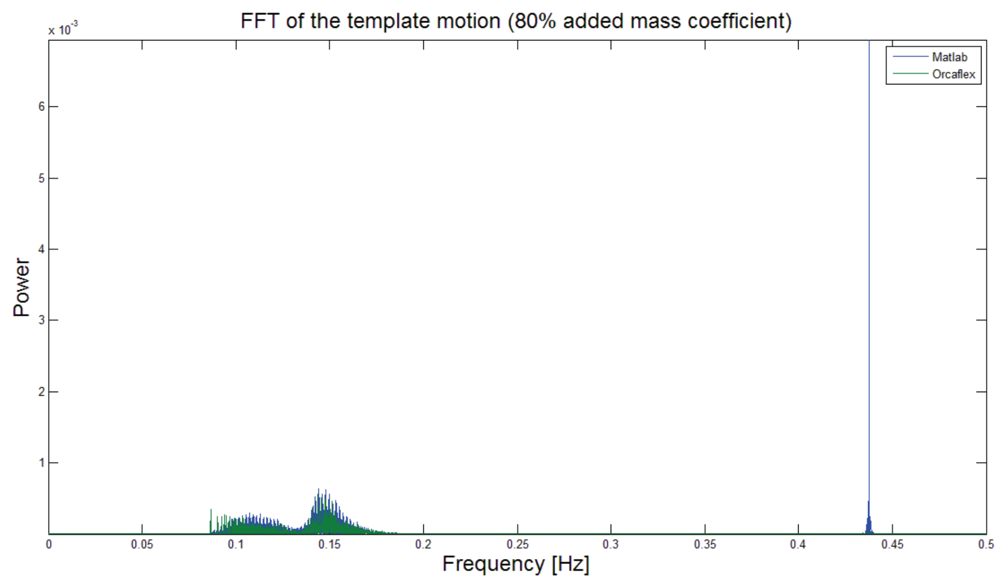


Figure 52 FFT of the template motion. 80% of the added mass coefficient.

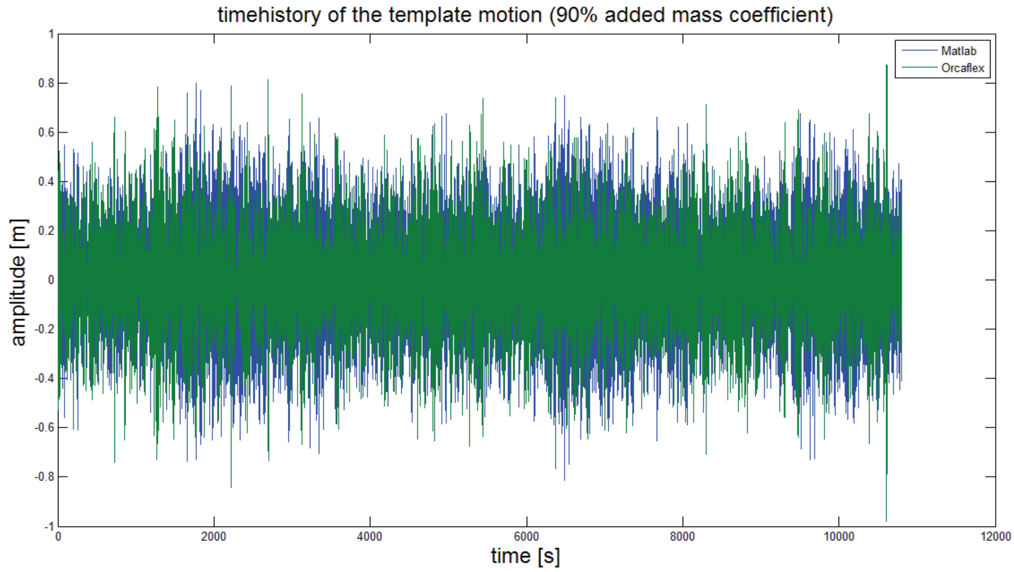


Figure 53 Time history of the template motion. 90% of the added mass coefficient.

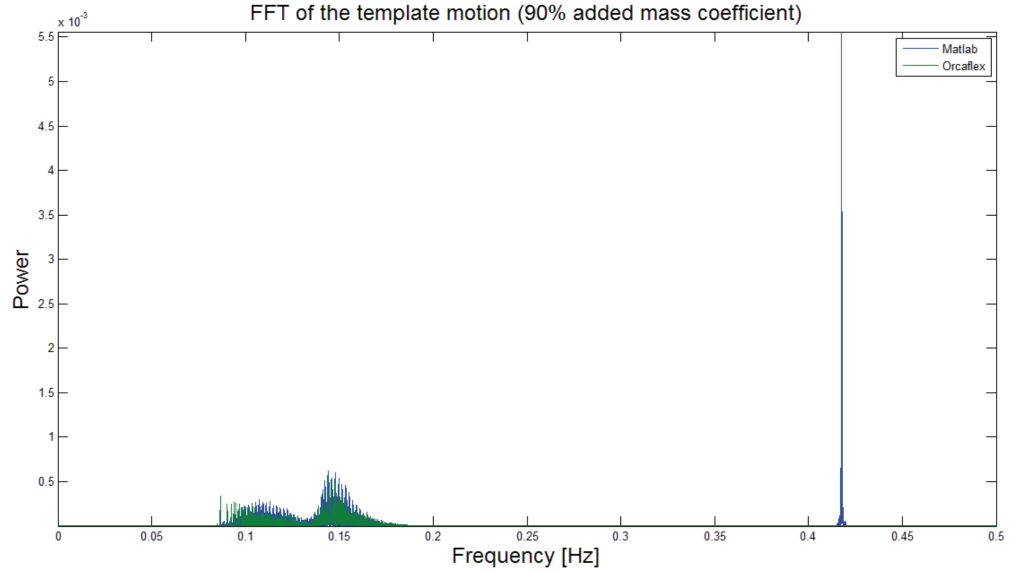


Figure 54 FFT of the template motion. 90% of the added mass coefficient.

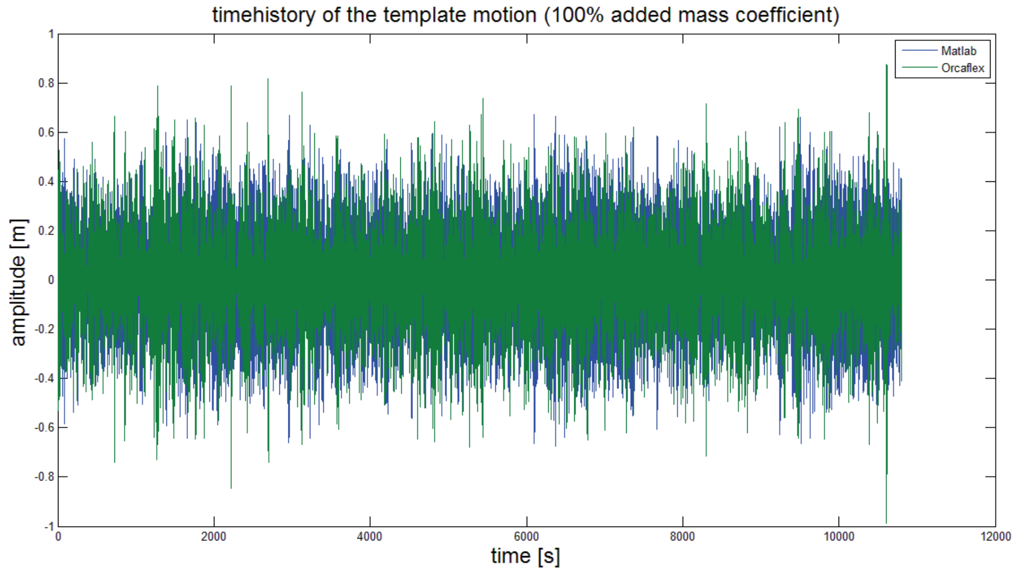


Figure 55 Time history of the template motion. 100% of the added mass coefficient.

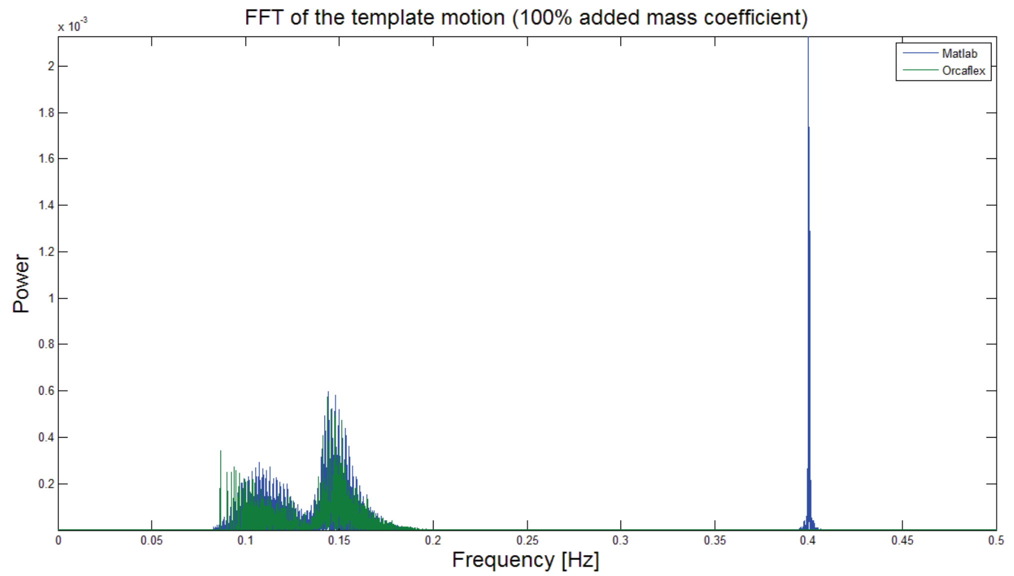


Figure 56 FFT of the template motion. 100% of the added mass coefficient.

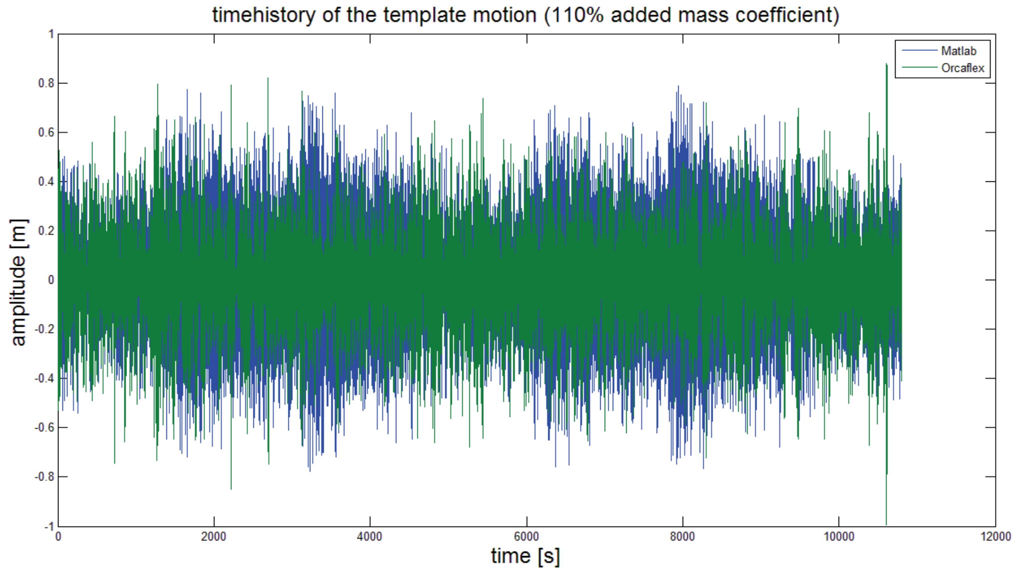


Figure 57 Time history of the template motion. 110% of the added mass coefficient.

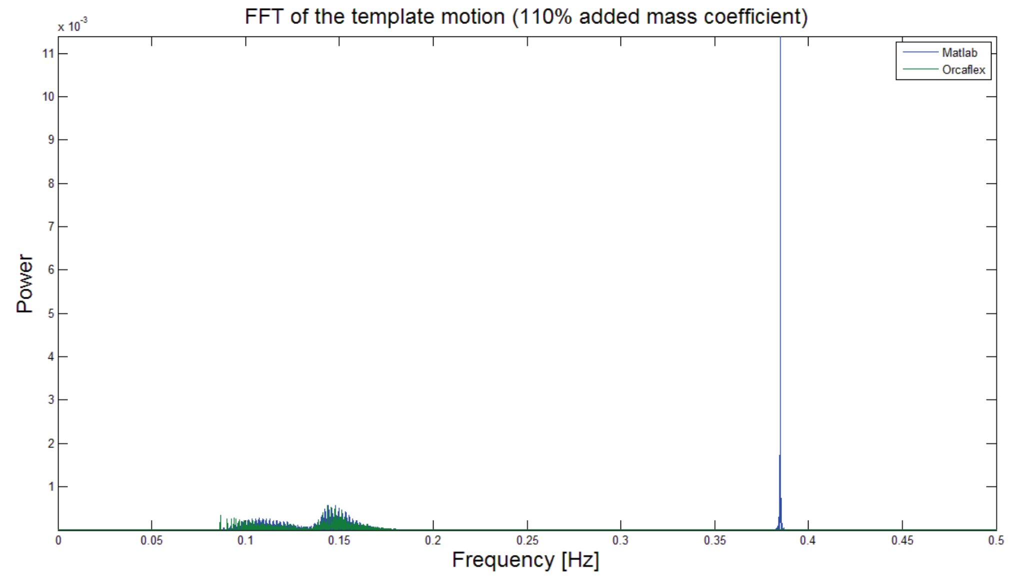


Figure 58 FFT of the template motion. 110% of the added mass coefficient.

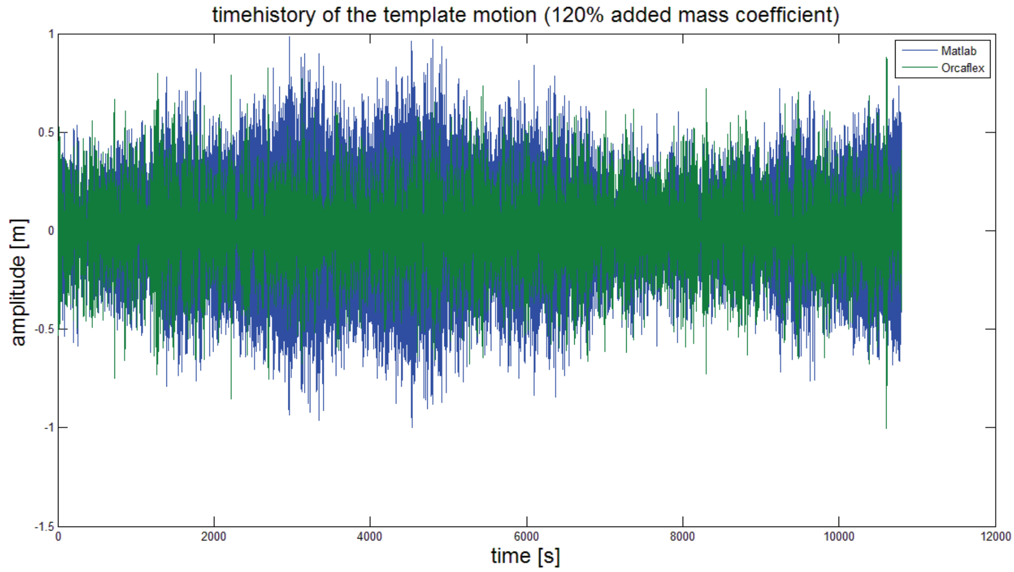


Figure 59 Time history of the template motion. 120% of the added mass coefficient.

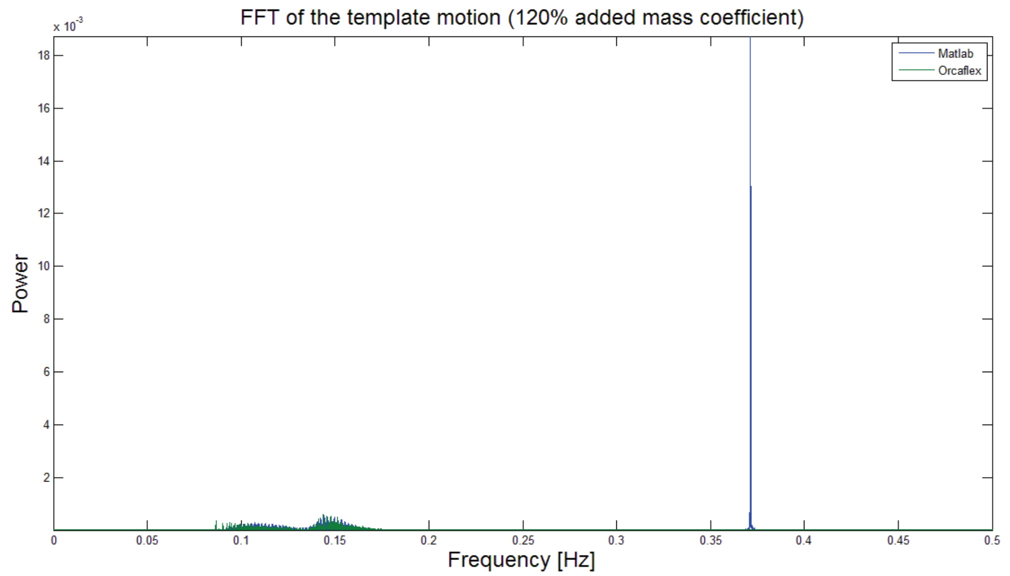


Figure 60 FFT of the template motion. 120% of the added mass coefficient.

To examine the peaks of the FFT for the template motion with varying added mass coefficient from 80-120% of the initial calculated value (Figure 52, Figure 54, Figure 56, Figure 58 and Figure 60 respectively) it is convenient to plot these in separate graphs. These plots can be seen in appendix 12.7 .The standard deviations of the time histories and the peaks of the FFT plots are calculated in Matlab.

Matlab provides the following results in comparison of the time histories of the motion of the template in the 1-DOF and multiple-DOF analysis:

	Standard deviation 1-DOF	Frequency peak 1-DOF (FFT)	Standard deviation multiple-DOF	Frequency peak multiple-DOF (FFT)
80% of initial added mass coefficient	0,2292	0,1441 [Hz]	0,2123	0,1436 [Hz]
90% of initial added mass coefficient	0,2102	0,1441 [Hz]	0,2129	0,1436 [Hz]
100% of initial added mass coefficient	0,1982	0,1441 [Hz]	0,2136	0,1436 [Hz]
110% of initial added mass coefficient	0,2262	0,1441 [Hz]	0,2143	0,1436 [Hz]
120% of initial added mass coefficient	0,2669	0,1441 [Hz]	0,2150	0,1436 [Hz]

Table 6 Results of The parametric study of the template motion with varying added mass coefficient.

	Additional frequency peak 1-DOF (FFT)	Power of the additional frequency peak. 1-DOF (FFT)
80% of initial added mass coefficient	0,4386 [Hz]	0,0070
90% of initial added mass coefficient	0,4171 [Hz]	0,0056
100% of initial added mass coefficient	0,4003 [Hz]	0,0021
110% of initial added mass coefficient	0,3860 [Hz]	0,0114
120% of initial added mass coefficient	0,3717 [Hz]	0,0187

Table 7 Characteristics of the additional frequency peak in the 1-DOF analysis (FFT), with varying added mass coefficient.

Frequency peak

The frequency peak tells us at which frequency the signal is at its strongest. From Table 6 we can observe that the frequency peaks of the main signal in the 1-DOF and multiple-DOF analysis are close to the same and remains the same with varying added mass coefficients.

Standard deviation

The standard deviation of the time histories shows how much variation there is from the mean. As observed in Table 6 the standard deviation in the time history produced by the multiple-DOF analysis slightly increases as the added mass coefficient increases. The standard deviation of time history produced in the 1-DOF analysis changes rapidly and does not increase with the added mass coefficient as it does for the multiple-DOF analysis. It decreases from 80% - 100% and increases from 100% - 120%.

Time history

The different time histories show a large difference in motion of the template in the 1-DOF and multiple-DOF analysis. Especially for 120% of the initial added mass coefficient (Figure 59) the amplitude of motion is a lot higher in the 1-DOF analysis. It seems as if the template motion experiences a dynamic amplification due to the increase of added mass. This leads us to presume that there is a problem with the numerical integration and most likely in how the damping is modeled.

Additional frequency peak

As observed in Figure 52, Figure 54, Figure 56,Figure 58 and Figure 60 there is an additional frequency peak in the FFT of the 1-DOF time history of the template. This additional frequency peak is the signal of the oscillation problem discovered in chapter 7.1.

The power of this additional frequency peak has the same behavior as the standard deviation. It decreases from 80% - 100% and increases from 100% - 120%. By identifying and solving this oscillating problem we will also eliminate the problem of the standard deviation behavior.

8.1.1 Solution of the oscillating problem

We have presumed that the oscillating problem discovered in chapter 7.1 has its roots in the numerical integration scheme used in the 1-DOF program, and most likely in how the damping is implemented.

There is a difference in how the numerical integration is done in the two different programs. The 1-DOF program uses the Newmark beta method while Orcaflex uses a combination of an explicit (forward Euler) and an implicit method (generalized alpha integration). Both forward

Euler and the generalized alpha integration are generalizations of the Newmark beta method, and the difference of these methods is the assumption of how the acceleration is modeled. The difference in the numerical integration may provide small differences in the results, but the oscillating problem was found to be in the implementing of damping in 1-DOF program.

The Morrison force is implemented in the external force acting on the template, and should be sufficient to model the correct damping. As the behavior of the template indicates that we don't have sufficient damping, an artificial damping is introduced.

We recall from chapter 3.1.9:

The parameter γ determines the accuracy of the Newmark beta method by deciding if the method has an artificial damping.

$$\gamma > \frac{1}{2} \rightarrow \text{Positive artificial damping. When the time step increases, the amplitude decreases}$$

$$\gamma < \frac{1}{2} \rightarrow \text{Negative artificial damping. When the time step increases, the amplitude increases}$$

$$\gamma = \frac{1}{2} \rightarrow \text{No Artificial damping}$$

By choosing γ equal to 1, the template motion with a positive artificial damping was calculated. This solved the oscillating problem, removed the additional frequency peak (Figure 62) and eliminated the problem of dynamic amplification (Figure 61).

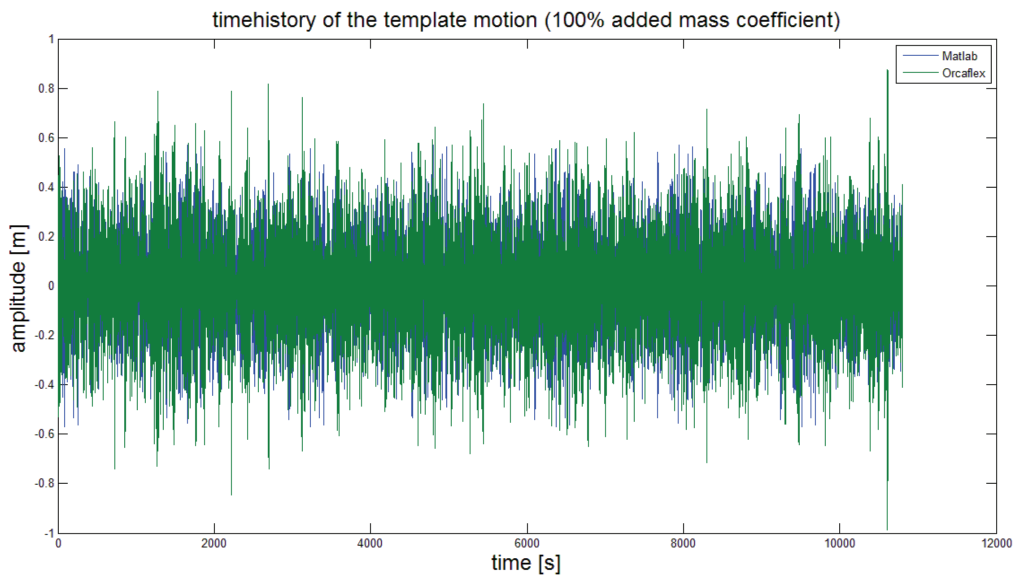


Figure 61 Time history of the template motion, after introducing an artificial damping in Newmark beta. 100% of the added mass coefficient.

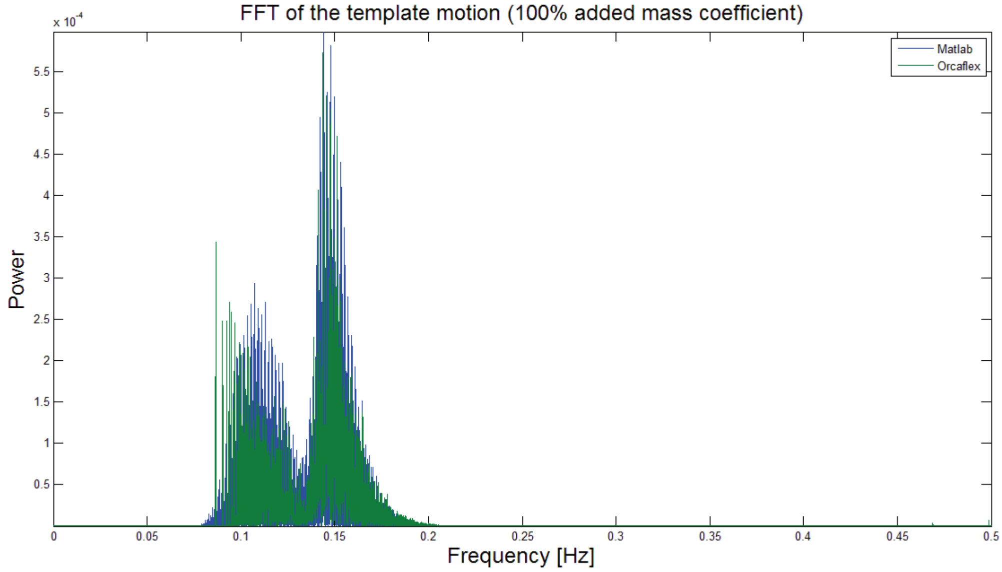


Figure 62 FFT of the template motion, after introducing an artificial damping in Newmark beta. 100% of the added mass coefficient.

After introducing the artificial damping in the Newmark beta method, a new parametric study of the template motion with varying added mass coefficients was performed.

	Standard deviation 1-DOF	**Standard deviation 1-DOF with artificial damping**	**Frequency peak 1-DOF (FFT)**	**Standard deviation multiple-DOF**	**Frequency peak multiple-DOF (FFT)**
80% of initial added mass coefficient	0,2292	0,1835	0,1441 [Hz]	0,2123	0,1436 [Hz]
90% of initial added mass coefficient	0,2102	0,1811	0,1441 [Hz]	0,2129	0,1436 [Hz]
100% of initial added mass coefficient	0,1982	0,1787	0,1441 [Hz]	0,2136	0,1436 [Hz]
110% of initial added mass coefficient	0,2262	0,1761	0,1441 [Hz]	0,2143	0,1436 [Hz]
120% of initial added mass coefficient	0,2669	0,1735	0,1441 [Hz]	0,2150	0,1436 [Hz]

Table 8 Results of the parametric study of the template motion, after introducing an artificial damping in Newmark beta.

The results of the parametric study of the template motion (Table 8) shows that also the strange behavior of the standard deviation has been eliminated. However the behavior of the standard deviation from the 1-DOF analysis slightly decreases with increasing added mass. This is the opposite behavior of the multiple-DOF analysis and may be caused by introducing the artificial damping.

A 3D plot of the parametric study of the template motion in the 1-DOF analysis shows that the results seem reasonable.

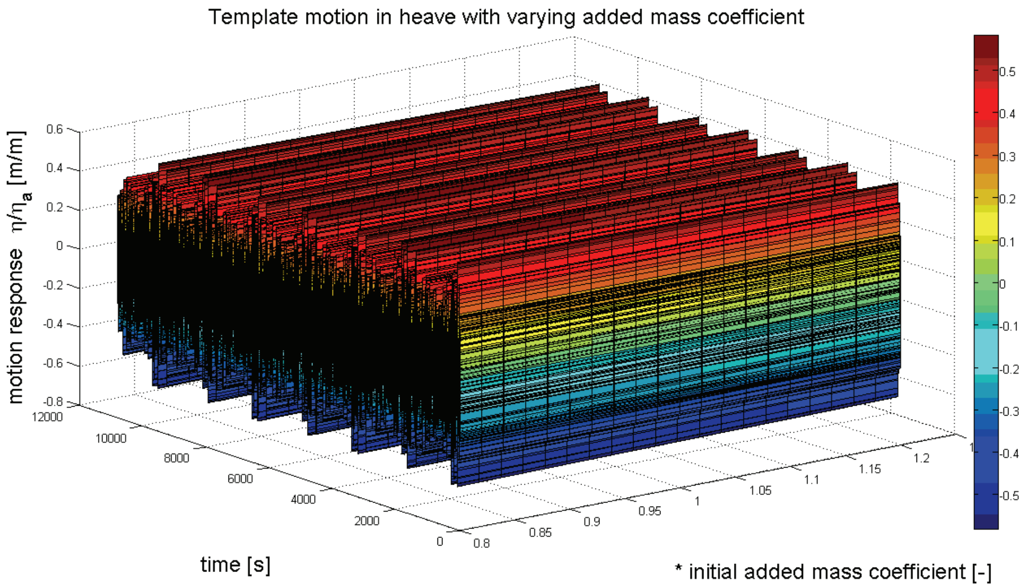


Figure 63 3D plot of the parametric study of the template motion in the 1-DOF analysis after introducing an artificial damping.

All results from comparison of the time histories and parametric study of the template motion with an artificial damping can be found in appendices.

- Results of comparing regular waves after introducing an artificial damping – Appendix 12.4
- Results of comparing irregular waves after introducing an artificial damping – Appendix 12.6
- Results of the parametric study of the template motion with varying added mass coefficient. After introducing an artificial damping – Appendix 12.8

8.2 Parametric study of the offset angle

The offset angle is both in the 1-DOF program and Orcaflex calculated from a static analysis and plotted as a function of the towing velocity. By varying the drag coefficient from 80% to 120% of the initial calculated value, we can observe the effect of increased or decreased drag on the template.

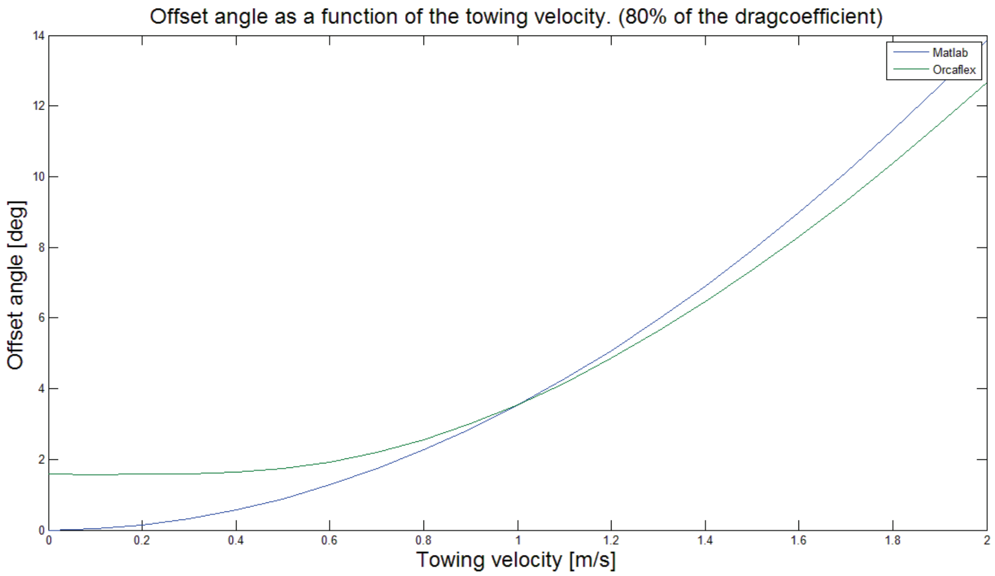


Figure 64 Offset angle as a function of the towing velocity. 80% of the initial drag coefficient.

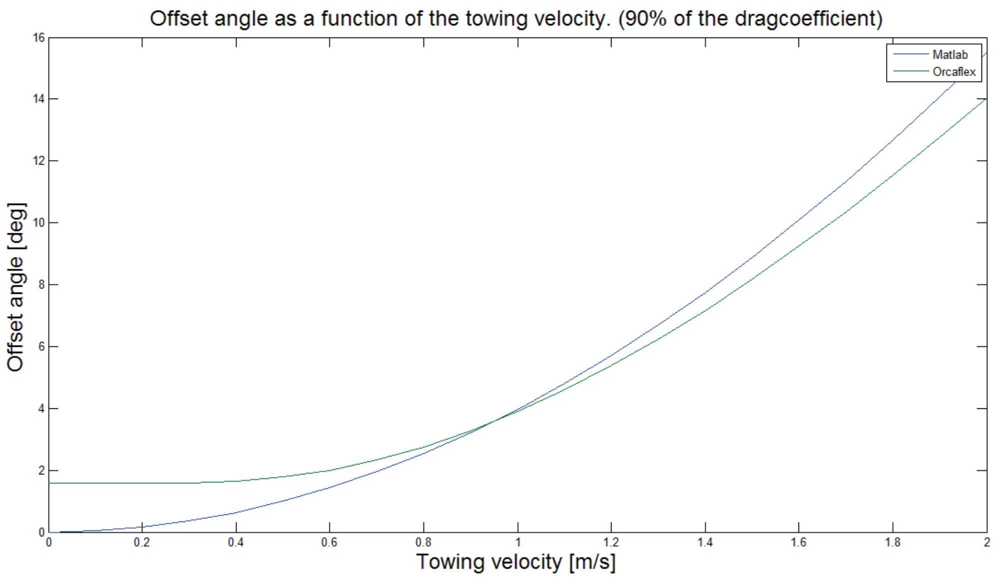


Figure 65 Offset angle as a function of the towing velocity. 90% of the initial drag coefficient.

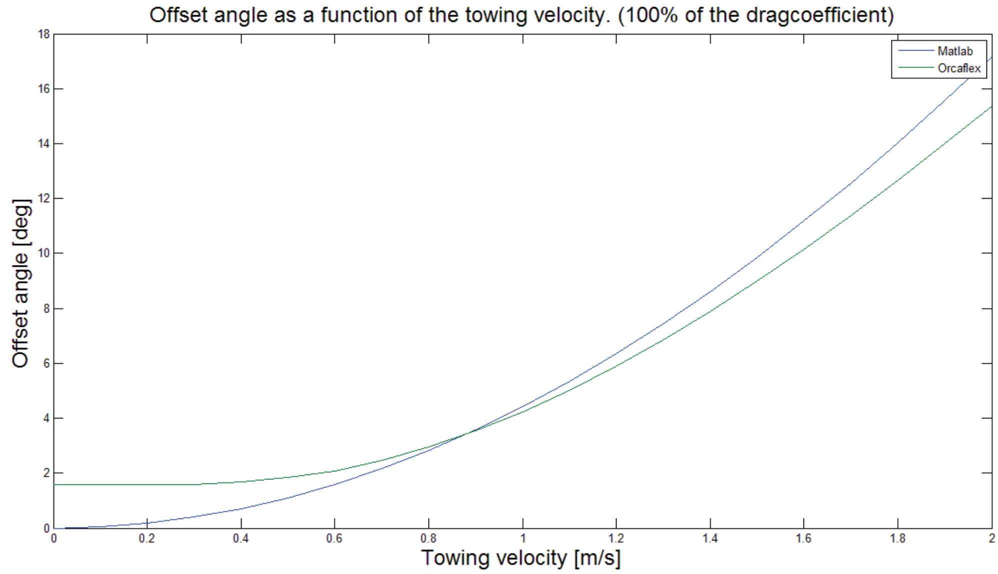


Figure 66 Offset angle as a function of the towing velocity. 100% of the initial drag coefficient.

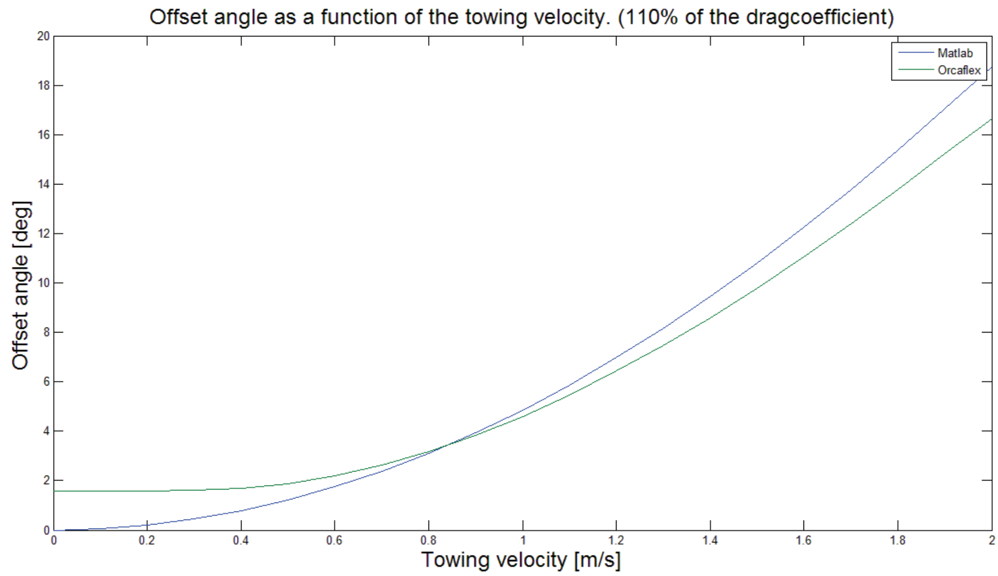


Figure 67 Offset angle as a function of the towing velocity. 110% of the initial drag coefficient.

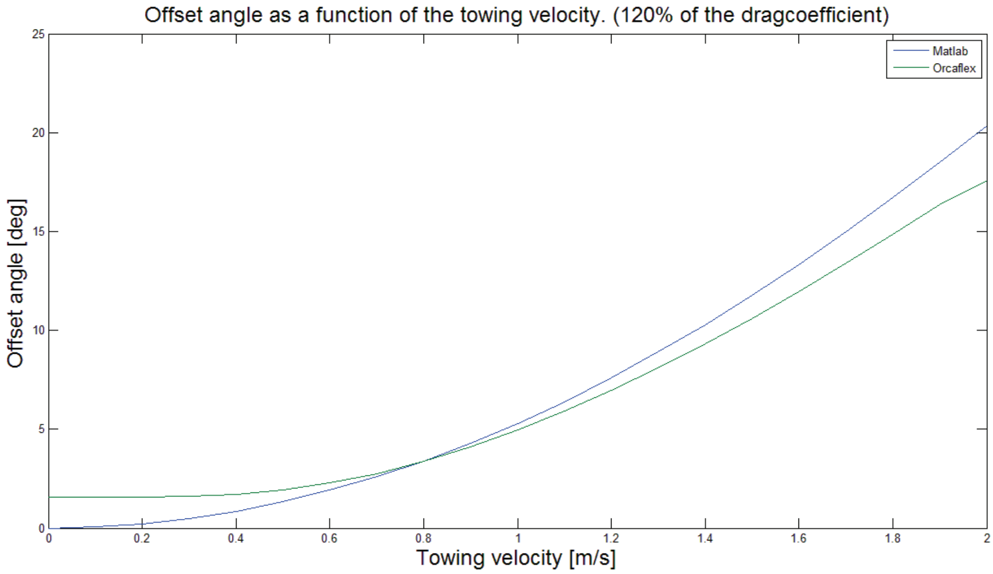


Figure 68 Offset angle as a function of the towing velocity. 120% of the initial drag coefficient.

The results plotted in Figure 64 to Figure 68 allow us to observe the following:

- The multiple-DOF analysis has an initial offset angle that is caused by non-symmetry in the distribution of weight of the template. This is confirmed by a static check of the tension in the wires. The difference in tension in the four sling wires (Figure 28) results in an initial offset angle.
- The offset angle in the 1-DOF analysis increases rapidly with the towing speed
- The shape of the plots remain the same with varying drag coefficients

The 1-DOF program uses the initial calculated reference area of the template. This is the area with no offset angle as illustrated in Figure 70. By increasing the towing velocity we get an increased offset angle and the reference area will change as shown in Figure 71. Since this change of reference area is calculated in Orcaflex, but neglected in the 1-DOF program, it is presumed that this is the reason of the differences in the results.

To do a further check of the calculations in the 1-DOF program a simple model is made in Orcaflex with the same configuration as in the 1-DOF program, i.e. the template is modeled as a lump mass and the rigging is replaced by an equivalent wire.

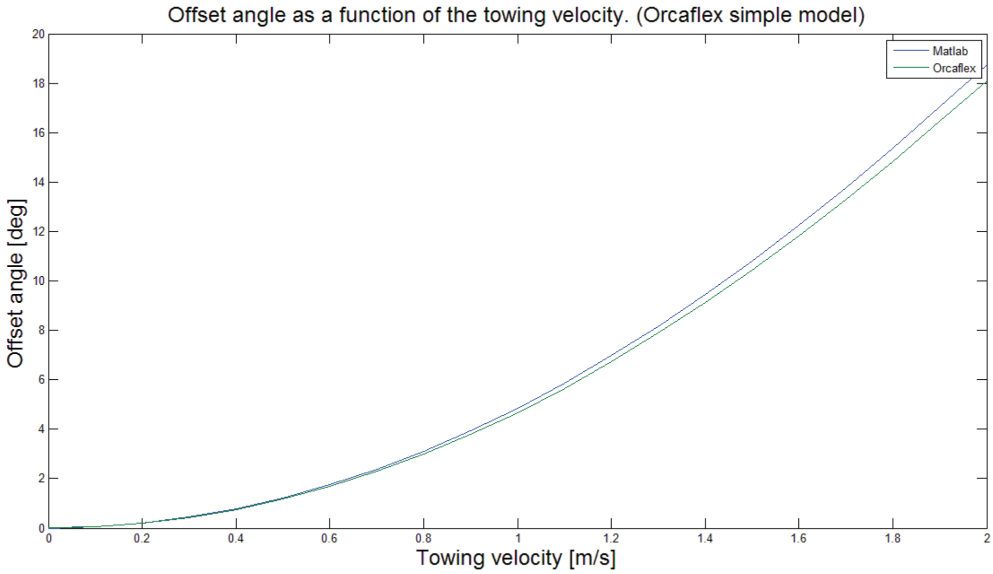


Figure 69 Offset angle with a simple model used for the multiple-DOF analysis

Figure 69 shows that the offset angle in the 1-DOF and multiple-DOF analysis only deviate with a small angle for high towing velocities. This small deviation is because the lump mass modeling the template in Orcaflex is modeled as a 6D buoy. This means that it can move in all 6 DOF and we get a small effect of the changing reference area as shown in Figure 70 and Figure 71 .

If we compare the results in Figure 64 to Figure 68 for the design towing velocity of 1,5 meters per second, the difference in the 1-DOF and multiple-DOF analysis is satisfying and the results are well inside the limits of the allowable offset angle (Table 9).

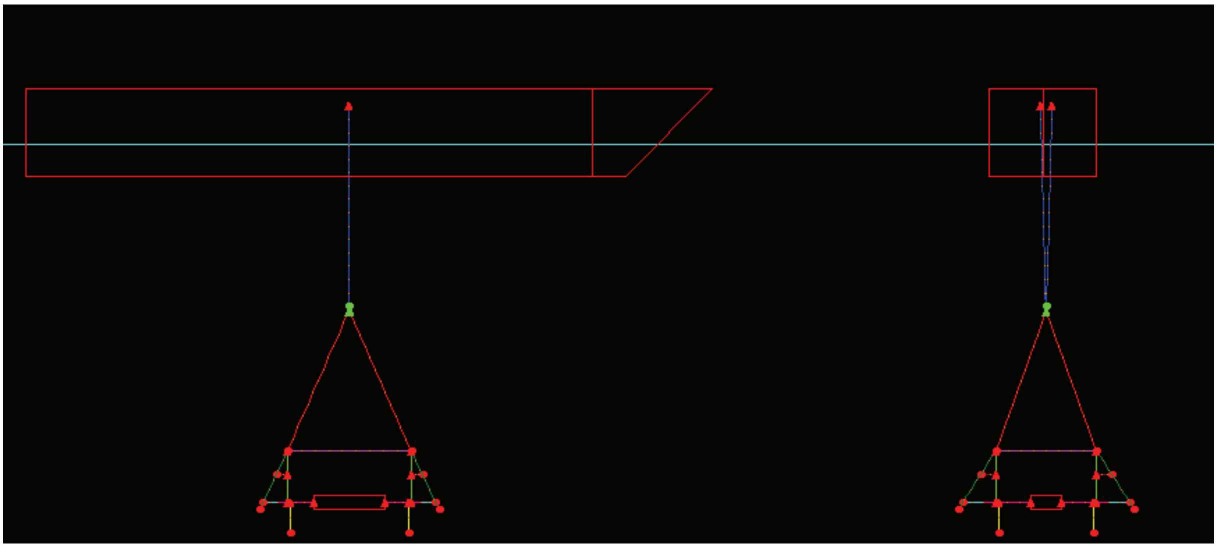


Figure 70 Reference area of the template with no offset angle

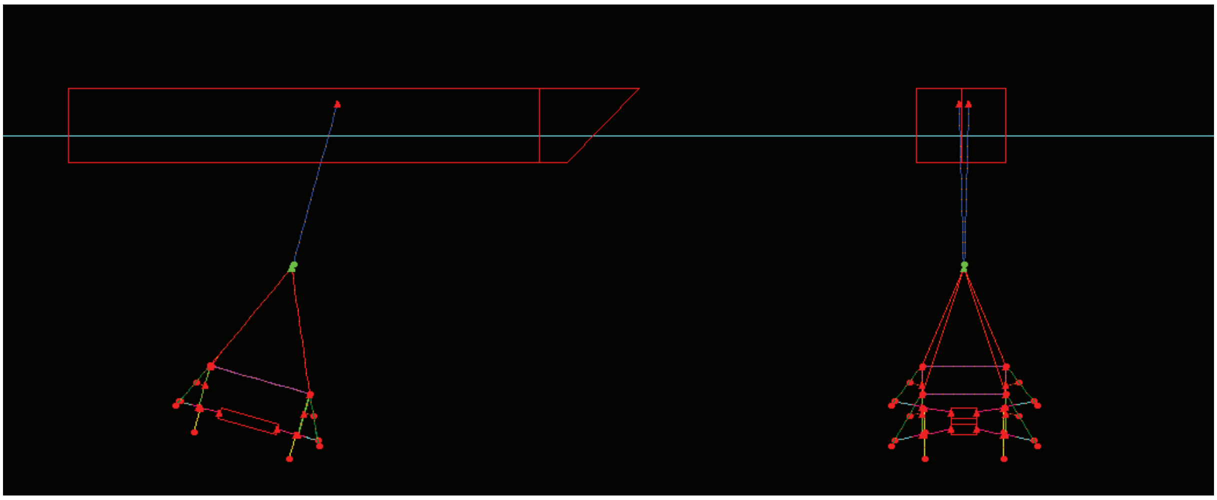


Figure 71 Reference area of the template with an offset angle

	Allowable longitudinal offset angle [deg]
Pennant 1	22

Table 9 Allowable offset angle. Reference [9].

9 Extreme tension in suspension line

Extreme statistics is useful to investigate the worst case scenarios of a given set of data. In this chapter we will investigate the extreme tension in one suspension line (one pennant wire, Figure 28).

To do an extreme statistics analysis we need a large set of samples, in this case 20. Each sample is a 3 hour analysis with 1000 wave components of the system described in chapter 5.1.

The expected extreme tension in the suspension line is calculated in three different ways.

9.1 Average of the maximum peaks

From each of the 20 samples, the maximum value is extracted. The expected extreme value is the average of these 20 values.

9.2 Weibull distributed sample

Each of the 20 samples is assumed to follow a Weibull distribution and an expected extreme value is calculated for each of the samples as described in chapter 4.1. The expected extreme value in the suspension line is the average of the expected extremes for each sample.

The number of peaks of the sample are greater than 1000, this gives according to Table 4 a constant C2=0,57722.

This is inserted into equation 4.1.7 to obtain the expected extreme value of each sample.

The expected extreme value is the average of the expected extreme for each sample:

$$\mu_{ex} = \frac{1}{N}\sum_{i=1}^{N}\left(\mu_y\right)_i$$

$N =$ Number of samples

$\mu_y =$ Expected extreme value of one sample

$\mu_{ex} =$ Average of the expected extreme values (9.2.1)

As mentioned in chapter 4, it is important to do the Weibull fit with only the global peaks of the data. To see the effect of this the expected extreme value is calculated for both cases. With all local peaks and only the global peaks.

9.3 Gumbel distributed maxima

As in chapter 9.1 the maximum value of each of the 20 samples are extracted. These 20 maxima are assumed to follow a Gumbel distribution and the expected extreme value can be found according to chapter 4.2. To calculate the expected extreme value we recall equation 4.2.5 and with a sample size of N=20 we insert the constants C1=1,06283 and C2=0,52355 from Table 4.

9.4 Results

The results the extreme statistics calculations are calculated in Matlab and presented in Table 10.

	All local peaks above the mean	Only global peaks
(1) Average of the 20 maximum peaks	1694,3 [kN]	1694,3 [kN]
(2) Average of the 20 expected extremes from a Weibull distributed set of data	1689,3 [kN]	1711,2 [kN]
(3) Expected extreme from a Gumbel distributed set of maxima	1697,2 [kN]	1697,2 [kN]

Table 10 Results of the extreme tension in the suspension line

From Table 10 we see that the average of the 20 maximum peaks is 1694,3 kN (1). It is natural to see how the other methods compare to this value.

- (2) Average of the 20 Expected extremes from a Weibull distributed set of data: Underestimates the extreme value when all local peaks above the mean are part of the calculations. Overestimates when only the global peaks are part of the calculations.
- (3) Expected extreme from a Gumbel distributed set of maxima. Gives a slightly higher value than the average of the 20 maximum peaks.

On the basis of these comparisons it can be concluded that the best option is to calculate the expected extreme from a Gumbel distributed set of maxima as it only give a slightly higher value than the average of the 20 maximum peaks.

The average of the 20 expected extremes from a Weibull distributed set of data with only the global peaks is also a good option. It overestimates the expected extreme compared to the Gumbel distributed set of maxima, but for design purposes this overestimating is relatively small and gives a conservative estimate of the expected extreme tension.

10 Conclusion

The main objective of this thesis was to extend and improve the 1-DOF program developed in the project thesis (reference[7]). This turned out to be an extensive task, but has made the program able to analyze the towing operation done in the Tyrihans project performed by Subsea 7. The results have been compared with the multiple-DOF results produced by the Orcaflex model for the same project. A comparison of these results led to a conclusion that the 1-DOF program is suitable for an initial simple check of a subsurface towing configuration according to the subsea 7 method. However the program needs to be tested for different projects and compared to different programs to check its reliability.

10.1 Comparison of time histories in the 1-DOF and multiple-DOF analysis

As the time histories are based on irregular waves, the comparison is done by calculating the standard deviation and performing a FFT to examine the frequency at which the signal is at its strongest.

The results of comparing the time histories produced by the 1-DOF and multiple-DOF analysis indicate the following conclusions:

- The frequency at which the signal is strongest for the time histories of the waves is the same and the shapes of the FFT are similar. This implies that the method to model the waves in the 1-DOF program gives results close to the ones modeled by Orcaflex.

- The frequency at which the signal is strongest for the time histories of the vessel motion is the same and the shapes of the FFT are similar. This implies that the method to model the vessel motion in the 1-DOF program gives results close to the ones modeled by Orcaflex.

- The frequency at which the main signal is strongest for the time histories of the template motion is the same, and the shapes of the FFT are similar. However the FFT of the time history in the 1-DOF analysis gives an additional frequency peak that is quite high. This frequency peak is due to an oscillating motion of the motion of the template. Due to this problem, the time history- and FFT of the template motion has been investigated further in the parametric study of the template motion.

10.2 Parametric study of the template motion

Because of the dynamic amplification of the templates’ motion and the additional frequency peak in the FFT of the time history produced by the 1-DOF analysis, a parametric study of the motion of the template with varying added mass coefficients has been performed.

The results of the parametric study of the template motion in the 1-DOF and multiple-DOF analysis indicate the following conclusions:

- The frequency peak and shape of the main signal are the same and remain the same with varying added mass coefficients.

- The 1-DOF time history experiences a dynamic amplification with increasing or decreasing the added mass from 100% of the initial value.

- The frequency at which the additional frequency peak occurs in the FFT of the 1-DOF time history decreases with increasing added mass. This additional frequency peak describes the oscillation of the templates motion, and is due to an insufficient damping. This was solved by introducing an artificial damping in the Newmark beta method in the 1-DOF program.

- After eliminating the additional frequency peak, the results from the 1-DOF analysis are satisfyingly close to the multiple-DOF results.

10.3 Parametric study of the offset angle

The offset angle in the 1-DOF and multiple-DOF analysis is compared by doing a parametric study with varying drag coefficients, from 80% to 120% of the initial calculated value. There is also done a comparison of the offset angle after creating a simple model in Orcaflex. This model has the same set up as the 1-DOF model, with the template modeled as a lump mass with all the hydrodynamic properties and the rigging modeled as an equivalent wire.

The results of the parametric study of the offset angle indicate the following conclusions:

- The multiple-DOF analysis gives has an initial offset angle due to non symmetry in the templates distribution of weight. This is confirmed by observing that the four sling wires have slightly different tensions.

- The offset angle has been plotted as a function of the towing velocity for drag coefficients varying from 80% to 120% of the initial calculated value. The shape of these plots in the 1-DOF and multiple-DOF analysis remains the same and the values at the towing velocity of 1,5 m/s are close, with the 1-DOF results being slightly more conservative. The difference in values is due to the change of drag area. The 1-DOF analysis is done with a constant drag area whilst the multiple-DOF analysis allows the drag area to change with the offset angle.

- Comparing the offset angle in the 1-DOF analysis with the multiple-DOF analysis of the simple model created in Orcaflex gives almost the same results. It is still a difference due to a small change of the drag area in the multiple-DOF analysis. This makes the values in the 1-DOF analysis slightly more conservative.

10.4 Extreme tension in suspension line

20 three hour analysis have been performed in Orcaflex and the extreme statistics have been calculated. The calculations have been done for all the peaks above the mean of each sample and for only the global peaks. This has been done with three different methods.

1. Average of the 20 maximum peaks. One for each sample.
2. Average of the 20 expected extremes from a Weibull distributed set of data. One expected extreme for each sample.
3. Expected extreme from a Gumbel distributed set of maxima. One maximum for each sample.

The results of the extreme statistics calculations of the 20 samples created in Orcaflex indicate the following conclusions:

- The average of the 20 Expected extremes from a Weibull distributed set of data underestimates the extreme value when all local peaks above the mean are part of the calculations. It overestimates when only the global peaks are part of the calculations. This alternative can be used for a conservative estimate of the expected extreme tension in the suspension line as long as only the global peaks are part of the sample data.

- The expected extreme from a Gumbel distributed set of maxima gives a slightly higher value than the average of the 20 maximum peaks and is the best alternative to calculate the expected extreme tension of the suspension line.

10.5 Further work

The conclusions in this chapter lead to the following recommendations for further work:

- The numerical integration used to model the template motion in the 1-DOF program and Orcaflex is different. The significance of this difference must be investigated.

- The implementation of the damping of the template in the Newmark beta method in the 1-DOF program is done by introducing the Morrison force. This proved to be insufficient and was fixed by adding an artificial damping. The 1-DOF program should be improved to eliminate the need of this artificial damping. After this is done the behavior of the standard deviation of the time histories should be checked.

- The 1-DOF program must be compared to other programs and projects to see if it gives reasonable results.

- The 1-DOF program must be extended and improved further to make it more versatile for other projects, as well as reduce the calculation time.

11 Bibliography

[1] The pencil buoy method – A subsurface transportation and installation method; T. Risoey, H. Mork, H. Johnsgard, and J. Gramnaes, Aker Marine Contractors.

[2] Deep water pipeline and riser installation by the combined tow method, by Alf Roger Hellestø, Daniel Karunakaran and Ove T Gudmestad. Subsea 7, Statoil and University of Stavanger, Norway.

[3] Combined tow method for deep water pipeline and riser installation. Alf Roger Hellestø; Daniel Karunakaran and Trond Gytten. Subsea 7; Ove Tobias Gudmestad, Statoil and University of Stavanger.

[4] Lessons learnt from lifting operations and towing of heavy structures in North sea. Kenneth Aarset, Arunjyoti Sarkar and Daniel Karunakaran, Subsea 7.

[5] Wet tow and installation of subsea templates. Kenneth Aarseth/Subsea 7, Tale Kristine Ulstein/Subsea 7, Håvard Strand/Subsea 7.

[6] Subsea 7 – Wet tow & installation of templates – Norsk. Powerpoint of Kenneth Aarset.

[7] T.Olsen 2010 – *Submerged towing of a template, project thesis by Torbjørn Aakerøy Olsen*

[8] Multipurpose icebreaker MSV "Botnica". Motion characteristics data. DSND Subsea AS, march 1998. Author: Wojciech Kauczynski.

[9] Tyrihans field development Contract No. 4501139048. Document No: C089-UAK-N-CA-0001, Structure tow analysis. Subsea 7 – Acergy consortium.

[10] *Sea loads on ships and offshore structures by O.M. Faltinsen*

[11] Nielsen 2007 – *Lecture notes in marine operations by Finn Gunnar Nielsen. NTNU*

[12] Pettersen 2007 – *TMR 4247 Marin teknikk 3 hydrodynamikk by Bjørnar Pettersen*

[13] Hughes 2008 – *The finite element method, linear static and dynamic finite element analysis, by Thomas J.R. Hughes*

[14] Langen & Sigbjørnsen – *Dynamisk analyse av konstruksjoner by Ivar Langen and Ragnar Sigbjørnsen, Sintef department of structures*

[15] Myrhaug 2007 – *TMR4180 Marin Dynamikk, uregelmessig sjø by Dag Myrhaug*

[16] Larsen 2009 – *TMR 4180 Marin dynamikk by prof. Carl M. Larsen*

[17] OrcaFlex manual
http://www.orcina.com/SoftwareProducts/OrcaFlex/Documentation/OrcaFlex.pdf

[18] Tyrihans field development Contract No. 4501139048. Document No: C089-UAK-N-KA-0003, Structure installation procedure. Subsea 7 – Acergy consortium.

[19] Tyrihans field development Contract No. 4501139048. Document No: C089-UAK-N-CA-0002, Structure installation procedure. Subsea 7 – Acergy consortium.

[20] Nonlinear Dynamic Response and Reliability analysis of Drag-dominated Offshore Platforms, by Daniel N. Karunakaran.

[21] Probability & Statistics for Engineers &Scientists by Ronald E. Walpole, Raymond H. Myers, Sharon L. Myers and Keying Ye.

[22] An introduction to mathematical statistics and its applications by Richard J. Larsen and Morris L. Marx.

[23] Analyse av usikkerhet by T. Moan, N. Spidsøe and S. Haver. November 1980.

[24] Prediction of characteristic response for design purposes by Sverre Haver. 2010.

[25] Methods for estimating the parameters of the Weibull distribution by Mohammad A. Al-Fawzan.

[26] Estimation the shape, location and scale parameters of the Weibull distribution by Dr. Zouani Chikr el-Mezouar.

[27] www.mathworks.com

[28] Spectral analysis using th FFT. Brett Ninness, Department of electrical and computer engineering. The University of Newcastle, Australia.

[29] Prediction of Characteristic Response for Design Purposes. Sverre K. Haver. 2010

[30] Technical Note: A revised parametrisation of the Jonswap spectrum. R.M. Isherwood. Orcina Ltd, Plumpton Hall, Ulverston, Cumbria LA12 7QN

[31] Pipeline, Risers & Marine Engineering, Technical Note. Work Method/Guideline/Recommended Practice. Report No: KM02NA-P-TN-004 – Subsea 7

[32] Satelite tow out, powerpoint presentation. Towing of structures by Bjarte Landa, Tananger 14.03.05

[33] Heidrun Satelite tow out project. Structure tow and installation analysis. Document No: SUBS-H-RD-002, Subsea 7.

[34] Probability concepts in engineering planning and design. Volume 1 – Basic principles. Alfredo H.S. Ang & Wilson H. Tang

[35] Probability concepts in engineering planning and design. Volume 2 – Decision, Risk and Reliability. Alfredo H.S. Ang & Wilson H. Tang

[36] Stochastic theory of sea loads. Probabilistic modeling and estimation. Bernt J. Leira 2010

12 Appendix

12.1 Matlab routines

The matlab code in this appendix is the code for the whole 1-DOF program as well as all the Matlab routines used to do the comparison, parametric study and extreme statistics calculations.

The routines for the comparison of the time histories of the surface elevation, vessel motion and template motion only differ in what time history is the input file. For this reason only the routine for comparison of the surface elevation will be attached.

For the parametric study of the template motion the routines are almost identical for 80,90,100,110 and 120% of the initial added mass coefficient. The difference is what input files the routine reads. For this reason only the routines for 80% of the initial added mass coefficient will be attached.

For the extreme statistics calculations with all peaks, the calculations of each of the 20 samples are almost identical with only the choice of input file as a difference. For this reason only the routine for sample 1 will be attached. This is also the case for the extreme statistics with only the global peaks.

All the routines that are not attached in the appendices can be found on the attached CD.

12.1.1 Main program

12.1.1.1 Main.m

```
%-----------------------------------------------------------------------
% Program developed in the project and Master thesis of Torbjørn Aakerøy
Olsen
%-----------------------------------------------------------------------

clear all
clc

disp('******************************************************************' )
disp(' ' )
disp('                      Master thesis for Torbjørn Aakerøy Olsen.
Spring 2011 ' )
disp('              The program will guide you through all the steps to
obtain the results' )
disp(' ' )
disp('******************************************************************' )

disp(' ')
disp(' ')

param

clc

option1=input('Press enter to do the calculations and display a plot of the
horizontal offset:');
offset
proceed1=input('Press enter to proceed to the next calculation:');
clc

option2=input('Press enter to do the calculations and display a plot of
dynamic displacement and force:');
dof
proceed2=input('Press enter to proceed to the next calculation:');
clc

option3=input('Press enter to do the calculations and display a 3D plot of
dynamic displacement and force:');
dof3d
proceed3=input('Press enter to proceed to the next calculation:');
clc

fid = fopen('optionwavespec.txt');
optionwave = fscanf(fid, '%f\n');
fclose(fid);

if optionwave~=3
option4=input('To calculate and plot:\n The chosen wavespectrum \n The
transferfunction in heave for the vessel \n The time history of the wave \n
The time history of the heave motion of the vessel \n The time history of
```

```
the heave motion of the template \n The template and vessel displacement in
heave \n The absolute value of the motion difference between the vessel and
template in heave \n Press enter:');
tempres
proceed4=input('Press enter to proceed to the next calculation:');
clc
elseif optionwave==3
option4=input('To calculate and plot:\n The chosen wavespectrum \n The
transferfunction in heave for the vessel \n The time history of the wave \n
The time history of the heave motion of the vessel \n The time history of
the heave motion of the template \n The template and vessel displacement in
heave \n The absolute value of the motion difference between the vessel and
template in heave \n Press enter:');
tempresregular
proceed4=input('Press enter to end the program:');
clc
end

fid = fopen('optionwavespec.txt');
optionwave = fscanf(fid, '%f\n');
fclose(fid);

if optionwave~=3

option5=0;
while option5~=1 && option5~=2
option5=input('Press 1 to do a parametric study of the horizontal offset
with varying drag coefficient.\n Press 2 to proceed to the next
calculation:');
clc
end
if option5==1
dragparameter
proceed5=input('Press enter to proceed to the next calculation:');
end
clc

option6=0;
while option6~=1 && option6~=2
option6=input('Press 1 to do a parametric study of the motion difference in
heave between the \n vessel and the template with varying added mass
coefficient. Press 2 to end the program:');
clc
end
if option6==1
addedparameter
end
clc

end
```

12.1.1.2 Param.m

```
% ************************************************************************
% Routine:                  Param
% ------------------------------------------------------------------------
% Intension:                Allow the user to insert pre calculated values
%                           needed to do the analysis
%
% Method:                   Prompt the user for the values
%
% Parameter:                Description
% ------------------------------------------------------------------------
% m_t                       Mass of the template [kg]
% a33                       added mass in heave for template[kg]
% A_z                       Projected area of load (z-direction) [m^2]
% w0                        Submerged weight of the template [N]
% Cd                        Drag coefficient (x-direction) [-]
% U_max                     Maximum towing velocity velocity [m/s]
% A_x                       Refferance area on the template (x-direction)
[m^2]
% Cd_z                      Drag coefficient for template in z-direction [-]
% L                         Equivalent length of the cable [m]
% d                         Equivalent diameter of the cable [m]
% p_cable                   Equivalent density of the cable [kg/m^3]
% E                         Equivalent E-modulus of the cable [Pa]
% Hs                        Significant wave height [m]
% Tp                        Peak period [s]
% Gamma                     Constant in the jonswap spectrum
% time_w                    Duration of sea state [h]
% T                         Period of regular wave [s]
% H                         Wave height from trough to crest of regular wave
[m]
% tmax                      Length of analysis [s]
% timestep                  Timestep in the analysis [s]
% numfreq                   Number of frequencies/wave components [-]
% ------------------------------------------------------------------------
% Programmer:               Torbjørn Aakerøy Olsen
% Last changed:             26.05.2011
% ************************************************************************
clear all
clc

disp('#####################################################################');
disp('#                                                                   #');
disp('#          Matlab program developed in the master thesis of         #');
disp('#                Torbjørn Aakerøy Olsen spring 2011.                #');
disp('#      1 DOF analysis of a simplified subsurface towing system.     #');
disp('#                                                                   #');
disp('#####################################################################');

disp('*********************************************************************')
disp('*                                                                   *')
disp('*              Drawing of the towing system and axes                *')
disp('*                                                                   *')
disp('*********************************************************************')
disp('
')
```

```
disp('                              ^ Z
disp('                              |
disp('______________________________|_____________________________________
disp('|                             |                                    /
disp('|                             |----------> X                      /
disp('|                             |                                 /
disp('|_____________________________|______________________________/
disp('                              |
disp('                              |
disp('                              |
disp('                              |
disp('                              |
disp('                              |
disp('                              |
disp('                     _________|________
disp('                    |                  |
disp('                    |     Template     |
disp('                    |__________________|


proceedtoparameters=input('Press enter to proceed:');
clc

disp('######################################################################
disp('#
disp('# Before the analysis can be done some information about  the system
is needed
disp('#
disp('######################################################################

option1=0;
option2=0;
option3=0;
option4=0;
option5=0;
option6=0;
option_wavespec=0;

while option1~=1 && option1~=2
disp('*******************************************************************' )
disp(' ' )
disp('                       Template
disp('  ' )
disp('*******************************************************************' )
disp(' ' )
disp(' ' )
disp('The default values for added mass in heave, projected area (x-
direction),projected area of load (z-direction),  ')
disp('submerged weight and drag coefficient in x- and z-direction of the
template is:' )
disp('a33=1258.31*1000 kg, A_x=267.4 m^2, A_z=287.463 m^2, W0=2216000 N,
Cd_x=1.248 [-], Cd_z=1.166 [-]  ' )
disp(' ' )
disp(' ' )
option1=input('If you want to use the default values, press 1. If you want
to select own values, press 2:');
clc
```

```
end

if option1==1
    a33=1258.31*1000;
    A_x=267.4;
    A_z=287.463;
    w0=2216000;
    Cd_x=1.248;
    Cd_z=1.166;

elseif option1 == 2
    a33=input('Added mass [kg]                                    :
(default value 1258.31*1000kg)');
    A_x=input('Projected area in x-direction [m^2]                :
(default value 267.4 [m^2])');
    A_z=input('projected area in z-direction [m^2]                :
(default value 287.463 [m^2])');
    w0=input('Submerged weight of the template [N]                :
(default value 2216000 [N])');
    Cd_x=input('Drag coefficient of the template in x-direction [-]:
(default value 1.248 [-])');
    Cd_z=input('Drag coefficient of the template in z-direction [-]:
(default value 1.166 [-])');
end

m_t=w0/9.80665;

fid = fopen('tempparam.txt','w');
fprintf(fid, '%f %f %f %f %f %f %f\n' , a33, A_x, A_z, w0, m_t, Cd_x,
Cd_z);
fclose(fid);
clc

while option2~=1 && option2~=2
disp('*******************************************************************' )
disp(' ' )
disp('                             Wire ' )
disp('  ' )
disp('*******************************************************************' )
disp(' ' )
disp(' ' )
disp('The default values for equivalent length, diameter,density and E-
modulus of ')
disp('the wire is: L=51.763m , d=0.146m , p=7850kg/m^3, E=30.3*10^9 Pa' )
disp(' ' )
disp(' ' )
option2=input('If you want to use the default values, press 1. If you want
to select own values, press 2:');
clc
end
disp(' ' )
disp(' ' )

if option2==1
    L=51.76242099;
    d=0.1460142;
```

```
    p_cable=7850;
    E=30307836229;
elseif option2 == 2
    L=input('Equivalent length in meters: (default value 50.022m)');
    d=input('Equivalent diameter of the cable: (default value 0.1485m)');
    p_cable=input('Density of the cable: (default value 7850 kg/m^3 )');
    E=input('Equivalent E-modulus of the cable: (default value 30.3*10^9 Pa
)');
end

fid2 = fopen('wireparam.txt','w');
fprintf(fid2, '%f %f %f %f\n' , L, d, p_cable, E);
fclose(fid2);
clc

while option_wavespec~=1 && option_wavespec~=2 && option_wavespec~=3
disp('*******************************************************************' )
disp(' ' )
disp('Choose what wavespectrum you want to use. You can choose between
JONSWAP and Pierson-Moskowitz' )
disp(' ' )
disp('*******************************************************************' )
disp(' ' )
disp(' ' )
option_wavespec=input('If you want to use JONSWAP, press 1. If you want to
use Pierson-Moskowitz, press 2. If you want to use regular waves, press
3:');
clc
end

fid6 = fopen('optionwavespec.txt','w');
fprintf(fid6, '%f\n' , option_wavespec);
fclose(fid6);

if option_wavespec~=3

while option3~=1 && option3~=2
disp('*******************************************************************' )
disp(' ' )
disp('                           Environment' )
disp(' ' )
disp('*******************************************************************' )
disp(' ' )
disp(' ' )
disp('The default values for peak period, signifcant wave height, gamma and
the duration of')
disp('the sea state is: Tp=7.5104s , Hs=2.9m , gamma=4.5495 and time=3
hours' )
disp(' ' )
disp(' ' )
option3=input('If you want to use the default values, press 1. If you want
to select own values, press 2:');
clc
end
disp(' ' )
disp(' ' )
```

```
if option3==1
    Tp=7.5104;
    Hs=2.9;
    gamma=4.5495;
    time_w=3;
elseif option3 == 2
    Tp=input('Peak period [s] (default value 7,5104s) :');
    Hs=input('Significant wave height [m] (default value 2,9m):');
    gamma=input('Value of Gamma [-] (default value 4,5495)        :');
    time_w=input('Duration of sea state [h] (default value 3h)   :');
end

elseif option_wavespec==3

disp('******************************************************************' )
disp(' ' )
disp('                         Environment' )
disp(' ' )
disp('******************************************************************' )
disp(' ' )
disp(' ' )
disp('The default values for wave period and wave height(from trough to
crest) is')
disp('T=6s and H=2.9m' )
disp(' ' )
disp(' ' )
option4=input('If you want to use the default values, press 1. If you want
to select own values, press 2:');
clc
end
disp(' ' )
disp(' ' )

if option4==1
    T=6;
    H=2.9;
elseif option4 == 2
    T=input('period [s] (default value 6s) :');
    H=input('Wave height (from trough to crest) [m] (default value
2.9m):');
end

if option_wavespec~=3
fid3 = fopen('environmentparam.txt','w');
fprintf(fid3, '%f %f %f %f %f\n' , Tp, Hs, gamma, time_w, option_wavespec);
fclose(fid3);
elseif option_wavespec==3

fid3 = fopen('environmentparam.txt','w');
fprintf(fid3, '%f %f\n' , T, H);
fclose(fid3);
end
clc
```

```
while option5~=1 && option5~=2
disp('*******************************************************************' )
disp(' ' )
disp('Choose the towing velocity of the operation in meters per second' )
disp(' ' )
disp('*******************************************************************' )
disp(' ' )
disp(' The default towing velocity is 2 m/s')
disp(' ' )
option5=input('If you want to use the default value, press 1. If you want
to insert your own value, press 2:');
clc
end

if option5==1
    U_max=2;
elseif option5==2
    U_max=input('Towing velocity [m/s] (default value 2m/s) :');
end

fid4 = fopen('towingvel.txt','w');
fprintf(fid4, '%f\n' , U_max);
fclose(fid4);
clc
while option6~=1 && option6~=2
disp('*******************************************************************' )
disp(' ' )
disp('Choose the length , timestep and number of wave components of the
analysis' )
disp(' ' )
disp('*******************************************************************' )
disp(' ' )
disp(' The default length is 10800s , the default timestep is 0.2s and the
default number of wavecomponents is 1000')
disp(' ' )
option6=input('If you want to use the default value, press 1. If you want
to insert your own value, press 2:');
clc
end

if option6==1
    tmax=10800;
    timestep=0.2;
    numfreq=1000;
elseif option6==2
    tmax=input('Length of analysis (default value: 10800s) :');
    timestep=input('Timestep of the analysis (default value: 0.2s) :');
    numfreq=input('Number of wave components (default value: 1000) :');
end

fid5 = fopen('numfreq.txt','w');
fprintf(fid5, '%f %f %f\n' , numfreq, timestep, tmax);
fclose(fid5);
```

12.1.1.3 Offset.m

```
% ************************************************************************
% Routine:                  offset
% ------------------------------------------------------------------------
% Intension:                Calculate the horizontal offset and plot the
%                           dragforce and offset angle as a function of the
%                           towing velocity
%
% Parameter:                Description
% ------------------------------------------------------------------------
% Fd                        Drag force [N]
% m_t                       Mass of the template [kg]
% a33                       added mass in heave for template[kg]
% A_z                       Projected area of load (z-direction) [m^2]
% w0                        Submerged weight of the template [N]
% p_sw                      Density of sea water [kg/m^3]
% g                         Gravity [m/s^2]
% Te                        Effective tension [N]
% alfa_rad                  Angle of the horizontal offset [rad]
% alfa_deg                  Angle of the horizontal offset [deg]
% Cd_x                      Drag coefficient of the load (x-direction) [-]
% U_ms                      Towing velocity [m/s]
% U_max                     Maximum towing velocity velocity [m/s]
% U_knot                    Current velocity [knots]
% A_x                       Refferance area on the template (x-direction)
[m^2]
% ------------------------------------------------------------------------
% Programmer:               Torbjørn Aakerøy Olsen
% Last changed:             26.05.2011
% ************************************************************************
clear all
clc

disp('*******************************************************************' )
disp(' ' )
disp('   Calculation of the horizontal offset. The plots shows Dragforce in
newton and offset angle' )
disp('   in degrees, as a function of towing velocity in m/s ' )
disp(' ' )
disp('*******************************************************************' )

%import values selected by the user
fid = fopen('tempparam.txt');
temp = fscanf(fid, '%f %f %f %f %f %f %f\n');
fclose(fid);

a33=temp(1);
A_x=temp(2);
A_z=temp(3);
w0=temp(4);
m_t=temp(5);
Cd_x=temp(6);
```

```
fid2 = fopen('towingvel.txt');
tvel = fscanf(fid2, '%f\n');
fclose(fid2);

U_max=tvel;

% values-------------------------------------------------------------
p_sw = 1025;
g=9.80665;
%--------------------------------------------------------------------------

inc=0.1;
loop=U_max/inc;

for i = 1:loop

    U_ms(i)=i*inc;
    U_knot(i)=i*inc*1.9438612860586;
    Fd(i)=0.5*p_sw*U_ms(i)^2*Cd_x*A_x;
    Te(i)=sqrt(w0^2+Fd(i)^2);
    alfa_rad(i)=asin(Fd(i)/Te(i));
    alfa_deg(i)=asin(Fd(i)/Te(i))*180/pi;

end

set(figure,'units','normalized','outerposition',[0 0 1 1]);
plot(U_ms,Fd,'color','red','linewidth',2);
xlabel('Towing velocity [m/s]','fontsize',18);
ylabel('Drag force [N]','fontsize',18);
title('Drag force in newton as a function of towing velocity in
m/s','fontsize',18);
saveas(gcf,'dragforce','bmp');

set(figure,'units','normalized','outerposition',[0 0 1 1]);
plot(U_ms,alfa_deg,'linewidth',2);
xlabel('Towing velocity [m/s]','fontsize',18);
ylabel('Offset angle [deg]','fontsize',18);
title('Offset angle in degrees as a function of towing velocity in
m/s','fontsize',18);
saveas(gcf,'offset angle','bmp');

disp(' ');
disp(' ');
disp('###################################################################');
disp('##        Table with towing velocities [m/s] and corresponding offset
angles [deg]       ##');
disp('###################################################################');
disp(' ');
disp(' ');

fprintf('Towing velocity [m/s] and offset angle [deg] respectively: %f
%f\n', U_ms(1), alfa_deg(1));
for i=2:(length(U_ms))
fprintf('                                                          %f
%f\n', U_ms(i), alfa_deg(i));
end
```

12.1.1.4 Dof.m

```
% ************************************************************************
% Routine:                  dof
% ------------------------------------------------------------------------
% Intension:                Calculate and plot the dynamic displacement and
%                           force as a function of the frequency
%
% Parameter:                Description
% ------------------------------------------------------------------------
% etta                      mean amplitude when x=L [m]
% w                         Vector with frequency values [rad/s]
% k                         Vector with wave number values [1/m]
% L                         Length of the cable [m]
% x                         Point at cable where the calculation is done [m]
% m_cable                   Mass of the cable [kg/m]
% m_template                Mass of the template [kg]
% mc_cable                  Complex mass of cable [kg]
% mc_template               Complex mass of template [kg]
% A                         Crossection area of the cable [m^2]
% E                         E-modulus of cable [Pa]
% a33                       added mass in heave for template[kg]
% d                         Diameter of the cable [m]
% p_cable                   Density of the cable [kg/m^3]
% p_sw                      Density of sea water [kg/m^3]
% p_template                Density of the template [kg/m^3]
% g                         Gravity [m/s^2]
% m_temp                    Mass of the template [kg]
% Cd_cable                  Drag coefficient for cable in z-direction due to
%                           skin friction [-]
% Cd_z                      Drag coefficient for template in z-direction [-]
% A_z                       Projected area of load, z-direction [m^2]
% cv_cable                  Estimated drag coefficient for cable [-]
% cv_template               Estimated drag coefficient for template [-]
% ------------------------------------------------------------------------
% Programmer:               Torbjørn Aakerøy Olsen
% Last changed:             26.05.2011
% ************************************************************************
clear all
clc

disp('*****************************************************************' )
disp(' ' )
disp('   Calculation of dynamic displacement and dynamic force as a
function of frequency.' )
disp('   The plots shows results at the endpoint of the cable (x=L). ' )
disp(' ' )
disp('*****************************************************************' )

%------------ Input values --------------------------------------------
%import values from textfile
fid = fopen('tempparam.txt');
temp = fscanf(fid, '%f %f %f %f %f %f %f\n');
fclose(fid);

a33=temp(1);
```

```
A_z=temp(3);
Cd_z=temp(7);
m_temp = temp(5);

fid1 = fopen('wireparam.txt');
wire = fscanf(fid1, '%f %f %f %f\n');
fclose(fid1);

L=wire(1);
d=wire(2);
p_cable=wire(3);
E = wire(4);

fid2 = fopen('numfreq.txt');
analysis = fscanf(fid2, '%f %f %f\n');
fclose(fid2);

numfreq=analysis(1);

%------    Parameters    --------------------------------------------------

g = 9.80665;
p_sw=1025;
t=0;
Cd_cable=0.05;
etta=1;

%-------------------calculated values--------------------------------------
x=L;
A=pi*(d^2)*(1/4);
m_cable=p_cable*A;
cv_cable=0.5*p_sw*Cd_cable*2*pi*(d/2);
cv_template=0.5*p_sw*Cd_z*A_z;
mc_cable=m_cable+(8*1i*cv_cable*etta/(3*pi));
mc_template=(m_temp+a33)+(8*1i*cv_template*etta/(3*pi));

wmax = 10 ;
wmin = 0 ;
inc = (wmax-wmin)/numfreq;
w=zeros(1,numfreq);

for j=1 : numfreq
   w(j) = inc * (j-0.5);
   k(j)=w(j)*sqrt(mc_cable/(E*A));
   disp(j)= cos(w(j)*t) * ((cos(k(j)*x)+ ((
1+(mc_cable*tan(k(j)*L)/(mc_template*k(j)))))/
((mc_cable/(mc_template*k(j)))-tan(k(j)*L)))*sin(k(j)*x)));
   force(j)=E*A*k(j)*cos(w(j)*t) * ((-sin(k(j)*x)+ ((
1+(mc_cable*tan(k(j)*L)/(mc_template*k(j)))))/
((mc_cable/(mc_template*k(j)))-tan(k(j)*L)))*cos(k(j)*x)));
   result_disp(j)=sqrt(real(disp(j))^2+imag(disp(j))^2);
   result_force(j)=sqrt(real(force(j))^2+imag(force(j))^2);

end
```

```
%------------ plot---------------------------------------------------------

set(figure,'units','normalized','outerposition',[0 0 1 1]);
plot(w,result_disp,'linewidth',2);
xlabel('\omega [rad/s]','fontsize',18);
ylabel('\eta/\eta_{a} [m/m]','fontsize',18);
title('dynamic displacement with damping','fontsize',18);
axis('tight');
saveas(gcf,'dynamic disp','bmp');

set(figure,'units','normalized','outerposition',[0 0 1 1]);
plot(w,result_force,'color','green','linewidth',2);
xlabel('\omega [rad/s]','fontsize',18);
ylabel('F_{d}/\eta_{a} [N/m]','fontsize',18);
title('dynamic force with damping','fontsize',18);
axis('tight');
saveas(gcf,'dynamic force','bmp');
```

12.1.1.5Dof3d.m

```
% ***********************************************************************
% Routine:                dof3d
% -----------------------------------------------------------------------
% Intension:              Create 3D plots of the motion response and the
%                         absolute part of the force as a function of the
%                         frequency and distanse from the surface.
%
% Parameter:              Description
% -----------------------------------------------------------------------
% etta                    mean amplitude when x=L [m]
% w                       Vector with frequency values [rad/s]
% k                       Vector with wave number values [1/m]
% L                       Length of the cable [m]
% x                       Point at cable where the calculation is done [m]
% m_cable                 Mass of the cable [kg/m]
% m_template              Mass of the template [kg]
% mc_cable                Complex mass of cable [kg]
% mc_template             Complex mass of template [kg]
% A                       Crossection area of the cable [m^2]
% E                       E-modulus of cable [Pa]
% a33                     added mass in heave for template[kg]
% d                       Diameter of the cable [m]
% p_cable                 Density of the cable [kg/m^3]
% p_sw                    Density of sea water [kg/m^3]
% g                       Gravity [m/s^2]
% m_temp                  Submerged weight of template [kg]
% Cd_cable                Drag coefficient for cable z-dir (skin friction)
[-]
% Cd_z                    Drag coefficient for template z-dir [-]
% A_z                     Projected area of load z-direction [m^2]
% cv_cable                Estimated drag coefficient for cable z-dir [-]
% cv_template             Estimated drag coefficient for template z-dir [-]
% disp                    Motion response of the cable [m]
% force                   Force in the cable [N]
% dist                    Vector of points along the cable. Origin at the
%                         surface and positive direction downwards [m]
% -----------------------------------------------------------------------
% Programmer:             Torbjørn Aakerøy Olsen
% Last changed:           26.05.2011
% ***********************************************************************
clear all
clc

disp('*******************************************************************' )
disp(' ' )
disp('   3D plots of the absolute part of the motion response, the real
part of the motion response and the ' )
disp('   absolute part of the force as a function of the frequency and
distanse from the surface.' )
disp(' ' )
disp('*******************************************************************' )

%------------ Input values ------------------------------------------------
%import values from textfile
fid = fopen('tempparam.txt');
temp = fscanf(fid, '%f %f %f %f %f %f %f\n');
```

```
fclose(fid);

a33=temp(1);
A_z=temp(3);
Cd_z=temp(7);
m_temp = temp(5);

fid1 = fopen('wireparam.txt');
wire = fscanf(fid1, '%f %f %f %f\n');
fclose(fid1);

L=wire(1);
d=wire(2);
p_cable=wire(3);
E = wire(4);

fid2 = fopen('numfreq.txt');
analysis = fscanf(fid2, '%f %f %f\n');
fclose(fid2);

numfreq=analysis(1);
%--------------------------------------------------------------------------

g = 9.80665;
p_sw=1025;
t=0;
Cd_cable=0.05;
etta=1;

%-------------------calculated values-------------------------------------
A=pi*(d^2)*(1/4);
m_cable=p_cable*A;
cv_cable=0.5*p_sw*Cd_cable*2*pi*(d/2);
cv_template=0.5*p_sw*Cd_z*A_z;
mc_cable=m_cable+(8*1i*cv_cable*etta/(3*pi));
mc_template=(m_temp+a33)+(8*1i*cv_template*etta/(3*pi));

wmax = 10 ;
wmin = 0 ;
inc = (wmax-wmin)/numfreq;
w=zeros(1,numfreq);

for x=1:L
        for j = 1:numfreq
             dist(x)=x;
             w(j) = inc * (j-0.5);
             k(j)=w(j)*sqrt(mc_cable/(E*A));
             disp(j,x)= cos(w(j)*t) * ((cos(k(j)*x)+ ((
1+(mc_cable*tan(k(j)*L)/(mc_template*k(j)))))/
((mc_cable/(mc_template*k(j)))-tan(k(j)*L)))*sin(k(j)*x)));
             force(j,x)=E*A*k(j)*cos(w(j)*t) * ((-sin(k(j)*x)+ ((
1+(mc_cable*tan(k(j)*L)/(mc_template*k(j)))))/
((mc_cable/(mc_template*k(j)))-tan(k(j)*L)))*cos(k(j)*x)));
             result_disp(j,x)=sqrt(real(disp(j,x))^2+imag(disp(j,x))^2);
             result_force(j,x)=sqrt(real(force(j,x))^2+imag(force(j,x))^2);
             real_force(j,x)=real(force(j,x));
```

```
            real_disp(j,x)=real(disp(j,x));
            imag_force(j,x)=imag(force(j,x));
            imag_disp(j,x)=imag(disp(j,x));

        end
end
%--------- 3D plot----------------------------------------------------------

set(figure,'units','normalized','outerposition',[0 0 1 1]);
surf(dist,w,result_disp)
xlabel('wire distance [m]','fontsize',18)
ylabel('\omega [rad/s]','fontsize',18)
zlabel('Absolute value motion response \eta/\eta_{a} [m/m]','fontsize',18)
title('Absolute part of motion response','fontsize',18)
colorbar;
shading flat;
saveas(gcf,'3d abs motion','bmp');

set(figure,'units','normalized','outerposition',[0 0 1 1]);
surf(dist,w,real_disp)
xlabel('wire distance [m]','fontsize',18)
ylabel('\omega [rad/s]','fontsize',18)
zlabel('Real part of motion response \eta/\eta_{a} [m/m]','fontsize',18)
title('Real part of motion response','fontsize',18)
colorbar;
shading flat;
saveas(gcf,'3d real part motion','bmp');


set(figure,'units','normalized','outerposition',[0 0 1 1]);
surf(dist,w,result_force)
xlabel('wire distance [m]','fontsize',18)
ylabel('\omega [rad/s]','fontsize',18)
zlabel('F_{d}/\eta{a} [N/m]','fontsize',18)
title('Absolute value of the force','fontsize',18)
colorbar;
shading flat;
saveas(gcf,'3d abs value force','bmp');
```

12.1.1.6 Tempres.m

```
% ************************************************************************
% Routine:                  tempres
% ------------------------------------------------------------------------
% Intension:                Create the wavespectrum (either JONSWAP or
%                           Pierson-moskowitz. Plot the
%                           transferfunction of the vessel from equation
%                           obtained in transfer.m. Create and plot
%                           the surface elevation of the waves. Create and
%                           plot the heave motion of the vessel. Create and
%                           plot the heave motion of the template.
%
% Parameter:                Description
% ------------------------------------------------------------------------
% Hs                        Significant wave height [m]
% Tp                        Peak period [s]
% Gamma                     Constant in the jonswap spectrum
% wp                        Top frequency [rad/s]
% time_w                    Duration of sea state [h]
% numfreq                   Number of frequency intervals
% wmax                      Maximum frequency value [rad/s]
% wmin                      Minimum frequency value [rad/s]
% w                         Vector of frequency values [rad/s]
% S                         Vector of wavespectrum values [m^2/(rad/s)]
% k                         Vector of wavenumber values [1/m]
% sigma                     Constant in the calculation of JONSWAP-spectrum
% inc                       Increment of each frequency value
% m0                        The 0. moment
% m1                        The 1. moment
% m2                        The 2. moment
% m4                        The 4. moment
% Tm0z                      Zero crossing period [s]
% Tm0e                      Mean period [s]
% Tm24                      Mean period between wave crests [s]
% Tz                        Mean zero crossing period [s]
% Hm0                       Calculated significant wave height [m]
% N                         Duration of the sea state (default 5 hours)
% Hmax                      Highest wave in a given period (default 5 hours)
% H1n                       Highest wave with a possibility of 1/N [m]
% p_sw                      Density of sea water [kg/m^3]
% trans_b                   Vector with values of the transferfunction for
%                           the vessel [m/m]
% tmax                      Maximum time [s]
% step                      Timestep of the analysis [s]
% zeta_w                    Vector with values of the surface elevation [m]
% vel_w                     Vector with values of the surface velocity
%                           in heave [m/s]
% acc_w                     Vector with values of the surface acceleration
%                           in heave [m/s^2]
% zeta_b                    Vector with values of the heave motion of the
%                           vessel [m]
% vel_b                     Vector with values of the heave velocity of the
%                           vessel [m/s]
% acc_b                     Vector with values of the heave acceleration of
%                           the vessel [m/s^2]
% eps                       Random phaseangel of the waves [rad]
% pos                       X-position from the COG of the vessel
```

```
% a33_t                      Added mass of the template [kg]
% m_t                        Mass of the template [kg]
% E_c                        E-modulus of the cable [Pa]
% d_c                        Diameter of the cable [m]
% A_c                        Cross-section area of the cable [m^2]
% L_c                        Length of the cable [m]
% k_c                        Stiffness of the cable[N/m]
% b_t                        Damping of the cable [kg/s]     VISCOUS DAMPING
OF THE TEMPLATE?
% eigenfreq                  Eigenfrequency [rad/s]
% d                          Vector with approximation of the template
%                            displacement in heave [m]
% v                          Vector with approximation of the template
%                            velocity in heave [m/s]
% a                          Vector with approximation of the template
%                            acceleration in heave [m/s^2]
% d_p                        Vector with predicted values of the template
%                            displacement in heave [m]
% v_p                        Vector with predicted values of the template
%                            velocity in heave [m/s]
% gamma_newmark                      Parameter to determine the stability and
accuracy
%                            characteristics of the Newmark_beta algorithm
% beta                       Parameter to determine the stability and accuracy
%                            characteristics of the Newmark_beta algorithm
% F3                         Vector of applied forces to the template [N]
% zeta_t                     Vector with values of the heave motion of the
%                            template [m]
% motiondiff                 Vector with the motion difference of the vessel
%                            and the template [m]
% -------------------------------------------------------------------------
% Programmer:                Torbjørn Aakerøy Olsen
% Last changed:              26.05.2011
% *************************************************************************
clear all
clc

disp('******************************************************************')
disp('
disp('     Calculation and plotting of:
disp('     The chosen wavespectrum
disp('     The transferfunction in heave for the vessel
disp('     The time history of the wave
disp('     The time history of the heave motion of the vessel
disp('     The time history of the heave motion of the template
disp('     The template and vessel displacement in heave
disp('     The absolute value of the motion difference between the vessel
and template in heave                ')
disp('
disp('******************************************************************')

%--------------    INPUT    ---------------------------------------

%import values selected by the user
fid = fopen('environmentparam.txt');
environ = fscanf(fid, '%f %f %f %f %f\n');
fclose(fid);
```

```
Tp=environ(1);
Hs=environ(2);
gamma=environ(3);
time_w=environ(4);
option_wavespec=environ(5);

fid10 = fopen('numfreq.txt');
analysis = fscanf(fid10, '%f %f %f\n');
fclose(fid10);

numfreq=analysis(1);
step=analysis(2);
tmax=analysis(3);

%input values----------------------------------------------------------------
wmax = 4 ;
wmin = 0;

g=9.80665;

%----------- CALCULATION OF THE WAVESPECTRUM THAT IS CHOSEN  --------------

wp=2*pi/Tp;
inc = (wmax-wmin)/numfreq;

w=zeros(1,numfreq);
S=zeros(1,numfreq);

m0_komp=zeros(1,numfreq);
m1_komp=zeros(1,numfreq);
m2_komp=zeros(1,numfreq);
m4_komp=zeros(1,numfreq);

% save the vectors with: frequency, wavenumber and spectrum values
% into a textfile

fid1 = fopen('omega.txt','w');
fid2 = fopen('wavenumber.txt','w');
fid3 = fopen('spectrum.txt','w');

if option_wavespec==1

for i = 1 : numfreq

    w(i) = inc * (i-0.5);


       if wp<w(i)
           sigma=0.09;
       else
           sigma = 0.07;
       end
```

```
        x=(5/(32*pi))*(Hs^2)*Tp;
        y=(wp/w(i))^5;
        z=(-1.0*5/4)*((wp/w(i))^4);
        v=(-1.0*((w(i)/wp)-1.0)^2)/(2*sigma^2);
        m=exp(v);

        S(i)=x*y*exp(z)*(1-(0.287*log(gamma)))*(gamma^m);
        Sf(i)=S(i)*2*pi;
        fhz(i)=w(i)/(2*pi);
        k(i)=w(i)^2/g;

        m0_komp(i)=inc*(w(i)^0)*S(i);
        m1_komp(i)=inc*(w(i)^1)*S(i);
        m2_komp(i)=inc*(w(i)^2)*S(i);
        m4_komp(i)=inc*(w(i)^4)*S(i);

           fprintf(fid1, '%6.11f\n' , w(i));
           fprintf(fid2, '%6.11f\n' , k(i));
           fprintf(fid3, '%6.11f\n' , S(i));

    end




    elseif option_wavespec==2
        for i = 1 : numfreq

        w(i) = inc * (i-0.5);


        S(i)= (Hs^2) * Tp * (0.11/(2*pi)) * (w(i)*Tp/(2*pi))^-5 * exp(-
    0.44*(w(i)*Tp/(2*pi))^-4);


        k(i)=w(i)^2/g;

        m0_komp(i)=inc*(w(i)^0)*S(i);
        m1_komp(i)=inc*(w(i)^1)*S(i);
        m2_komp(i)=inc*(w(i)^2)*S(i);
        m4_komp(i)=inc*(w(i)^4)*S(i);

           fprintf(fid1, '%6.11f\n' , w(i));
           fprintf(fid2, '%6.11f\n' , k(i));
           fprintf(fid3, '%6.11f\n' , S(i));

        end
    end

    m0=sum(m0_komp);
    m1=sum(m1_komp);
    m2=sum(m2_komp);
    m4=sum(m4_komp);
```

```
fclose(fid1);
fclose(fid2);
fclose(fid3);

% %--------- CALCULATION OF RESULTS  -------------------------------------

        Tm0z=2*pi*sqrt(m0/m2);
        Tm0e=2*pi*(m0/m1);
        Tm24=2*pi*sqrt(m2/m4);
        Tz=(pi*Hs)/(2*sqrt(m2));
        Hm0=4*sqrt(m0);
        N=time_w*60*60/Tz;
        Hmax=Hs*sqrt(log(N)/2);
        H1n=Hs*sqrt(log(N)/2);

disp('###################################################################');
disp('## All the results and plots are based on the choices made by the 
user. ##');
disp('###################################################################');
disp(' ');
disp(' ');
fprintf('Peak period in seconds:...........................:%f\n', Tp);
fprintf('Significant wave height in meters:................:%f\n', Hs);
fprintf('Value of Gamma:...................................:%f\n', gamma);
fprintf('Duration of sea state in hours:...................:%f\n', time_w);
if option_wavespec==1
    disp('Chosen wavespectrum:............................: JONSWAP')
elseif option_wavespec==2
    disp('Chosen wavespectrum:......................: Pierson-Moskowitz')
end
disp(' ');
disp(' ');
disp('###################################################################');
disp('##                     Some results                            ##');
disp('###################################################################');
disp(' ');
fprintf('Zero crossing period, Tm0z [s].....................:%f\n', Tm0z);
fprintf('Mean period, Tm0e [s]..............................:%f\n', Tm0e);
fprintf('Mean period between wave crests, Tm24 [s]..........:%f\n', Tm24);
fprintf('Mean zero crossing period, Tz [s]..................:%f\n', Tz);
fprintf('Calculated significant wave height, Hm0 [m]........:%f\n', Hm0);
fprintf('Highest wave in a given period (default 5 hours), Hmax [m]:%f\n', 
Hmax);
fprintf('Highest wave with a possibility of 1/N, H1n [m]....:%f\n', H1n);
disp(' ');
disp('#################################################################');
disp(' ');
disp(' ');

%Calculation and plot of the transferfunction in heave for the vessel------
-

%the phaseangel for the ship is ignored since only the relative magnitude
%of heavemotion is of interest.

% Equation for the transfer function found from transfer.m----------------
```

```
trans_b=zeros(numfreq,1);

        p1 =-0.194;
        p2 =1.76;
        p3 =-5.679;
        p4 =8.408;
        p5 =-5.793;
        p6 =1.505;
        q1 =-1.07;
        q2 =-3.239;
        q3 =7.347;
        q4 =-5.493;
        q5 =1.471;

        fid4 = fopen('transb.txt','w');

        period=zeros(numfreq,1);

        for i=1:numfreq
        period(i)=2*pi/w(i);
        trans_b(i)=abs((p1*w(i)^5 + p2*w(i)^4 + p3*w(i)^3 + p4*w(i)^2 +
p5*w(i) + p6) / (w(i)^5 + q1*w(i)^4 + q2*w(i)^3 + q3*w(i)^2 + q4*w(i) +
q5));
        fprintf(fid4, '%6.11f\n' , trans_b(i));
        end

        fclose(fid4);

%---- Calculation and plotting of surface elevation-----------------------

zeta_w=zeros((tmax/step),1);
vel_w=zeros((tmax/step),1);
acc_w=zeros((tmax/step),1);
zeta_b=zeros((tmax/step),1);
pos=0;

% save the x-position into a textfile-------------------------------------
fid5 = fopen('pos.txt','w');
fprintf(fid5, '%f\n' , pos);
fclose(fid5);
%-------------------------------------------------------------------------

% save the vectors: time and phaseangel of the wave into a textfile -------
fid6 = fopen('time.txt','w');
fid7 = fopen('epswave.txt','w');

for i = 1:(tmax/step)
    time(i)=step*i;

    fprintf(fid6, '%f\n' , time(i));

end

for i =1:numfreq
```

```
  eps(i)=rand(1)*2*pi;
  fprintf(fid7, '%f\n' , eps(i));
end

fclose(fid6);
fclose(fid7);

%---------------------------------------------------------------------------

fid22 = fopen('wave_matlab.txt','w');

counter=0;

timestart=time(1);

for t=timestart:step:tmax

    counter=counter+1;
        j_zeta=0;
        j_vel=0;
        j_acc=0;
        y_zeta=0;
        for i=1:numfreq

        j_zeta=j_zeta+(sqrt(2*S(i)*inc)*sin((w(i)*t)-(k(i)*pos)+eps(i)));
        j_vel=j_vel+(sqrt(2*S(i)*inc)*(w(i)*cos((w(i)*t)-
(k(i)*pos)+eps(i))));
        j_acc=j_acc+(sqrt(2*S(i)*inc)*(-1*w(i)^2*sin((w(i)*t)-
(k(i)*pos)+eps(i))));
        y_zeta=y_zeta+trans_b(i)*(sqrt(2*S(i)*inc)*cos((w(i)*t)+eps(i)));

        end

        zeta_w(counter)=j_zeta;
        vel_w(counter)=j_vel;
        acc_w(counter)=j_acc;
        zeta_b(counter)=y_zeta;

        fprintf(fid22, '%f\n' , zeta_w(counter));

end

fclose(fid22);

vel_b = gradient(zeta_b)*(1/step);
acc_b = gradient(vel_b)*(1/step);

%calculation of template response------------------------------------------

%import values from textfile -----------------------------------------------

fid8 = fopen('tempparam.txt');
temp = fscanf(fid8, '%f %f %f %f %f %f %f\n');
fclose(fid8);
```

```
a33_t=temp(1);
A_z=temp(3);
m_t_dry = temp(5);
Cd_z=temp(7);
p_sw=1025;

fid9 = fopen('wireparam.txt');
wire = fscanf(fid9, '%f %f %f %f\n');
fclose(fid9);

L_c=wire(1);
d_c=wire(2);
E_c = wire(4);

%-----------------------------------------------------------------------

m_t_w_added=((m_t_dry)+a33_t);
A_c=(pi/4)*d_c^2;
k_c=E_c*A_c/L_c;
b_t=0;
eigenfreq=sqrt(k_c/m_t_w_added);

% save the damping into a textfile-----------------------------------------
fid10 = fopen('damp.txt','w');
fprintf(fid10, '%f\n' ,b_t);
fclose(fid10);
%-----------------------------------------------------------------------

d=zeros((tmax/step),1);
v=zeros((tmax/step),1);
a=zeros((tmax/step),1);
d_p=zeros((tmax/step),1);
v_p=zeros((tmax/step),1);

gamma_newmark=1;
beta=0.25;

%establishment of the Load vector (F)--------------------------------------

teller = 0;
for t=timestart:step:tmax
    teller=teller+1;
    F3(teller)=-m_t_w_added*acc_b(teller)-
b_t*vel_b(teller)+0.5*p_sw*vel_b(teller)^2*Cd_z*A_z;
end

%iteration according to the newmark beta method ---------------------------

clear teller;
teller=0;

for i=1:((tmax/step)-1)

    teller=teller+1;
```

```
    d_p(teller+1)=d(teller)+step*v(teller)+ (step^2/2)*(1-
2*beta)*a(teller);
    v_p(teller+1)=v(teller)+(1-gamma_newmark)*step*a(teller);
    a(teller+1)=(F3(teller+1)-b_t*v_p(teller+1)-
k_c*d_p(teller+1))/(m_t_w_added+gamma_newmark*step*b_t+beta*step^2*k_c);
    d(teller+1)=(d_p(teller+1)+beta*step^2*a(teller+1));
    v(teller+1)=v_p(teller+1)+gamma_newmark*step*a(teller+1);

end

for i = 1:(tmax/step)
    zeta_t(i)=zeta_b(i)-d(i);
    motiondiff(i)=zeta_b(i)-zeta_t(i);
end

fid17=fopen('vessel_heave_matlab.txt','w');
fid18=fopen('template_heave_matlab.txt','w');
for i=1:(tmax/step)
fprintf(fid17,'%6.11f\r\n', zeta_b(i));
fprintf(fid18,'%6.11f\r\n', zeta_t(i));
end
fclose(fid17);
fclose(fid18);

%Plotting of the results --------------------------------------------------

if option_wavespec==1

set(figure,'units','normalized','outerposition',[0 0 1 1]);
plot (w,S,'linewidth',2);
ylabel('S(\omega) [m^2/(rad/s)]','fontsize',18,'linewidth',2);
xlabel('\omega [rad/s]','fontsize',18);
title('Plot of the JONSWAP spectrum with given values of Tp, Hs and
gamma','fontsize',18);
saveas(gcf,'Jonswap spectrum(w)','bmp');

set(figure,'units','normalized','outerposition',[0 0 1 1]);
plot (fhz,Sf,'linewidth',2);
ylabel('S(f) [m^2/(Hz)]','fontsize',18,'linewidth',2);
xlabel('\omega [Hz]','fontsize',18);
title('Plot of the JONSWAP spectrum with given values of Tp, Hs and
gamma','fontsize',18);
saveas(gcf,'Jonswap spectrum(f)','bmp');

elseif option_wavespec==2

set(figure,'units','normalized','outerposition',[0 0 1 1]);
plot (w,S,'linewidth',2);
ylabel('S(\omega) [m^2/(rad/s)]','fontsize',18);
xlabel('\omega [rad/s]','fontsize',18);
title('Plot of the Pierson-Moskowitz spectrum with given values of Tp, Hs
and gamma','fontsize',18);
saveas(gcf,'PM spectrum(w)','bmp');
end
```

```
set(figure,'units','normalized','outerposition',[0 0 1 1]);
plot(w,trans_b,'linewidth',2)
xlabel('\omega [rad/s]','fontsize',18);
ylabel('H(\omega) [m/m]','fontsize',18);
title('Transferfunction in heave for the vessel','fontsize',18);
saveas(gcf,'transfer vessel','bmp');

set(figure,'units','normalized','outerposition',[0 0 1 1]);
plot(time,zeta_w,'linewidth',2)
xlabel('Time [s]','fontsize',18);
ylabel('Amplitude [m]','fontsize',18);
title('Time history of the wave','fontsize',18);
saveas(gcf,'time history wave','bmp');

set(figure,'units','normalized','outerposition',[0 0 1 1]);
plot(time,zeta_b,'color','green','linewidth',2)
xlabel('Time [s]','fontsize',18);
ylabel('Amplitude [m]','fontsize',18);
title('Time history of the heave motion of the vessel','fontsize',18);
saveas(gcf,'time history vessel','bmp');

set(figure,'units','normalized','outerposition',[0 0 1 1]);
plot(time,zeta_t,'linewidth',2)
xlabel('Time [s]','fontsize',18);
ylabel('Amplitude [m]','fontsize',18);
title('Time history of the heave motion of the template','fontsize',18);
saveas(gcf,'time history template','bmp');

set(figure,'units','normalized','outerposition',[0 0 1 1]);
plot(time,zeta_b,time,zeta_t,'linewidth',2)
xlabel('Time [s]','fontsize',18);
ylabel('Amplitude [m]','fontsize',18);
title('The template and vessel displacement in heave','fontsize',18);
legend('vessel motion','template motion')
saveas(gcf,'time history template and vessel','bmp');

set(figure,'units','normalized','outerposition',[0 0 1 1]);
plot(time,abs(motiondiff),'color','red','linewidth',2)
xlabel('Time [s]','fontsize',18);
ylabel('Amplitude [m]','fontsize',18);
title('The absolute value of the motion difference between vessel and
template in heave','fontsize',18);
saveas(gcf,'abs motiondiff','bmp');
```

12.1.1.7 Tempresregular.m

```
% ************************************************************************
% Routine:                tempresregular
% ------------------------------------------------------------------------
% Intension:              Create and plot the surface elevation of the
%                         regular waves. Create and
%                         plot the heave motion of the vessel. Create and
%                         plot the heave motion of the template.
%
% Parameter:              Description
% ------------------------------------------------------------------------
% T                       Period of the reqular wave [s]
% H                       Wave height from trough to crest [m]
% step                    timestep of the analysis [s]
% time_w                  Duration of sea state [h]
% numfreq                 Number of frequency intervals
% wmax                    Maximum frequency value [rad/s]
% wmin                    Minimum frequency value [rad/s]
% g                       gravitation constant [m^2/s]
% w                       Vector of frequency values [rad/s]
% S                       Vector of wavespectrum values [m^2/(rad/s)]
% k                       Vector of wavenumber values [1/m]
% inc                     Increment of each frequency value
% p_sw                    Density of sea water [kg/m^3]
% trans_b                 Vector with values of the transferfunction for
%                         the vessel [m/m]
% tmax                    Maximum time [s]
% step                    Timestep
% zeta_w                  Vector with values of the surface elevation [m]
% vel_w                   Vector with values of the surface velocity
%                         in heave [m/s]
% acc_w                   Vector with values of the surface acceleration
%                         in heave [m/s^2]
% zeta_b                  Vector with values of the heave motion of the
%                         vessel [m]
% vel_b                   Vector with values of the heave velocity of the
%                         vessel [m/s]
% acc_b                   Vector with values of the heave acceleration of
%                         the vessel [m/s^2]
% eps                     Random phaseangel of the waves [rad]
% pos                     X-position from the COG of the vessel
% a33_t                   Added mass of the template [kg]
% m_t                     Mass of the template [kg]
% E_c                     E-modulus of the cable [Pa]
% d_c                     Diameter of the cable [m]
% A_c                     Cross-section area of the cable [m^2]
% L_c                     Length of the cable [m]
% k_c                     Stiffness of the cable[N/m]
% b_t                     Damping of the cable [kg/s]     VISCOUS DAMPING
OF THE TEMPLATE?
% eigenfreq               Eigenfrequency [rad/s]
% d                       Vector with approximation of the template
%                         displacement in heave [m]
% v                       Vector with approximation of the template
%                         velocity in heave [m/s]
% a                       Vector with approximation of the template
%                         acceleration in heave [m/s^2]
```

```
% d_p                       Vector with predicted values of the template
%                           displacement in heave [m]
% v_p                       Vector with predicted values of the template
%                           velocity in heave [m/s]
% gamma_newmark             Parameter to determine the stability and accuracy
%                           characteristics of the Newmark_beta algorithm
% beta                      Parameter to determine the stability and accuracy
%                           characteristics of the Newmark_beta algorithm
% F3                        Vector of applied forces to the template [N]
% zeta_t                    Vector with values of the heave motion of the
%                           template [m]
% motiondiff                Vector with the motion difference of the vessel
%                           and the template [m]
% -----------------------------------------------------------------------------
% Programmer:               Torbjørn Aakerøy Olsen
% Last changed:             26.05.2011
% *****************************************************************************
clear all
clc

disp('******************************************************************')
disp('
disp('     Calculation and plotting of:
disp('     The chosen wavespectrum
disp('     The transferfunction in heave for the vessel
disp('     The time history of the wave
disp('     The time history of the heave motion of the vessel
disp('     The time history of the heave motion of the template
disp('     The template and vessel displacement in heave
disp('     The absolute value of the motion difference between the vessel
and template in heave              ')
disp('
disp('******************************************************************')

%--------------    INPUT    ---------------------------------------

%import values selected by the user
fid = fopen('environmentparam.txt');
environ = fscanf(fid, '%f %f\n');
fclose(fid);

T=environ(1);
H=environ(2);

fid10 = fopen('numfreq.txt');
analysis = fscanf(fid10, '%f %f %f\n');
fclose(fid10);

numfreq=analysis(1);
step=analysis(2);
tmax=analysis(3);

%input values-----------------------------------------------------------

wmax = 4 ;
wmin = 0 ;
```

```
g=9.80665;

%----------- CALCULATION OF THE WAVESPECTRUM THAT IS CHOSEN  --------------

inc = (wmax-wmin)/numfreq;

w=zeros(1,numfreq);

% save the vectors with: frequency, wavenumber and spectrum values
% into a textfile

fid1 = fopen('omega.txt','w');
fid2 = fopen('wavenumber.txt','w');

for i = 1 : numfreq

    w(i) = 2*pi/T;
    k(i)=w(i)^2/g;

       fprintf(fid1, '%6.11f\n' , w(i));
       fprintf(fid2, '%6.11f\n' , k(i));

end

fclose(fid1);
fclose(fid2);



%Calculation and plot of the transferfunction in heave for the vessel------
-

%the phaseangel for the ship is ignored since only the relative magnitude
%of heavemotion is of interest.

% Equation for the transfer function found from transfer.m-----------------

trans_b=zeros(numfreq,1);

        p1 =-0.194;
        p2 =1.76;
        p3 =-5.679;
        p4 =8.408;
        p5 =-5.793;
        p6 =1.505;
        q1 =-1.07;
        q2 =-3.239;
        q3 =7.347;
        q4 =-5.493;
        q5 =1.471;

        fid4 = fopen('transb.txt','w');

        period=zeros(numfreq,1);
```

```
        for i=1:numfreq
        omega(i)=inc * (i-0.5);
        period(i)=2*pi/w(i);
        trans_b(i)=abs((p1*omega(i)^5 + p2*omega(i)^4 + p3*omega(i)^3 +
p4*omega(i)^2 + p5*omega(i) + p6) / (omega(i)^5 + q1*omega(i)^4 +
q2*omega(i)^3 + q3*omega(i)^2 + q4*omega(i) + q5));
        fprintf(fid4, '%6.11f\n' , trans_b(i));
        end

        transb_use=0;

        for i = 1:numfreq
            if abs(omega(i)-w(1))<=(inc/2)
                transb_use=transb_use+trans_b(i);
                position=i;
            end
        end

        fclose(fid4);


%---- Calculation and plotting of surface elevation-----------------------

zeta_w=zeros((tmax/step),1);
vel_w=zeros((tmax/step),1);
acc_w=zeros((tmax/step),1);
zeta_b=zeros((tmax/step),1);
pos=0;

% save the x-position into a textfile-------------------------------------
fid5 = fopen('pos.txt','w');
fprintf(fid5, '%f\n' , pos);
fclose(fid5);
%-------------------------------------------------------------------------

% save the vectors: time and phaseangel of the wave into a textfile -------
fid6 = fopen('time.txt','w');
fid7 = fopen('epswave.txt','w');

for i = 1:(tmax/step)
    time(i)=step*i;
    eps(i)=0;

    fprintf(fid6, '%f\n' , time(i));
    fprintf(fid7, '%f\n' , eps(i));

end

fclose(fid6);
fclose(fid7);

%-------------------------------------------------------------------------

fid22 = fopen('wave_matlab.txt','w');
```

```
counter=0;

timestart=time(1);
        j_zeta=0;
        j_vel=0;
        j_acc=0;
        y_zeta=0;

for t=timestart:step:tmax

    counter=counter+1;

        j_zeta=(H/2)*sin(w(1)*t);
        j_vel=w(1)*(H/2)*cos(w(1)*t);
        j_acc=(-w(1)^2)*(H/2)*sin(w(1)*t);
        y_zeta=j_zeta*transb_use;

        zeta_w(counter)=j_zeta;
        vel_w(counter)=j_vel;
        acc_w(counter)=j_acc;
        zeta_b(counter)=y_zeta;

        fprintf(fid22, '%f\n' , zeta_w(counter));

end

fclose(fid22);

vel_b = gradient(zeta_b)*(1/step);
acc_b = gradient(vel_b)*(1/step);

%calculation of template response-----------------------------------------

%import values from textfile ---------------------------------------------

fid8 = fopen('tempparam.txt');
temp = fscanf(fid8, '%f %f %f %f %f %f %f\n');
fclose(fid8);

a33_t=temp(1);
A_z=temp(3);
m_t_dry = temp(5);
Cd_z=temp(7);
p_sw=1025;

fid9 = fopen('wireparam.txt');
wire = fscanf(fid9, '%f %f %f %f\n');
fclose(fid9);

L_c=wire(1);
d_c=wire(2);
E_c = wire(4);
```

```
%-------------------------------------------------------------------------

m_t_w_added=((m_t_dry)+a33_t);
A_c=(pi/4)*d_c^2;
k_c=E_c*A_c/L_c;
b_t=0;
eigenfreq=sqrt(k_c/m_t_w_added);

% save the damping into a textfile----------------------------------------
fid10 = fopen('damp.txt','w');
fprintf(fid10, '%f\n' ,b_t);
fclose(fid10);
%-------------------------------------------------------------------------

d=zeros((tmax/step),1);
v=zeros((tmax/step),1);
a=zeros((tmax/step),1);
d_p=zeros((tmax/step),1);
v_p=zeros((tmax/step),1);

gamma_newmark=1;
beta=0.25;

%establishment of the Load vector (F)-------------------------------------

teller = 0;
for t=timestart:step:tmax
    teller=teller+1;
    F3(teller)=-m_t_w_added*acc_b(teller)-
b_t*vel_b(teller)+0.5*p_sw*vel_b(teller)^2*Cd_z*A_z;
end

%iteration according to the newmark beta method ---------------------------

clear teller;
teller=0;

for i=1:((tmax/step)-1)

    teller=teller+1;

    d_p(teller+1)=d(teller)+step*v(teller)+ (step^2/2)*(1-
2*beta)*a(teller);
    v_p(teller+1)=v(teller)+(1-gamma_newmark)*step*a(teller);
    a(teller+1)=(F3(teller+1)-b_t*v_p(teller+1)-
k_c*d_p(teller+1))/(m_t_w_added+gamma_newmark*step*b_t+beta*step^2*k_c);
    d(teller+1)=(d_p(teller+1)+beta*step^2*a(teller+1));
    v(teller+1)=v_p(teller+1)+gamma_newmark*step*a(teller+1);

end

for i = 1: (tmax/step)
    zeta_t(i)=zeta_b(i)-d(i);
    motiondiff(i)=zeta_b(i)-zeta_t(i);
```

```
end

fid17=fopen('vessel_heave_matlab.txt','w');
fid18=fopen('template_heave_matlab.txt','w');
for i=1:(tmax/step)
fprintf(fid17,'%6.11f\r\n', zeta_b(i));
fprintf(fid18,'%6.11f\r\n', zeta_t(i));
end
fclose(fid17);
fclose(fid18);

%Plotting of the results --------------------------------------------------

set(figure,'units','normalized','outerposition',[0 0 1 1]);
plot(omega,trans_b,'linewidth',2)
xlabel('\omega [rad/s]','fontsize',18);
ylabel('H(\omega) [m/m]','fontsize',18);
title('Transferfunction in heave for the vessel','fontsize',18);
saveas(gcf,'transfer vessel','bmp');

set(figure,'units','normalized','outerposition',[0 0 1 1]);
plot(time,zeta_w,'linewidth',2)
xlabel('Time [s]','fontsize',18);
ylabel('Amplitude [m]','fontsize',18);
title('Time history of the wave','fontsize',18);
saveas(gcf,'time history wave','bmp');

set(figure,'units','normalized','outerposition',[0 0 1 1]);
plot(time,zeta_b,'color','green','linewidth',2)
xlabel('Time [s]','fontsize',18);
ylabel('Amplitude [m]','fontsize',18);
title('Time history of the heave motion of the vessel','fontsize',18);
saveas(gcf,'time history vessel','bmp');

set(figure,'units','normalized','outerposition',[0 0 1 1]);
plot(time,zeta_t,'linewidth',2)
xlabel('Time [s]','fontsize',18);
ylabel('Amplitude [m]','fontsize',18);
title('Time history of the heave motion of the template','fontsize',18);
saveas(gcf,'time history template','bmp');

set(figure,'units','normalized','outerposition',[0 0 1 1]);
plot(time,zeta_b,time,zeta_t,'linewidth',2)
xlabel('Time [s]','fontsize',18);
ylabel('Amplitude [m]','fontsize',18);
title('The template and vessel displacement in heave','fontsize',18);
legend('vessel motion','template motion')
saveas(gcf,'time history template and vessel','bmp');

set(figure,'units','normalized','outerposition',[0 0 1 1]);
plot(time,motiondiff,'color','red','linewidth',2)
xlabel('Time [s]','fontsize',18);
ylabel('Amplitude [m]','fontsize',18);
title('The motion difference between vessel and template in
heave','fontsize',18);
saveas(gcf,'motiondiff regular','bmp');
```

12.1.1.8Drag parameter.m

```
% ***************************************************************************
% Routine:                dragparameter
% ---------------------------------------------------------------------------
% Intension:              Do a parametric study of the horizontal offset
%                         with varying drag force.
%
% Parameter:              Description
% ---------------------------------------------------------------------------
% Fd                      Drag force [N]
% m_t                     Mass of the template [kg]
% a33                     added mass in heave for template[kg]
% A_z                     Projected area of load (z-direction) [m^2]
% w0                      Submerged weight of the template [N]
% p_sw                    Density of sea water [kg/m^3]
% g                       Gravity [m/s^2]
% Te                      Effective tension [N]
% alfa_rad                Angle of the horizontal offset [rad]
% alfa_deg                Angle of the horizontal offset [deg]
% Cd_x                    Drag coefficient of the load (x-direction) [-]
% U_ms                    Towing velocity [m/s]
% U_max                   Maximum towing velocity velocity [m/s]
% U_knot                  Current velocity [knots]
% A_x                     Refference area on the template (x-direction)
[m^2]
% ---------------------------------------------------------------------------
% Programmer:             Torbjørn Aakerøy Olsen
% Last changed:           26.05.2011
% ***************************************************************************
clear all
clc

disp('*****************************************************************' )
disp(' ' )
disp('     Parametric study of the horizontal offset with varying drag
coefficient. ')
disp('     From 80% to 120% of the calculated drag coefficient' )
disp(' ' )
disp('*****************************************************************' )

%import values selected by the user
fid = fopen('tempparam.txt');
temp = fscanf(fid, '%f %f %f %f %f %f\n');
fclose(fid);

a33=temp(1);
A_x=temp(2);
A_z=temp(3);
w0=temp(4);
m_t=temp(5);
Cd_x=temp(6);

fid2 = fopen('towingvel.txt');
tvel = fscanf(fid2, '%f\n');
fclose(fid2);
```

```
U_max=tvel;

% values---------------------------------------------------------------
p_sw = 1025;
g=9.80665;
%---------------------------------------------------------------------------

ph=1.20;
pl=0.80;
pstep=(ph-pl)/(21-1);

    teller=0;
        for p=pl:pstep:ph
            teller=teller+1;
            dragforcepre(teller)=p;
        end
    clear teller

    inc=U_max/20;
    loop=21;
    teller=0;

    for i = 1:21
              teller=teller+1;
              U_ms(teller)=(i-1)*inc;
              U_knot(teller)=U_ms(teller)*1.9438612860586;
    end
    clear teller



    for L=1:21
        teller=0;
             for i = 1:loop
                 teller=teller+1;

Fd(L,teller)=dragforcepre(L)*0.5*p_sw*U_ms(teller)^2*Cd_x*A_x;
                 Te(L,teller)=sqrt(w0^2+Fd(L,teller)^2);
                 alfa_rad(L,teller)=asin(Fd(L,teller)/Te(L,teller));
                 alfa_deg(L,teller)=asin(Fd(L,teller)/Te(L,teller))*180/pi;

             end
    end


param08=alfa_deg(1,:);
param09=alfa_deg(6,:);
param1=alfa_deg(11,:);
param11=alfa_deg(16,:);
param12=alfa_deg(21,:);

fid12 = fopen('drag_mat08.txt','w');
fid13 = fopen('drag_mat09.txt','w');
fid14 = fopen('drag_mat1.txt','w');
fid15 = fopen('drag_mat11.txt','w');
```

```
fid16 = fopen('drag_mat12.txt','w');
fid17 = fopen('param_U_ms.txt','w');

fprintf(fid12, '%6.11f\n' , param08);
fprintf(fid13, '%6.11f\n' , param09);
fprintf(fid14, '%6.11f\n' , param1);
fprintf(fid15, '%6.11f\n' , param11);
fprintf(fid16, '%6.11f\n' , param12);
fprintf(fid17, '%6.11f\n' , U_ms);

fclose(fid12);
fclose(fid13);
fclose(fid14);
fclose(fid15);
fclose(fid16);
fclose(fid17);

% 3D plot of the parametric study
set(figure,'units','normalized','outerposition',[0 0 1 1]);
surf(U_ms,dragforcepre,alfa_deg)
xlabel('Towing velocity [m/s]','fontsize',18)
ylabel('* initial drag coefficient [-]','fontsize',18)
zlabel('Horizontal offset [deg]','fontsize',18)
title('Horizontal offset with varying dragforce','fontsize',18)
colorbar;
saveas(gcf,'dragparam','bmp');
```

12.1.1.9 Addedparameter.m

```
% ***********************************************************************
% Routine:                addedparameter
% -----------------------------------------------------------------------
% Intension:              Do a parametric study of the template motion in
%                         heave, as a function of time and added mass.
%                         The added mass coefficient is to be varied from
%                         80% to 120% of the initial calculated value.
%
% Parameter:              Description
% -----------------------------------------------------------------------
% numfreq                 Number of frequency intervals
% wmax                    Maximum frequency value [rad/s]
% wmin                    Minimum frequency value [rad/s]
% w                       Vector of frequency values [rad/s]
% S                       Vector of wavespectrum values
% k                       Vector of wavenumber values [1/m]
% p_sw                    Density of sea water [kg/m^3]
% trans_b                 Vector with values of the transferfunction for
%                         the vessel
% tmax                    Maximum time [s]
% step                    Timestep
% zeta_w                  Vector with values of the surface elevation [m]
% vel_w                   Vector with values of the surface velocity
%                         in heave [m/s]
% acc_w                   Vector with values of the surface acceleration
%                         in heave [m/s^2]
% zeta_b                  Vector with values of the heave motion of the
%                         vessel [m]
% vel_b                   Vector with values of the heave velocity of the
%                         vessel [m/s]
% acc_b                   Vector with values of the heave acceleration of
%                         the vessel [m/s^2]
% eps                     Random phaseangel of the waves [rad]
% pos                     X-position from the COG of the vessel
% a33_t                   Vector with varying added mass values of
%                         the template [kg]
% m_t                     Vector with mass values of the template [kg]
% m_t_w_added             Vector with mass values of the template
%                         (including added mass) [kg]
% E_c                     E-modulus of the cable [Pa]
% d_c                     Diameter of the cable [m]
% A_c                     Cross-section area of the cable [m^2]
% L_c                     Length of the cable [m]
% k_c                     Stiffness of the cable[N/m]
% b_t                     Damping coefficient of the cable [-]
% eigenfreq               Eigenfrequency [rad/s]
% d                       Vector with approximation of the template
%                         displacement in heave [m]
% v                       Vector with approximation of the template
%                         velocity in heave [m/s]
% a                       Vector with approximation of the template
%                         acceleration in heave [m/s^2]
% d_p                     Vector with predicted values of the template
%                         displacement in heave [m]
% v_p                     Vector with predicted values of the template
```

```
%                           velocity in heave [m/s]
% gamma                     Parameter to determine the stability and accuracy
%                           characteristics of the Newmark_beta algorithm
% beta                      Parameter to determine the stability and accuracy
%                           characteristics of the Newmark_beta algorithm
% F3                        Vector of applied forces to the template [N]
% zeta_t                    Vector with values of the heave motion of the
%                           template [m]
% motiondiff                Vector with the motion difference of the vessel
%                           and the template [m]
% --------------------------------------------------------------------------
% Programmer:               Torbjørn Aakerøy Olsen
% Last changed:             26.05.2011
% **************************************************************************
clear all
clc

disp('****************************************************************************************************' )
disp(' ' )
disp('     Parametric study of the template motion in heave with varying added mass coefficient. From 80% to 120% of the initial calculated value' )
disp(' ' )
disp('****************************************************************************************************' )


% import values from textfile --------------------------------------------
fid1 = fopen('omega.txt');
fid2 = fopen('wavenumber.txt');
fid3 = fopen('spectrum.txt');
fid4 = fopen('time.txt');
fid5 = fopen('epswave.txt');
fid6 = fopen('transb.txt');

w = fscanf(fid1, '%f');
k = fscanf(fid2, '%f');
S = fscanf(fid3, '%f');
time = fscanf(fid4, '%f');
eps = fscanf(fid5, '%f');
trans_b=fscanf(fid6, '%f');

fclose(fid1);
fclose(fid2);
fclose(fid3);
fclose(fid4);
fclose(fid5);
fclose(fid6);

fid11 = fopen('numfreq.txt');
analysis = fscanf(fid11, '%f %f %f\n');
fclose(fid11);

numfreq=analysis(1);
step=analysis(2);
tmax=analysis(3);
```

```
%input values---------------------------------------------------------------
wmax = 4 ;
wmin = 0 ;
g=9.80665;
inc = (wmax-wmin)/numfreq;
p_sw=1025;

%---- Calculation and plotting of surface elevation------------------------

zeta_w=zeros(numfreq,1);
vel_w=zeros(numfreq,1);
acc_w=zeros(numfreq,1);
zeta_b=zeros(numfreq,1);

% import value from textfile ----------------------------------------------
fid7 = fopen('pos.txt');
pos = fscanf(fid7, '%f');
fclose(fid7);
% -------------------------------------------------------------------------
counter=0;

timestart=time(1);

for t=timestart:step:tmax

    counter=counter+1;
        j_zeta=0;
        j_vel=0;
        j_acc=0;
        y_zeta=0;
        for i=1:numfreq

        j_zeta=j_zeta+(sqrt(2*S(i)*inc)*sin((w(i)*t)-(k(i)*pos)+eps(i)));
        j_vel=j_vel+(sqrt(2*S(i)*inc)*(w(i)*cos((w(i)*t)-
(k(i)*pos)+eps(i))));
        j_acc=j_acc+(sqrt(2*S(i)*inc)*(-1*w(i)^2*sin((w(i)*t)-
(k(i)*pos)+eps(i))));

y_zeta=y_zeta+(sqrt(2*(trans_b(i)^2)*S(i)*inc)*cos((w(i)*t)+eps(i)));

        end

        zeta_w(counter)=j_zeta;
        vel_w(counter)=j_vel;
        acc_w(counter)=j_acc;
        zeta_b(counter)=y_zeta;
end

vel_b = gradient(zeta_b)*(1/step);
acc_b = gradient(vel_b)*(1/step);

%calculation of template response-----------------------------------------
% import values from textfile ---------------------------------------------
fid8 = fopen('wireparam.txt');
```

```
wire = fscanf(fid8, '%f %f %f %f\n');
fclose(fid8);

L_c=wire(1);
d_c=wire(2);
E_c=wire(4);
%-------------------------------------------------------------------------

% Parametric study of the motiondifference in heave between the vessel and
% the template.
%
% The added mass is set as a variable

% import values from textfile --------------------------------------------
fid9 = fopen('tempparam.txt');
temp = fscanf(fid9, '%f %f %f %f %f %f %f\n');
fclose(fid9);

fid10 = fopen('damp.txt');
damp_init = fscanf(fid10, '%f\n');
fclose(fid10);

a33_init=temp(1);
m_t = temp(5);

b_t=damp_init;
%-------------------------------------------------------------------------

m_t_w_added=((m_t)+a33_init);
A_c=(pi/4)*d_c^2;
k_c=E_c*A_c/L_c;

eigenfreq=sqrt(k_c/m_t_w_added);

d=zeros(numfreq,21);
v=zeros(numfreq,21);
a=zeros(numfreq,21);
d_p=zeros(numfreq,21);
v_p=zeros(numfreq,21);

gamma=1;
beta=0.25;

%establishment of the Load vector (F)-------------------------------------

ph=1.2;
pl=0.8;
pstep=(ph-pl)/(21-1);

teller=0;
for p=pl:pstep:ph
    teller=teller+1;
    a33_t(teller)=p*(a33_init);
    m_t_w_added(teller)=((m_t)+a33_t(teller));
    addedpre(teller)=p;
```

```
end
clear teller

for L=1:21

            teller = 0;
            for t=timestart:step:tmax
                teller=teller+1;
                F3_damp(teller,L)=-m_t_w_added(L)*acc_b(teller)-
b_t*vel_b(teller);
            end

end

%iteration according to the newmark beta method ---------------------------

clear teller teller2;

for L=1:21

            teller=0;
            for i=1:((tmax/step)-1)

                teller=teller+1;

                d_p(teller+1,L)=d(teller,L)+step*v(teller,L)+
(step^2/2)*(1-2*beta)*a(teller,L);
                v_p(teller+1,L)=v(teller,L)+(1-gamma)*step*a(teller,L);
                a(teller+1,L)=(F3_damp(teller+1,L)-b_t*v_p(teller+1,L)-
k_c*d_p(teller+1,L))/(m_t_w_added(L)+gamma*step*b_t+beta*step^2*k_c);
                d(teller+1,L)=(d_p(teller+1,L)+beta*step^2*a(teller+1,L));
                v(teller+1,L)=v_p(teller+1,L)+gamma*step*a(teller+1,L);

            end
end

clear teller teller2;

for L=1:21

    teller=0;
            for i = 1:(tmax/step)
                teller=teller+1;

                zeta_t(teller,L)=zeta_b(teller)-d(teller,L);
                motiondiff(teller,L)=(zeta_b(teller)-zeta_t(teller,L));
                absmotiondiff(teller,L)=abs(zeta_b(teller)-
zeta_t(teller,L));

            end
end

param08=zeta_t(:,1);
param09=zeta_t(:,6);
```

```
param1=zeta_t(:,11);
param11=zeta_t(:,16);
param12=zeta_t(:,21);

fid12 = fopen('template_heave_mat_3h08.txt','w');
fid13 = fopen('template_heave_mat_3h09.txt','w');
fid14 = fopen('template_heave_mat_3h1.txt','w');
fid15 = fopen('template_heave_mat_3h11.txt','w');
fid16 = fopen('template_heave_mat_3h12.txt','w');

fprintf(fid12, '%6.11f\n' , param08);
fprintf(fid13, '%6.11f\n' , param09);
fprintf(fid14, '%6.11f\n' , param1);
fprintf(fid15, '%6.11f\n' , param11);
fprintf(fid16, '%6.11f\n' , param12);

fclose(fid12);
fclose(fid13);
fclose(fid14);
fclose(fid15);
fclose(fid16);

% 3D plot of the parametric study
set(figure,'units','normalized','outerposition',[0 0 1 1]);
surf(addedpre,time,zeta_t)
xlabel('* initial added mass coefficient [-]','fontsize',18)
ylabel('time [s]','fontsize',18)
zlabel('motion response \eta/\eta_{a} [m/m]','fontsize',18)
title('Template motion in heave with varying added mass
coefficient','fontsize',18)
colorbar;
saveas(gcf,'addedparam','bmp');
```

12.1.2 Curve fitting

12.1.2.1 Transfer.m

```
% ************************************************************************
% Routine:                  transfer
% ------------------------------------------------------------------------
% Intension:                Find the transfer function in heave at the hang
%                           off point from imported textfile
%
% Method:                   Use the curvefitting tool to establish an
%                           equation for the transferfuction in heave
%
%
% Parameter:                Description
% ------------------------------------------------------------------------
% transxdata                Given x-values from textfile transorca.txt
% transydata                Given y-values from textfile transorca.txt
% ------------------------------------------------------------------------
% Programmer:               Torbjørn Aakerøy Olsen
% Last changed:             26.05.2011
% ************************************************************************


transxdata=[4.189,2.094,1.571,1.257,1.047,0.898,0.838,0.785,0.739,0.698,0.6
61,0.628,0.598,0.571,0.546,0.524,0.503,0.491,0.483,0.472,0.465,0.449,0.433,
0.419,0.393,0.370,0.349,0.314,0.279,0.251,0.209];
transydata=[0.000,0.003,0.010,0.056,0.312,0.270,0.068,0.154,0.327,0.463,0.5
69,0.651,0.715,0.766,0.806,0.838,0.864,0.875,0.885,0.894,0.902,0.916,0.929,
0.938,0.953,0.964,0.972,0.983,0.990,0.994,0.997];

cftool(transxdata,transydata);

% General model Rat55:
%      f(x) =
%                 (p1*x^5 + p2*x^4 + p3*x^3 + p4*x^2 + p5*x + p6) /
%                 (x^5 + q1*x^4 + q2*x^3 + q3*x^2 + q4*x + q5)
% Coefficients (with 95% confidence bounds):
%        p1 =      -0.194  (-0.3746, -0.01333)
%        p2 =        1.76  (0.1575, 3.362)
%        p3 =      -5.679  (-10.76, -0.5975)
%        p4 =       8.408  (1, 15.82)
%        p5 =      -5.793  (-10.83, -0.7572)
%        p6 =       1.505  (0.2119, 2.798)
%        q1 =       -1.07  (-2.901, 0.7618)
%        q2 =      -3.239  (-9.42, 2.942)
%        q3 =       7.347  (-0.9763, 15.67)
%        q4 =      -5.493  (-10.77, -0.2185)
%        q5 =       1.471  (0.1563, 2.785)
%
% Goodness of fit:
%   SSE: 0.0004545
%   R-square: 0.9999
%   Adjusted R-square: 0.9998
%   RMSE: 0.004767
```

12.1.3 Comparison with regular waves

12.1.3.1 Compare_regular.m

```
% ***********************************************************************
% Routine:                compare_regular
% -----------------------------------------------------------------------
% Intension:              Compare the timeseries produced in matlab with
%                         the ones produced in orcaflex
%
% Method:                 Plot the wave elevation, vessel displacement and
%                         template displacement in heave for both matlab
%                         and orcaflex timeseries
%
% Parameter:              Description
% -----------------------------------------------------------------------
% time                    Vector of the time of the timeseries
% temp_mat                Vector of the timeseries of the template in
Matlab
% vess_mat                Vector of the timeseries of the vessel in Matlab
% wave_mat                Vector of the timeseries of the wave in Matlab
% temp_orc                Vector of the timeseries of the template in
Orcaflex
% vess_orc                Vector of the timeseries of the vessel in
Orcaflex
% wave_orc                Vector of the timeseries of the wave in Orcaflex
% -----------------------------------------------------------------------
% Programmer:             Torbjørn Aakerøy Olsen
% Last changed:           26.05.2011
% ***********************************************************************
clear all
clc

fid1 = fopen('time.txt');
time = fscanf(fid1, '%f');
fclose(fid1);

fid2 = fopen('template_heave_matlab.txt');
temp_mat = fscanf(fid2, '%f');
fclose(fid2);

fid3 = fopen('vessel_heave_matlab.txt');
vess_mat = fscanf(fid3, '%f');
fclose(fid3);

fid4 = fopen('wave_matlab.txt');
wave_mat = fscanf(fid4, '%f');
fclose(fid4);

fid5 = fopen('template_heave_orca_regular.txt');
temp_orc = fscanf(fid5, '%f');
fclose(fid5);

fid6 = fopen('vessel_heave_orca_regular.txt');
vess_orc = fscanf(fid6, '%f');
fclose(fid6);
```

```
fid7 = fopen('wave_orca_regular.txt');
wave_orc = fscanf(fid7, '%f');
fclose(fid7);

motiondiff_mat=vess_mat-temp_mat;
motiondiff_orc=vess_orc-temp_orc;

diff_wave=abs(wave_mat-wave_orc);
diff_vess=abs(vess_mat-vess_orc);
diff_temp=abs(temp_mat-temp_orc);
diff_motiondiff=abs(motiondiff_mat-motiondiff_orc);

set(figure,'units','normalized','outerposition',[0 0 1 1]);
plot(time,wave_mat,time,wave_orc)
xlabel('time [s]','fontsize',18);
ylabel('amplitude [m]','fontsize',18);
title('timehistory of the surface elevation','fontsize',18);
legend('Matlab','Orcaflex')
saveas(gcf,'time_wave_both_regular','bmp');

set(figure,'units','normalized','outerposition',[0 0 1 1]);
plot(time,vess_mat,time,vess_orc)
xlabel('Time [s]','fontsize',18);
ylabel('amplitude [m]','fontsize',18);
title('Timehistory of the heave motion of the vessel','fontsize',18);
legend('Matlab','Orcaflex')
saveas(gcf,'time_vessel_both_regular','bmp');

set(figure,'units','normalized','outerposition',[0 0 1 1]);
plot(time,temp_mat,time,temp_orc)
xlabel('Time [s]','fontsize',18);
ylabel('amplitude [m]','fontsize',18);
title('Timehistory of the heave motion of the template','fontsize',18);
legend('Matlab','Orcaflex')
saveas(gcf,'time_template_both_regular','bmp');

set(figure,'units','normalized','outerposition',[0 0 1 1]);
plot(time,vess_mat,time,temp_mat,time,vess_orc,time,temp_orc)
xlabel('Time [s]','fontsize',18);
ylabel('amplitude [m]','fontsize',18);
title('Timehistory of the heave motion of the template and
vessel','fontsize',18);
legend('Matlab vessel','Matlab template','Orcaflex vessel','Orcaflex
template')
saveas(gcf,'time_template_vessel_both_regular','bmp');

set(figure,'units','normalized','outerposition',[0 0 1 1]);
plot(time,motiondiff_mat,time,motiondiff_orc)
xlabel('Time [s]','fontsize',18);
ylabel('amplitude [m]','fontsize',18);
title('Timehistory of the absolute motion difference between the vessel and
the template','fontsize',18);
legend('Matlab','Orcaflex')
saveas(gcf,'time_motiondiff_both_regular','bmp');
```

12.1.4 Comparison with irregular waves

12.1.4.1 Compare_wave_3h.m

```
% ************************************************************************
% Routine:                  compare_wave_3h
% ------------------------------------------------------------------------
% Intension:                Compare the timeseries produced in matlab with
%                           the one produced in orcaflex
%
% Method:                   Do a fft of the timehistory of the wave in both
%                           Matlab and Orcaflex, and find the frequency at
%                           which the signal is at its strongest.
%                           Calculate the standard deviation and skewness for
%                           each timeseries
%
% Parameter:                Description
% ------------------------------------------------------------------------
% numfreq                   Number of frequencies [-]
% step                      Timestep of the analysis [s]
% tmax                      Length of analysis [s]
% mat                       Vector with the timseries produced in matlab
% orc                       Vector with the timeseries produced in Orcaflex
% time                      Vector with the time of the the timeseries
% mean_orca                 Mean of the orcaflex timeseries
% mean_matl                 Mean of the matlab timeseries
% skewness_matlab           Skewness of the matlab timeseries
% skewness_orcaflex         Skewness of the Orcaflex timeseries
% std_deviation_orca        Standard deviation of the Orcaflex timeseries
% std_deviation_matl        Standard deviation of the Matlab timeseries
% Fs                        Sampling frequency [Hz]
% frequency_peak_matlab     Frequency peak of the FFT of the Matlab
timeseries
% frequency_peak_orcaflex Frequency peak of the FFT of the Orcaflex
timeseries
% ------------------------------------------------------------------------
% Programmer:               Torbjørn Aakerøy Olsen
% Last changed:             26.05.2011
% ************************************************************************


%code- ref mathworks:
%http://www.mathworks.com/help/toolbox/signal/f12-6587.html#bruvukd-1
clear all
clc

fid10 = fopen('numfreq.txt');
analysis = fscanf(fid10, '%f %f %f\n');
fclose(fid10);

numfreq=analysis(1);
step=analysis(2);
tmax=analysis(3);


fid1 = fopen('time.txt');
time = fscanf(fid1, '%f');
```

```
fclose(fid1);

fid2 = fopen('wave_matlab.txt');
mat = fscanf(fid2, '%f');
fclose(fid2);

fid3 = fopen('wave_orca_3h_1000comp_ny.txt');
orc = fscanf(fid3, '%f');
fclose(fid3);


%standard deviation and skewness of the timeseries -----------------------
N_orc=length(orc);
N_matl=length(mat);
mean_orca=0;
mean_matl=0;

for i=1:N_orc
   mean_orca=mean_orca+(orc(i)/N_orc);
end

m2_orca=0;
m3_orca=0;
for i=1:N_orc
   m2_orca=m2_orca+(((orc(i)-mean_orca)^2)/N_orc);
end

for i=1:N_orc
   m3_orca=m3_orca+(((orc(i)-mean_orca)^3)/N_orc);
end

for i=1:N_matl
   mean_matl=mean_matl+(mat(i)/N_matl);
end

m2_matl=0;
m3_matl=0;
for i=1:N_matl
   m2_matl=m2_matl+(((mat(i)-mean_matl)^2)/N_matl);
end

for i=1:N_matl
   m3_matl=m3_matl+(((mat(i)-mean_matl)^3)/N_matl);
end

skewness_matlab=m3_matl/((m2_matl)^(3/2));
skewness_orcaflex=m3_orca/((m2_orca)^(3/2));
std_deviation_orca=sqrt(m2_orca);
std_deviation_matl=sqrt(m2_matl);

%-------------------------------------------------------------------------

Fs=1/step;

nfft = 2^(nextpow2(length(mat)));    % Find next power of 2
```

```
fftx = fft(mat,nfft);

NumUniquePts = ceil((nfft+1)/2);
fftx = fftx(1:NumUniquePts);
mx = abs(fftx);

mx = mx/length(time);
mx = mx.^2;
if rem(nfft, 2)==0;   % Odd nfft excludes Nyquist
  mx(2:end) = mx(2:end)*2;
else
  mx(2:end -1) = mx(2:end -1)*2;
end

ff = (0:NumUniquePts-1)*Fs/nfft;


nfft2 = 2^(nextpow2(length(orc)));   % Find next power of 2
fftx2 = fft(orc,nfft2);

NumUniquePts2 = ceil((nfft2+1)/2);
fftx2 = fftx2(1:NumUniquePts2);
mx2 = abs(fftx2);

mx2 = mx2/length(time);
mx2 = mx2.^2;
if rem(nfft2, 2)==0;   % Odd nfft excludes Nyquist
  mx2(2:end) = mx2(2:end)*2;
else
  mx2(2:end -1) = mx2(2:end -1)*2;
end

ff2 = (0:NumUniquePts2-1)*Fs/nfft2;

set(figure,'units','normalized','outerposition',[0 0 1 1]);
plot(time,mat,time,orc)
xlabel('time [s]','fontsize',18);
ylabel('amplitude [m]','fontsize',18);
title('timehistory of the surface elevation','fontsize',18);
legend('Matlab','Orcaflex')
saveas(gcf,'time_wave_both_3h','bmp');


set(figure,'units','normalized','outerposition',[0 0 1 1]);
plot(ff,mx,ff2,mx2)
xlabel('Frequency [Hz]','fontsize',18);
ylabel('Power','fontsize',18);
set(gca,'Xlim',[0 0.5],'Ylim',[min(mx) max(mx)]);
title('FFT of the surface elevation','fontsize',18);
legend('Matlab','Orcaflex')
saveas(gcf,'FFT_wave_3h','bmp');


set(figure,'units','normalized','outerposition',[0 0 1 1]);
plot(ff,mx)
```

```
xlabel('Frequency [Hz]','fontsize',18);
ylabel('Power','fontsize',18);
set(gca,'Xlim',[0 0.5],'Ylim',[min(mx) max(mx)]);
title('FFT of the surface elevation','fontsize',18);
legend('Matlab')
saveas(gcf,'FFT_wave_3h_matlab','bmp');

[frequency_peak_matlab,amplitude_matlab]=ginput;

set(figure,'units','normalized','outerposition',[0 0 1 1]);
plot(ff2,mx2)
xlabel('Frequency [Hz]','fontsize',18);
ylabel('Power','fontsize',18);
set(gca,'Xlim',[0 0.5],'Ylim',[min(mx2) max(mx2)]);
title('FFT of the surface elevation','fontsize',18);
legend('Orcaflex')
saveas(gcf,'FFT_wave_3h_orcaflex','bmp');

[frequency_peak_orcaflex,amplitude_orcaflex]=ginput;

fprintf('The frequency peak in Matlab is [Hz].......................:%f\n',
frequency_peak_matlab);
fprintf('The frequency peak in Orcaflex is [Hz].....................:%f\n',
frequency_peak_orcaflex);
fprintf('The standard deviation of the timehistory in Matlab is.....:%f\n',
std_deviation_matl);
fprintf('The standard deviation of the timehistory in Orcaflex is...:%f\n',
std_deviation_orca);
```

12.1.5 Parametric study of the template motion

12.1.5.1 Compare_template_3h_08.m

```
% ***********************************************************************
% Routine:                  compare_template_3h_08
% -----------------------------------------------------------------------
% Intension:                Compare the timeseries of the template produced
%                           in matlab with the one produced in orcaflex. For
%                           80 % of the initial added mass coefficient
%
% Method:                   Do a fft of the timehistory of the template in
both
%                           Matlab and Orcaflex, and find the frequency at
%                           which the signal is at its strongest.
%                           Calculate the standard deviation and skewness for
%                           each timeseries
%
% Parameter:                Description
% -----------------------------------------------------------------------
% numfreq                   Number of frequencies [-]
% step                      Timestep of the analysis [s]
% tmax                      Length of analysis [s]
% mat                       Vector with the timseries produced in matlab
% orc                       Vector with the timeseries produced in Orcaflex
% time                      Vector with the time of the the timeseries
% mean_orca                 Mean of the orcaflex timeseries
% mean_matl                 Mean of the matlab timeseries
% skewness_matlab           Skewness of the matlab timeseries
% skewness_orcaflex         Skewness of the Orcaflex timeseries
% std_deviation_orca        Standard deviation of the Orcaflex timeseries
% std_deviation_matl        Standard deviation of the Matlab timeseries
% Fs                        Sampling frequency [Hz]
% frequency_peak_matlab     Frequency peak of the FFT of the Matlab
timeseries
% frequency_peak_orcaflex Frequency peak of the FFT of the Orcaflex
timeseries
% -----------------------------------------------------------------------
% Programmer:               Torbjørn Aakerøy Olsen
% Last changed:             26.05.2011
% ***********************************************************************

%code- ref mathworks:
%http://www.mathworks.com/help/toolbox/signal/f12-6587.html#bruvukd-1
clear all
clc

fid10 = fopen('numfreq.txt');
analysis = fscanf(fid10, '%f %f %f\n');
fclose(fid10);

numfreq=analysis(1);
step=analysis(2);
tmax=analysis(3);

fid1 = fopen('time.txt');
time = fscanf(fid1, '%f');
```

```
fclose(fid1);

fid2 = fopen('template_heave_mat_3h08.txt');
mat = fscanf(fid2, '%f');
fclose(fid2);

fid3 = fopen('template_heave_orca_3h_08_1000comp.txt');
orc = fscanf(fid3, '%f');
fclose(fid3);


%standard deviation of the timeseries orcaflex-----------------------------
N_orc=length(orc);
N_matl=length(mat);
mean_orca=0;
mean_matl=0;

for i=1:N_orc
   mean_orca=mean_orca+(orc(i)/N_orc);
end

m2_orca=0;
for i=1:N_orc
   m2_orca=m2_orca+(((orc(i)-mean_orca)^2)/N_orc);
end

m3_orca=0;
for i=1:N_orc
   m3_orca=m3_orca+(((orc(i)-mean_orca)^2)/N_orc);
end

for i=1:N_matl
   mean_matl=mean_matl+(mat(i)/N_matl);
end

m2_matl=0;
for i=1:N_matl
   m2_matl=m2_matl+(((mat(i)-mean_matl)^2)/N_matl);
end

m3_matl=0;
for i=1:N_matl
   m3_matl=m3_matl+(((mat(i)-mean_matl)^2)/N_matl);
end

std_deviation_orca=sqrt(m2_orca);
std_deviation_matl=sqrt(m2_matl);
skewness_matlab=m3_matl/((m2_matl)^(3/2));
skewness_orcaflex=m3_orca/((m2_orca)^(3/2));

%--------------------------------------------------------------------------

Fs=1/step;

nfft = 2^(nextpow2(length(mat)));    % Find next power of 2
```

```
fftx = fft(mat,nfft);

NumUniquePts = ceil((nfft+1)/2);
fftx = fftx(1:NumUniquePts);
mx = abs(fftx);

mx = mx/length(time);
mx = mx.^2;
if rem(nfft, 2)==0;   % Odd nfft excludes Nyquist
  mx(2:end) = mx(2:end)*2;
else
  mx(2:end -1) = mx(2:end -1)*2;
end

ff = (0:NumUniquePts-1)*Fs/nfft;


nfft2 = 2^(nextpow2(length(orc)));   % Find next power of 2
fftx2 = fft(orc,nfft2);

NumUniquePts2 = ceil((nfft2+1)/2);
fftx2 = fftx2(1:NumUniquePts2);
mx2 = abs(fftx2);

mx2 = mx2/length(time);
mx2 = mx2.^2;
if rem(nfft2, 2)==0;   % Odd nfft excludes Nyquist
  mx2(2:end) = mx2(2:end)*2;
else
  mx2(2:end -1) = mx2(2:end -1)*2;
end

ff2 = (0:NumUniquePts2-1)*Fs/nfft2;


set(figure,'units','normalized','outerposition',[0 0 1 1]);
plot(time,mat,time,orc)
xlabel('time [s]','fontsize',18);
ylabel('amplitude [m]','fontsize',18);
title('timehistory of the template motion (80% added mass
coefficient)','fontsize',18);
legend('Matlab','Orcaflex')
saveas(gcf,'time_template_both_3h_08','bmp');


set(figure,'units','normalized','outerposition',[0 0 1 1]);
plot(ff,mx,ff2,mx2)
xlabel('Frequency [Hz]','fontsize',18);
ylabel('Power','fontsize',18);
set(gca,'Xlim',[0 0.5],'Ylim',[min(mx) max(mx)]);
title('FFT of the template motion (80% added mass
coefficient)','fontsize',18);
legend('Matlab','Orcaflex')
saveas(gcf,'FFT_template_3h_08','bmp');
```

```
set(figure,'units','normalized','outerposition',[0 0 1 1]);
plot(ff,mx)
xlabel('Frequency [Hz]','fontsize',18);
ylabel('Power','fontsize',18);
set(gca,'Xlim',[0 0.5],'Ylim',[min(mx) max(mx)]);
title('FFT of the template motion (80% added mass
coefficient)','fontsize',18);
legend('Matlab')
saveas(gcf,'FFT_template_3h_matlab_08','bmp');

[frequency_peak_matlab,amplitude_matlab]=ginput;

set(figure,'units','normalized','outerposition',[0 0 1 1]);
plot(ff2,mx2)
xlabel('Frequency [Hz]','fontsize',18);
ylabel('Power','fontsize',18);
set(gca,'Xlim',[0 0.5],'Ylim',[min(mx2) max(mx2)]);
title('FFT of the template motion (80% added mass
coefficient)','fontsize',18);
legend('Orcaflex')
saveas(gcf,'FFT_template_3h_orcaflex_08','bmp');

[frequency_peak_orcaflex,amplitude_orcaflex]=ginput;

fprintf('The frequency peak in Matlab is [Hz].......................:%f\n',
frequency_peak_matlab);
fprintf('The frequency peak in Orcaflex is [Hz].....................:%f\n',
frequency_peak_orcaflex);
fprintf('The standard deviation of the timehistory in Matlab is.....:%f\n',
std_deviation_matl);
fprintf('The standard deviation of the timehistory in Orcaflex is...:%f\n',
std_deviation_orca);
```

12.1.6 Parametric study of the offset angle

12.1.6.1 Compare_U_ms.m

```
% ***************************************************************************
% Routine:                  compare_U_ms
% ---------------------------------------------------------------------------
% Intension:                Compare the offset angle in Matlab and Orcaflex
%                           as a function of the towing velocity with a
%                           varying drag coefficient.(80-120%)
%
% Method:                   Compare the offset angle in Matlab with the one
%                           in Orcaflex by plotting them in the same graph.
%                           This is done for 80%-120% of the initial
%                           added mass coefficient.
%
% Parameter:                Description
% ---------------------------------------------------------------------------
% drag08_mat                Offset angle in Matlab with 80% of the initial
%                           added mass coefficient [deg]
% drag09_mat                Offset angle in Matlab with 90% of the initial
%                           added mass coefficient [deg]
% drag1_mat                 Offset angle in Matlab with 100% of the initial
%                           added mass coefficient [deg]
% drag11_mat                Offset angle in Matlab with 110% of the initial
%                           added mass coefficient [deg]
% drag12_mat                Offset angle in Matlab with 120% of the initial
%                           added mass coefficient [deg]
% drag08_orc                Offset angle in Orcaflex with 80% of the initial
%                           added mass coefficient [deg]
% drag09_orc                Offset angle in Orcaflex with 90% of the initial
%                           added mass coefficient [deg]
% drag1_orc                 Offset angle in Orcaflex with 100% of the initial
%                           added mass coefficient [deg]
% drag11_orc                Offset angle in Orcaflex with 110% of the initial
%                           added mass coefficient [deg]
% drag12_orc                Offset angle in Orcaflex with 120% of the initial
%                           added mass coefficient [deg]
% U_ms                      Vector of towing velocities [m/s]
% ---------------------------------------------------------------------------
% Programmer:               Torbjørn Aakerøy Olsen
% Last changed:             26.05.2011
% ***************************************************************************


clear all
clc

fid1 = fopen('drag_mat08.txt');
fid2 = fopen('drag_mat09.txt');
fid3 = fopen('drag_mat1.txt');
fid4 = fopen('drag_mat11.txt');
fid5 = fopen('drag_mat12.txt');
fid6 = fopen('drag_orca_08.txt');
fid7 = fopen('drag_orca_09.txt');
fid8 = fopen('drag_orca_1.txt');
fid9 = fopen('drag_orca_11.txt');
```

```
fid10 = fopen('drag_orca_12.txt');
fid11 = fopen('param_U_ms.txt');
fid12 = fopen('drag_orca_simple_model.txt');


drag08_mat = fscanf(fid1, '%f');
drag09_mat = fscanf(fid2, '%f');
drag1_mat = fscanf(fid3, '%f');
drag11_mat = fscanf(fid4, '%f');
drag12_mat = fscanf(fid5, '%f');
drag08_orc = fscanf(fid6, '%f');
drag09_orc = fscanf(fid7, '%f');
drag1_orc = fscanf(fid8, '%f');
drag11_orc = fscanf(fid9, '%f');
drag12_orc = fscanf(fid10, '%f');
U_ms = fscanf(fid11, '%f');
dragsimple_orc = fscanf(fid12, '%f');

fclose(fid1);
fclose(fid2);
fclose(fid3);
fclose(fid4);
fclose(fid5);
fclose(fid6);
fclose(fid7);
fclose(fid8);
fclose(fid9);
fclose(fid10);
fclose(fid11);
fclose(fid12);



set(figure,'units','normalized','outerposition',[0 0 1 1]);
plot(U_ms,drag08_mat,U_ms,drag08_orc)
xlabel('Towing velocity [m/s]','fontsize',18);
ylabel('Offset angle [deg]','fontsize',18);
title('Offset angle as a function of the towing velocity. (80% of the
dragcoefficient)','fontsize',18);
legend('Matlab','Orcaflex')
saveas(gcf,'dragparam_08','bmp');

set(figure,'units','normalized','outerposition',[0 0 1 1]);
plot(U_ms,drag09_mat,U_ms,drag09_orc)
xlabel('Towing velocity [m/s]','fontsize',18);
ylabel('Offset angle [deg]','fontsize',18);
title('Offset angle as a function of the towing velocity. (90% of the
dragcoefficient)','fontsize',18);
legend('Matlab','Orcaflex')
saveas(gcf,'dragparam_09','bmp');

set(figure,'units','normalized','outerposition',[0 0 1 1]);
plot(U_ms,drag1_mat,U_ms,drag1_orc)
xlabel('Towing velocity [m/s]','fontsize',18);
ylabel('Offset angle [deg]','fontsize',18);
```

```
title('Offset angle as a function of the towing velocity. (100% of the
dragcoefficient)','fontsize',18);
legend('Matlab','Orcaflex')
saveas(gcf,'dragparam_1','bmp');

set(figure,'units','normalized','outerposition',[0 0 1 1]);
plot(U_ms,drag11_mat,U_ms,drag11_orc)
xlabel('Towing velocity [m/s]','fontsize',18);
ylabel('Offset angle [deg]','fontsize',18);
title('Offset angle as a function of the towing velocity. (110% of the
dragcoefficient)','fontsize',18);
legend('Matlab','Orcaflex')
saveas(gcf,'dragparam_11','bmp');

set(figure,'units','normalized','outerposition',[0 0 1 1]);
plot(U_ms,drag12_mat,U_ms,drag12_orc)
xlabel('Towing velocity [m/s]','fontsize',18);
ylabel('Offset angle [deg]','fontsize',18);
title('Offset angle as a function of the towing velocity. (120% of the
dragcoefficient)','fontsize',18);
legend('Matlab','Orcaflex')
saveas(gcf,'dragparam_12','bmp');

set(figure,'units','normalized','outerposition',[0 0 1 1]);
plot(U_ms,drag11_mat,U_ms,dragsimple_orc)
xlabel('Towing velocity [m/s]','fontsize',18);
ylabel('Offset angle [deg]','fontsize',18);
title('Offset angle as a function of the towing velocity. (Orcaflex simple
model)','fontsize',18);
legend('Matlab','Orcaflex')
saveas(gcf,'dragparam_simple','bmp');
```

12.1.7 Extreme statistics with all peaks

12.1.7.1 Extreme_analysis_all.m

```
% ***********************************************************************
% Routine:                  extreme_analysis_all
% -----------------------------------------------------------------------
% Intension:                Calculate the extreme statistics from all N
%                           time histories
%
% Method:                   Find the average of the N largest peaks. One for
%                           each of the timeseries
%                           Find the average of the N expected extreme values
%                           according to the weibull fit of each sample
%                           Find the Gumbel extreme of the N largest peaks
%
%
% Parameter:                Description
% -----------------------------------------------------------------------
% N                         Number of sample maxima
% maxpeaks                  Vector of all N max peaks
% extremes_weibull          Vector of all N expected extremes from the
%                           Weibull fitted data
% av_maxpeaks               Average of all the max peaks
% av_weibull                Average of all the expected extremes from the
%                           weibull fitted data
% m1                        1. spectral moment of the time history
% m2                        2. spectral moment of the time history
% m3                        3. spectral moment of the time history% variance
% skewness                  Estimate of the skewness of the N maximas
% cov                       Estimate of the co variance of the N maximas
% alpha_est                 Estimate of the Gumbel parameter alpha. Describes
%                           the characteristic largest value of the initial
%                           variate X
% u_est                     Estimate of the Gumbel parameter u. Meassure of
%                           the inverse dispersion of Xn
% Gumbel_ex                 Expected extreme value of the Gumbel fitted
%                           extremes
% -----------------------------------------------------------------------
% Programmer:               Torbjørn Aakerøy Olsen
% Last changed:             26.05.2011
% ***********************************************************************
clear all
clc
extreme1
extreme2
extreme3
extreme4
extreme5
extreme6
extreme7
extreme8
extreme9
extreme10
extreme11
extreme12
extreme13
extreme14
```

```
extreme15
extreme16
extreme17
extreme18
extreme19
extreme20

fid1 = fopen('maxpeak.txt');
fid2 = fopen('expected_extreme_weibull.txt');

maxpeaks = fscanf(fid1, '%f');
extremes_weibull = fscanf(fid2, '%f');

fclose(fid1);
fclose(fid2);

N=length(maxpeaks);

av_maxpeaks=sum(maxpeaks)/N;
av_weibull=sum(extremes_weibull)/N;
%calculation of central moments--------------------------------------------
m1=0;
m2=0;
m3=0;
for i=1:N

    m2=m2+((maxpeaks(i)-av_maxpeaks)^2)/N;
    m3=m3+((maxpeaks(i)-av_maxpeaks)^3)/N;
end

variance=m2;
skewness=m3/(m2^(3/2));
cov=sqrt(m2)/m1;
%---Gumbel of extremes-----------------------------------------------------
alpha_est=(1.06283)/sqrt(m2);
u_est=av_maxpeaks-(0.52355/alpha_est);
Gumbel_ex=u_est+(0.57722/alpha_est);

fprintf('The mean of the maximum value of each sample i:%f\n',
av_maxpeaks);
fprintf('The mean of the expected extremes (from a Weibull fit) of each
sample is:%f\n', av_weibull);
fprintf('Expected maxima of the extremes (gumbel):%f\n', Gumbel_ex);
```

12.1.7.2 Extreme1.m

```
% ************************************************************************
% Routine:                  extreme1
% ------------------------------------------------------------------------
% Intension:                Calculate the extreme statistics of a time
%                           history
%
% Method:                   Find the largest peak of the timeseries
%                           Fit the data to the weibull distribution and find
%                           the expected extreme value
%
%
% Parameter:                Description
% ------------------------------------------------------------------------
% time                      Vector with the time of the simulation
% amp                       Vector with the amplitude of motion of the
%                           simulation
% maxpeak                   Maximum peak of the simulation
% mean                      mean of the time history
% m1                        1. spectral moment of the time history
% m2                        2. spectral moment of the time history
% m3                        3. spectral moment of the time history
% N                         Number of samples
% expec                     Estimation of the expected value of the time
%                           history
% variance                  Estimation of the variance of the time history
% skewness                  Estimation of the skewness of the time history
% cov                       Estimation of the co variance of the time
%                           history
% a                         Estimate of the Weibull location parameter
% b                         Estimate of the Weibull scaling parameter
% c                         Estimate of the Weibull shape factor
% expec_extreme             Expected extreme value of the weibull fitted data
% ------------------------------------------------------------------------
% Programmer:               Torbjørn Aakerøy Olsen
% Last changed:             26.05.2011
% ************************************************************************

clear all
clc

fid1 = fopen('timeex.txt');
time = fscanf(fid1, '%f');
fclose(fid1);

fid2 = fopen('sample1.txt');
amp = fscanf(fid2, '%f');
fclose(fid2);

maxpeak=max(amp);

fid3 = fopen('maxpeak.txt','w');
fprintf(fid3, '%f\n' , maxpeak);
fclose(fid3);

mean=0;
```

```
for i=1:(length(amp))
    mean=mean+(amp(i)/length(amp));
end

[pks,locs]=findpeaks(amp,'minpeakheight',mean,'sortstr','ascend');

N=length(pks);

% calculating the central moments of the sample----------------------------
m1=0;
m2=0;
m3=0;

for i=1:N
    m1=m1+(pks(i)-mean)/N;
    m2=m2+((pks(i)-mean)^2)/N;
    m3=m3+((pks(i)-mean)^3)/N;
end

expec=m1;
variance=m2;
skewness=m3/(m2^(3/2));
cov=sqrt(m2)/m1;

%------Estimate the Gumbell extreme parameters in terms of the weibull
%parameters in the initial distribution
c=0.1;
cov_est=sqrt(gamma(1+(2/c))-(gamma(1+(1/c)))^2)/(gamma(1+(1/c))*(1-
(1/(N^(1/c)))));

while abs(cov_est-cov)>10^-4
    c=c+0.0001;
    cov_est=sqrt(gamma(1+(2/c))-(gamma(1+(1/c)))^2)/(gamma(1+(1/c))*(1-
(1/(N^(1/c)))));
end

b=sqrt(variance)/sqrt(gamma(1+(2/c))-(gamma(1+(1/c)))^2);

a=m1-b*gamma(1+(1/c));

%---Expected extreme according to the weibull fit--------------------------

expec_extreme=a+b*(log(N))^(1/c)+0.57722/((c/b)*(log(N))^((c-1)/c));

fid4 = fopen('expected_extreme_weibull.txt','w');
fprintf(fid4, '%f\n' , expec_extreme);
fclose(fid4);
```

12.1.8 Extreme statistics with only global peaks

12.1.8.1 Extreme_analysis_global.m

```
% ************************************************************************
% Routine:                  extreme_analysis_global
% ------------------------------------------------------------------------
% Intension:                Calculate the extreme statistics from all N
%                           time histories
%
% Method:                   Find the average of the N largest peaks. One for
%                           each of the timeseries
%                           Find the average of the N expected extreme values
%                           according to the weibull fit of each sample
%                           Find the Gumbel extreme of the N largest peaks
%
%
% Parameter:                Description
% ------------------------------------------------------------------------
% N                         Number of sample maxima
% maxpeaks                  Vector of all N max peaks
% extremes_weibull          Vector of all N expected extremes from the
%                           Weibull fitted data
% av_maxpeaks               Average of all the max peaks
% av_weibull                Average of all the expected extremes from the
%                           weibull fitted data
% m1                        1. spectral moment of the time history
% m2                        2. spectral moment of the time history
% m3                        3. spectral moment of the time history% variance
% skewness                  Estimate of the skewness of the N maximas
% cov                       Estimate of the co variance of the N maximas
% alpha_est                 Estimate of the Gumbel parameter alpha. Describes
%                           the characteristic largest value of the initial
%                           variate X
% u_est                     Estimate of the Gumbel parameter u. Meassure of
%                           the inverse dispersion of Xn
% Gumbel_ex                 Expected extreme value of the Gumbel fitted
%                           extremes
% ------------------------------------------------------------------------
% Programmer:               Torbjørn Aakerøy Olsen
% Last changed:             26.05.2011
% ************************************************************************
clear all
clc

extreme1
extreme2
extreme3
extreme4
extreme5
extreme6
extreme7
extreme8
extreme9
extreme10
extreme11
extreme12
extreme13
```

```
extreme14
extreme15
extreme16
extreme17
extreme18
extreme19
extreme20

fid1 = fopen('maxpeak.txt');
fid2 = fopen('expected_extreme_weibull.txt');

maxpeaks = fscanf(fid1, '%f');
extremes_weibull = fscanf(fid2, '%f');

fclose(fid1);
fclose(fid2);
N=length(maxpeaks);

av_maxpeaks=sum(maxpeaks)/N;
av_weibull=sum(extremes_weibull)/N;
%calculation of central moments--------------------------------------------
m1=0;
m2=0;
m3=0;
for i=1:N
    m2=m2+((maxpeaks(i)-av_maxpeaks)^2)/N;
    m3=m3+((maxpeaks(i)-av_maxpeaks)^3)/N;
end

expec=m1;
variance=m2;
skewness=m3/(m2^(3/2));
cov=sqrt(m2)/m1;
%---Gumbel of extremes-----------------------------------------------------
alpha_est=(1.06283)/sqrt(m2);
u_est=av_maxpeaks-(0.52355/alpha_est);
Gumbel_ex=u_est+(0.57722/alpha_est);

fprintf('The mean of the maximum value of each sample
is.........................:%f\n', av_maxpeaks);
fprintf('The mean of the expected extremes (from a Weibull fit) of each
sample is:%f\n', av_weibull);
fprintf('Expected maxima of the extremes (gumbel):%f\n', Gumbel_ex);
```

12.1.8.2 Extreme1.m

```
% ***************************************************************************
% Routine:                extreme1
% ---------------------------------------------------------------------------
% Intension:              Calculate the extreme statistics of a time
%                         history
%
% Method:                 Find the largest peak of the timeseries
%                         Fit the data to the weibull distribution and find
%                         the expected extreme value
%
%
% Parameter:              Description
% ---------------------------------------------------------------------------
% time                    Vector with the time of the simulation
% amp                     Vector with the amplitude of motion of the
%                         simulation
% maxpeak                 Maximum peak of the simulation
% mean                    mean of the time history
% amp_shifted             Vector with the amplitude of motion of the
%                         simulation after extracting the mean value
% locs_zero               Locations of the zero crossings in amp
% num_intervals           Number of intervals. Between two zero crossings
%                         is an interval
% pks                     Vector with all global peaks
% m1                      1. spectral moment of the time history
% m2                      2. spectral moment of the time history
% m3                      3. spectral moment of the time history
% N                       Number of samples
% expec                   Estimation of the expected value of the time
%                         history
% variance                Estimation of the variance of the time history
% skewness                Estimation of the skewness of the time history
% cov                     Estimation of the co variance of the time
%                         history
% a                       Estimate of the Weibull location parameter
% b                       Estimate of the Weibull scaling parameter
% c                       Estimate of the Weibull shape factor
% expec_extreme           Expected extreme value of the weibull fitted data
% ---------------------------------------------------------------------------
% Programmer:             Torbjørn Aakerøy Olsen
% Last changed:           26.05.2011
% ***************************************************************************

clear all
clc

fid1 = fopen('timeex.txt');
time = fscanf(fid1, '%f');
fclose(fid1);

fid2 = fopen('sample1.txt');
amp = fscanf(fid2, '%f');
fclose(fid2);

maxpeak=max(amp);
```

```
fid3 = fopen('maxpeak.txt','w');
fprintf(fid3, '%f\n' , maxpeak);
fclose(fid3);

mean=0;
for i=1:(length(amp))
    mean=mean+(amp(i)/length(amp));
end


amp_shifted=amp-mean;

teller=0;
for i = 2:length(amp_shifted)-1
    teller=teller+1;
    if sign(amp_shifted(i))~=sign(amp_shifted(i+1))
        locs_zero(teller)=i;
    end
end

index = find(locs_zero);
locs_zero=locs_zero(index);

if amp(locs_zero(1))<=amp(locs_zero(1)-1)
    locs_zero(1)=[];
end

if amp(locs_zero(length(locs_zero)))>=amp(locs_zero(length(locs_zero))-1)
    locs_zero(length(locs_zero))=[];
end


num_intervals=length(locs_zero)-1;

for i = 1:2:num_intervals

   amp2=amp;
   time2=time;
   xx=locs_zero(i);
   yy=locs_zero(i+1);

   amp2((yy+1):end)=[];
   amp2(1:(xx-1))=[];
   time2((yy+1):end)=[];
   time2(1:(xx-1))=[];

   pks(i)=max(amp2);

   for j=1:length(amp2)
       if amp2(j)==pks(i)
           time_pks(i)=time2(j);
       end
   end
```

```
    t_pks(i)=(time_pks(i)/0.2)+1;

end

index = find(t_pks);
t_pks=t_pks(index);

index = find(pks);
pks=pks(index);

t_pks=round(t_pks);

% calculating the central moments of the sample----------------------------
N=length(pks);

m1=0;
m2=0;
m3=0;

for i=1:N
    m1=m1+(pks(i)-mean)/N;
    m2=m2+((pks(i)-mean)^2)/N;
    m3=m3+((pks(i)-mean)^3)/N;
end

expec=m1;
variance=m2;
skewness=m3/(m2^(3/2));
cov=sqrt(m2)/m1;

%------- Estimate the weibull parameters ----------------------------------
c=0.1;
cov_est=sqrt(gamma(1+(2/c))-(gamma(1+(1/c)))^2)/(gamma(1+(1/c))*(1-
(1/(N^(1/c)))));

while abs(cov_est-cov)>10^-4
    c=c+0.0001;
    cov_est=sqrt(gamma(1+(2/c))-(gamma(1+(1/c)))^2)/(gamma(1+(1/c))*(1-
(1/(N^(1/c)))));
end

b=sqrt(variance)/sqrt(gamma(1+(2/c))-(gamma(1+(1/c)))^2);

a=m1-b*gamma(1+(1/c));

%---Expected extreme according to the weibull fit--------------------------

expec_extreme=a+b*(log(N))^(1/c)+0.57722/((c/b)*(log(N))^((c-1)/c));

fid4 = fopen('expected_extreme_weibull.txt','w');
fprintf(fid4, '%f\n' , expec_extreme);
fclose(fid4);
```

12.2 System description

12.2.1 Tow parameters

Pennant wire connection points relative to vessel COG [m] (Vessel system of coordinates defined in Figur 3-1 and Figur 3-2)	(X, Y, Z) = (-1.473, 1.382 and -0.382, 4,0)
Pennant wire length, two off [m]	30
Sling wire length, four off [m]	24
Design tow velocity [m/s] / [knot] (*)	1.5 / 3
Design current velocity towards tow direction [knot] (*)	1
Tow distance [nm]	105
Planned tow operation period [hrs]	35

(*) The tow velocity must be adjusted according to current velocity.

Table 11 Tow parameters. Reference [9]

12.2.2 Vessel parameters

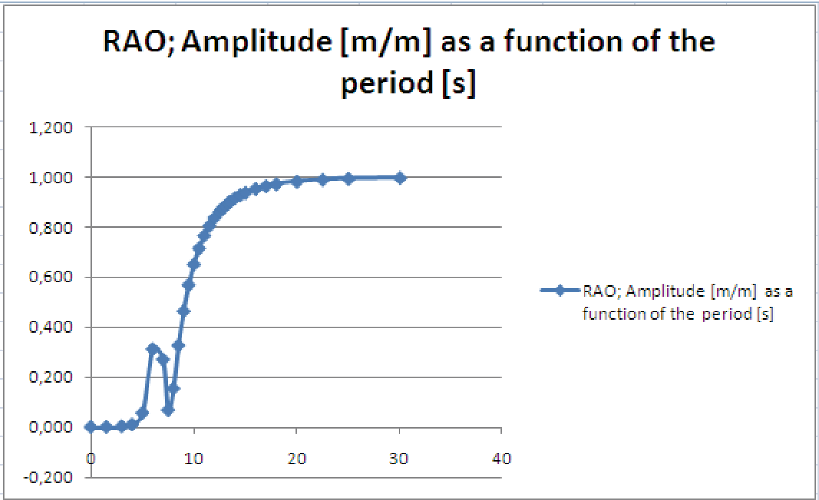
LOA [m]	96.37
LPP [m]	77.91
Breadth [m]	24
Draught [m]	8.5
Displacement [Te]	9161.9
KG [m]	10.23

Table 12 Tow and installation vessel parameters. Reference [9]

			T=7.56 m	T=8.50 m
Length over all	L_{oa}	(m)	96.37	96.37
Length between perpendiculars	L_{pp}	(m)	77.91	77.91
Breadth, maximum	B_{max}	(m)	24.00	24.00
Breadth on WL	B_{WL}	(m)	23.85	23.52
Displacement	Δ	(tonnes)	7542.0	9161.9
Vertical position of COG, above base	KG	(m)	8.94	10.23
Horizontal position of COG, relative to the midship ("-" i.e. forward)	LCG	(m)	-1.25	-1.04
Transverse metacentric height	GM_T	(m)	3.64	2.37
Longitudinal metacentric height	GM_L	(m)	92.78	95.03

Table 13 Main particulars of MSV Botnica. Reference [8]

Period	Heave		
	Ampl.	Phase	Frequency
(s)	m/m	deg	rad/s
0	0,000	0,000	0,000
1,5	0,000	0,000	4,189
3	0,003	249,312	2,094
4	0,010	270,754	1,571
5	0,056	62,176	1,257
6	0,312	259,332	1,047
7	0,270	219,505	0,898
7,5	0,068	181,991	0,838
8	0,154	35,764	0,785
8,5	0,327	22,349	0,739
9	0,463	16,077	0,698
9,5	0,569	12,247	0,661
10	0,651	9,681	0,628
10,5	0,715	7,731	0,598
11	0,766	6,295	0,571
11,5	0,806	5,228	0,546
12	0,838	4,408	0,524
12,5	0,864	3,720	0,503
12,8	0,875	3,479	0,491
13	0,885	3,147	0,483
13,3	0,894	2,918	0,472
13,5	0,902	2,794	0,465
14	0,916	2,349	0,449
14,5	0,929	2,115	0,433
15	0,938	1,787	0,419
16	0,953	1,446	0,393
17	0,964	1,116	0,370
18	0,972	0,893	0,349
20	0,983	0,565	0,314
22,5	0,990	0,345	0,279
25	0,994	0,231	0,251
30	0,997	0,118	0,209



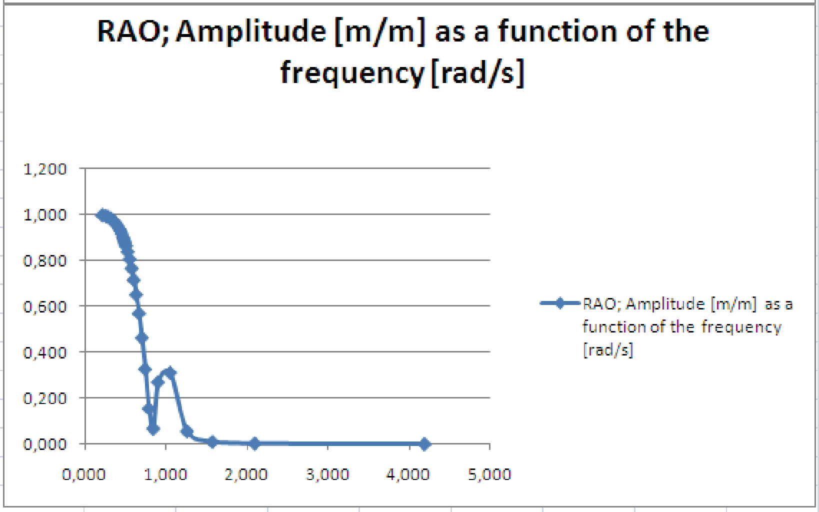


Table 14 Transfer function of MSV Botnica

12.2.3 ITS parameters

Mass without hatches and guide posts [Te]	260
Submerged weight [kN]	2216

Table 15 ITS parameters. Reference [9]

Mass [Te]	69
Steel volume [m^3]	8.7
Moment of inertia, x direction [Tem^2]	553.5
Moment of inertia, y direction [Tem^2]	1755.6
Moment of inertia, z direction [Tem^2]	2057.5
Added mass coefficient, x direction [-]	8.7
Added mass coefficient, y direction [-]	9.2
Added mass coefficient, z direction [-]	7.7
Drag coefficient, x direction [-]	1.192
Drag coefficient, y direction [-]	1.508
Drag coefficient, z direction [-]*	2.0
Drag area, x direction [m^2]	38.8
Drag area, y direction [m^2]	38.5
Drag area, z direction [m^2]	46.5

*Recommended by DNV for compounded units.

Table 16 ITS parameters, centre unit. Reference [9]

Total mass of four buckets [Te]	76.5
Mass [kg/m]	4.069
OD [m]	6
Equivalent ID [m]	5.945
Equivalent WT [mm]	27.6
Height [m]	4.7
Displaced mass including content [Te] (one bucket) (modelled and basis for added mass coeff)	136.2
Axial added mass coefficient [-], ref /8/ (Entrapped water included by water filled elements)	0,85
Normal added mass coefficient [-], ref /8/ (Entrapped water included by water filled elements)	0.57
Axial drag coefficient [-], ref /7/ (i.e. skin friction)	0.64
Normal drag coefficient [-], ref DNV Environmental condition and environmental loads, Classification notes 30.5, March 2000	Dependent on Reynolds number; [0.3 – 1.2]

Table 17 ITS parameters, foundation bucket. Reference [9]

	Inner	Outer
Total mass [Te]	37.9	2.4
Mass [kg/m]	1.058	0.155
Length [m]	8.96	3.83
Outer diameter, OD [mm]	1067	508
Equivalent ID [mm]	983	483
Equivalent WT [mm]	41.8	12.7
Axial added mass coefficient [-], ref /8/	0.0	0.0
Normal added mass coefficient [-], ref /8/	1.0	1.0
Axial drag coefficient [-], i.e skin friciton, ref /7/	0.0	0.0
Normal drag coefficient [-], ref DNV Environmental condition and environmental loads, Classification notes 30.5, March 2000	Dependent on Reynolds number; [0.3 – 1.2]	

Table 18 ITS parameters, bottom pipes. Reference [9]

	Vertical	Sloped	Horizontal
Total mass, four of [Te]	11.6	6.9	1.8
Mass [kg/m]	374.0	155.1	155.1
Length [m]	7.73	11.2	2.95
Outer diameter, OD [mm]	813	508	508
Inner diameter, ID [mm]	775	483	483
Wall thickness WT [mm]	19.1	12.7	12.7
Equivalent axial added mass coefficient [-]	0	0	0.261
Equivalent y-direction added mass coeff [-]	1	1	1.163
Equivalent x-direction added mass coeff [-]	1	1	13.121
Equivalent axial drag coefficient [-]	0	0	0
Equivalent y-direction drag coeff [-]	DnV	DnV	3.916
Equivalent x-direction drag coeff [-]	DnV	DnV	1.2

Table 19 ITS parameters, legs. Reference [9]

	Longitudinal direction	Transverse direction
Length [m]	18,6	15,1
Mass [kg/m]	803	
OD [mm]	914	
Equivalent ID [mm]	840	
Equivalent WT [mm]	37	
Axial added mass coefficient [-], ref /8/	0.0	
Normal added mass coefficient [-], ref /8/	1.0	
Axial drag coefficient [-], i.e skin friction, ref /7/	0.0	
Normal drag coefficient [-], ref DNV Environmental condition and environmental loads, Classification notes 30.5, March 2000	Dependent on Reynolds number; [0.3 – 1.2]	

Table 20 ITS parameters, top frame. Reference [9]

Added mass of the ITS		
name (number of units)	**Added mass in heave per unit [Te]**	**Total added mass in heave [Te]**
Center unit (1)	68,47	68,47
Foundation bucket (4)	250,78	1003,12
Bottom pipes (4)	17,113	68,452
Top frame (1)	45,32	45,32
Legs (sum of vertical, horizontal and sloped) (4)	18,237	72,948
	Total added mass in heave	**1258,31**

Table 21 Added mass of the ITS

Drag in x-direction of the ITS			
	Drag area x-direction per unit [m^2]	Total drag area x-direction [m^2]	Drag coefficient x-direction [-]
Center unit (1)	38,790	38,790	1,192
Foundation bucket (4)	28,200	112,800	1,200
Inner bottom pipes (4)	7,707	30,829	1,200
Outer bottom pipes (4)	1,569	6,274	1,200
Top frame (2)	13,801	27,603	1,200
Horizontal leg (4)	1,208	4,832	3,916
Vertical leg (4)	6,284	25,138	1,200
Sloped leg (4)	5,285	21,141	1,200
	Sum drag area x-direction [m^2]	267,406	
	Equivalent drag coefficient x-direction [-]	1,248	
	Projected length [m]	diameter [m]	
Foundation bucket (4)	4,700	6,000	
Inner bottom pipes (4)	7,223	1,067	
Outer bottom pipes (4)	3,088	0,508	
Top frame (2)	15,100	0,914	
Horizontal leg (4)	2,378	0,508	
Vertical leg (4)	7,730	0,813	
Sloped leg (4)	10,404	0,508	

Table 22 Drag in x-direction of the ITS

Drag in z-direction of the ITS			
	Drag area z-direction per unit [m^2]	Total drag area x-direction [m^2]	Drag coefficient z-direction [-]
Center unit (1)	46,500	46,500	2,000
Foundation bucket (4)	28,274	113,097	0,640
Inner bottom pipes (4)	9,560	38,241	1,200
Outer bottom pipes (4)	1,946	7,783	1,200
Top frame short side (2)	13,801	27,603	1,200
Top frame long side (2)	17,000	34,001	1,200
Horizontal leg (4)	1,499	5,994	3,916
Sloped leg (4)	3,561	14,244	1,200
	Sum drag area z-direction [m^2]	287,463	
	Equivalent drag coefficient z-direction [-]	1,166	
	Projected length [m]	diameter [m]	
Inner bottom pipes (4)	8,960	1,067	
Outer bottom pipes (4)	3,830	0,508	
Top frame short side (2)	15,100	0,914	
Top frame long side (2)	18,600	0,914	
Horizontal leg (4)	2,950	0,508	
Sloped leg (4)	7,010	0,508	

Table 23 Drag in z-direction of the ITS

12.2.4 Rigging parameters

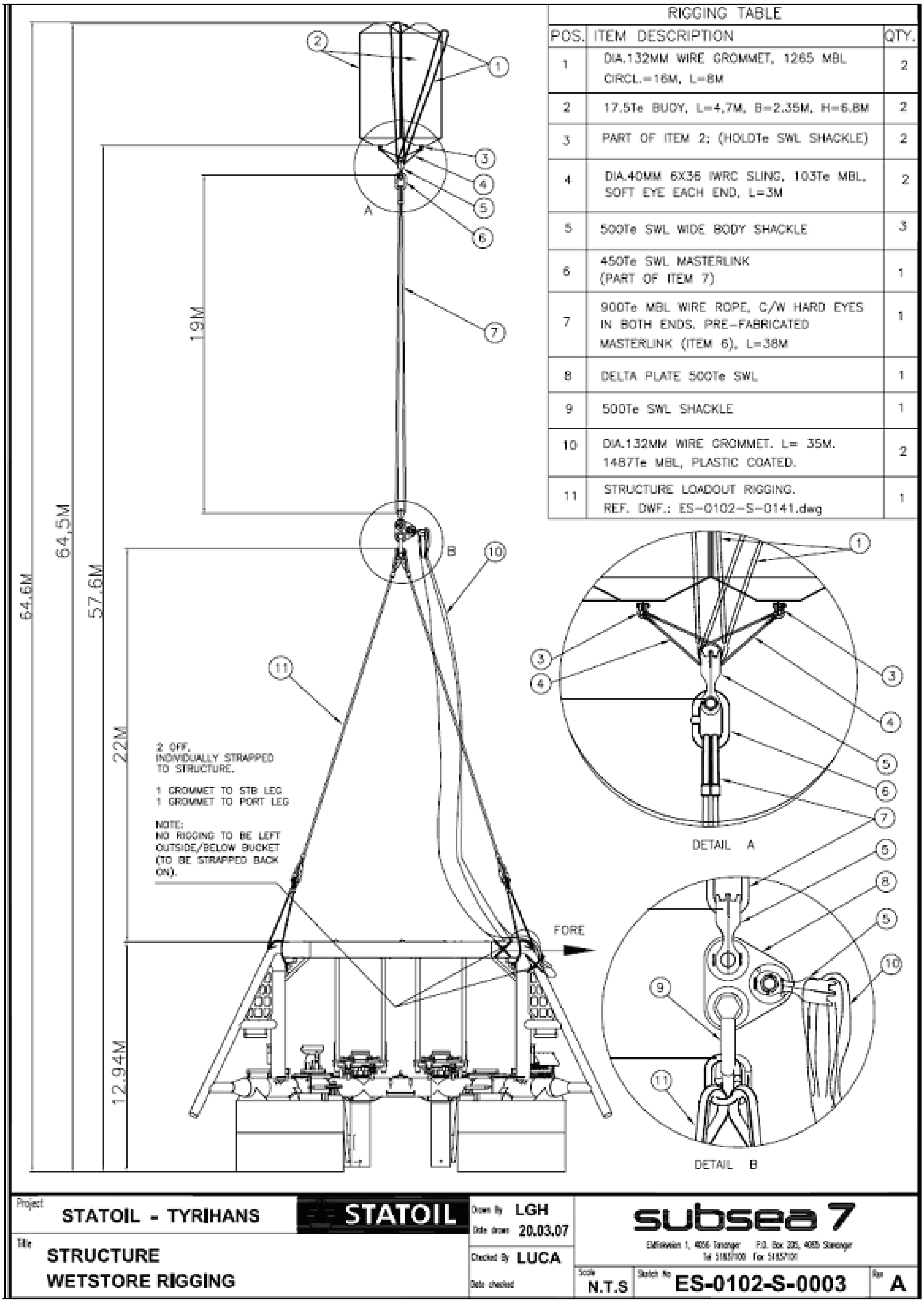


Figure 72 Rigging recommendations. Reference [19]

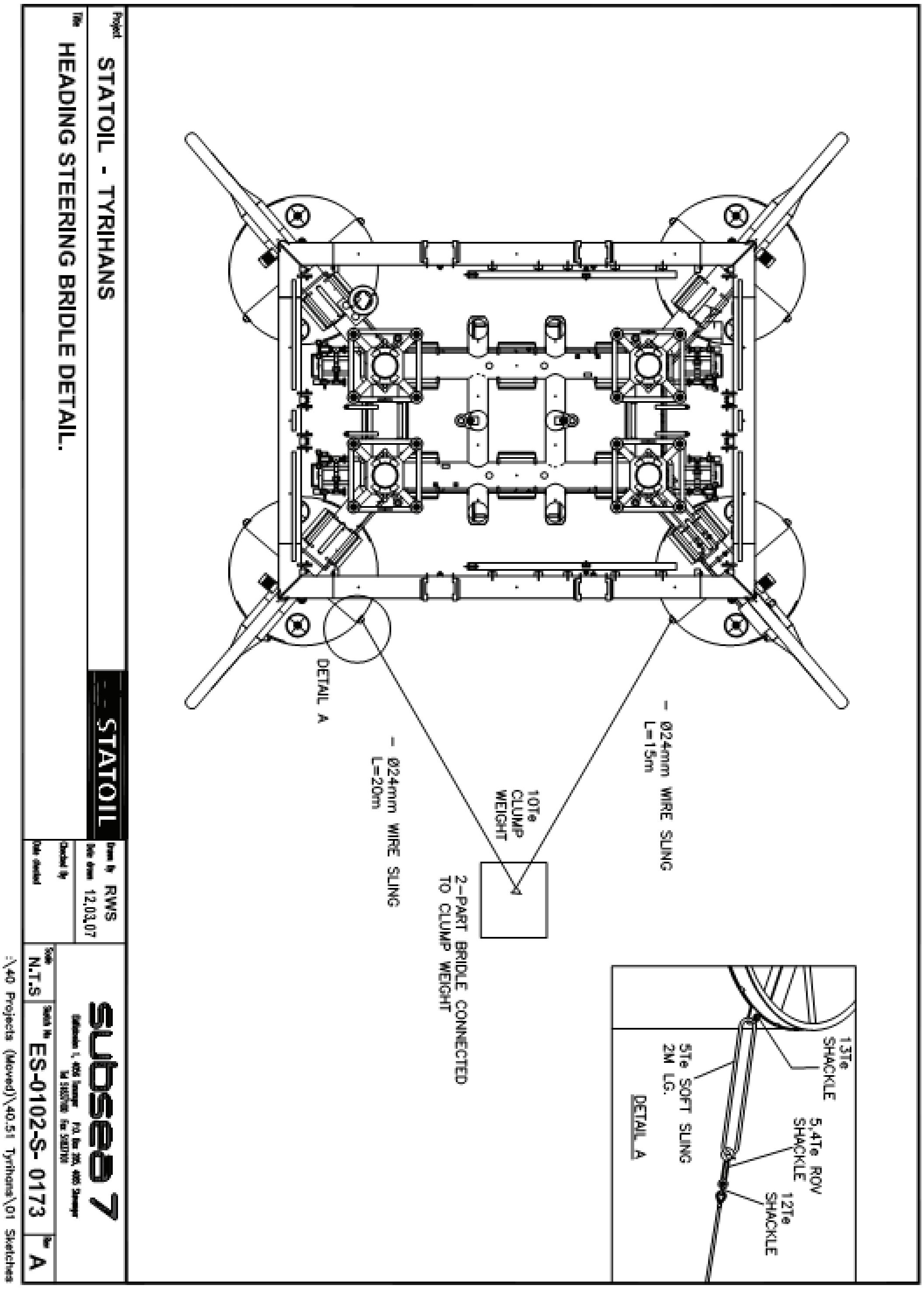


Figure 73 Steering wire configuration. Reference [18]

	Pennants (2 off)	4 slings bridle
Diameter [mm]	90*	83
Mass [kg/m]	67*	29
Axial stiffness [kN]	551500*	235000
MBL [kN]	11616*	4385
Axial added mass coefficient [-], ref /8/	0.0	
Normal added mass coefficient [-], ref /8/	1.0	
Axial drag coefficient [-] (i.e. skin friction), ref /7/	0.0	
Normal drag coefficient [-], ref DNV Environmental condition and environmental loads, Classification notes 30.5, March 2000	Dependent on Reynolds number; [0.3 – 1.2]	

* Grommets – mass, stiffness and MBL are given as per 2xnominal diameter.

Table 24 Wire parameters. Reference [9]

Connection point relative to vessel COG [m] (Vessel system of coordinates defined in Figur 3-1 and Figur 3-2)	(X, Y, Z) = (50.8, 0, 8)
Length [m]	90
Length between Botnica and clump weight [m]	75
Length between clump weight and ISS [m]	15
Diameter [mm]	24
Mass [kg/m]	2.4
Axial stiffness [kN]	19.65E3
MBL [kN]	363
Clump mass [kg]	900
Clump submerged weight [kN]	7.67

Table 25 Steering wire parameters. Reference [9]

All rigging except the delta plate and the sheave is included in the analysis. The additional weight is:

Additional rigging weight	
Sheave mass, [Te]	18
Delta plate mass, [Te]	1.5
Total submerged weight, [Te]	17

Table 26 Information about extra rigging weight. Reference [19]

name (number of units)	drag coefficient	total drag area [m^2]	cross section area [m^2]	Axial stifness [N] (EA)	E-modulus [Pa]
Pennant (2)	1,2	5,4	0,006361725	551500000	86690322089
sling (4)	1,2	7,3704	0,005410608	235000000	43433196837

Equivalent length [m]	51,76242099	Found from the parameters given in reference [9]
Equivalent mass [kg]	6804	Total mass/equivalent length
Equialent mass pr unit length [kg/m^3]	131,4467111	Equivalent mass/equivalent length
Equivalent submerged weight [kg]	5915,579618	Equivalent volume*(density of steel - density of sea water)
Density steel [kg/m^3]	7850	
Density seawater [kg/m^3]	1025	
Equivalent volume [m^3]	0,866751592	Equivalent mass/density steel
Equivalent diameter [m]	0,1460142	From the formula: volume=PI*radius^2*length
Equivalent Cross section area [m^2]	0,016744804	From the formula: Area=PI*radius^2
Equivalent drag area [m^2]	7,558048487	Equivalent length * equivalent diameter
Equivalent drag coefficient	2,027571009	Same procedure as equivalent drag coefficient for the template
Equivalent axial stiffness [N]	507498776,3	Described in text
Equivalent E-modulus [Pa]	30307836229	Equivalent axial stiffness/equivalent cross section area

Table 27 Equivalent wire

12.2.5 Environment

Wave spectrum		JONSWAP
Significant wave height	, Hs	2,9 [m]
Zero crossing periods	, Tz	6 [s]
Peak period	, Tp	7,5104 [s]
Gamma	, γ	4,5495 [-]
Current	,U	1,5 [m/s]
Density of sea water	, ρ	1025 [kg/m^3]

Table 28 Environmental properties

12.3 1-DOF results

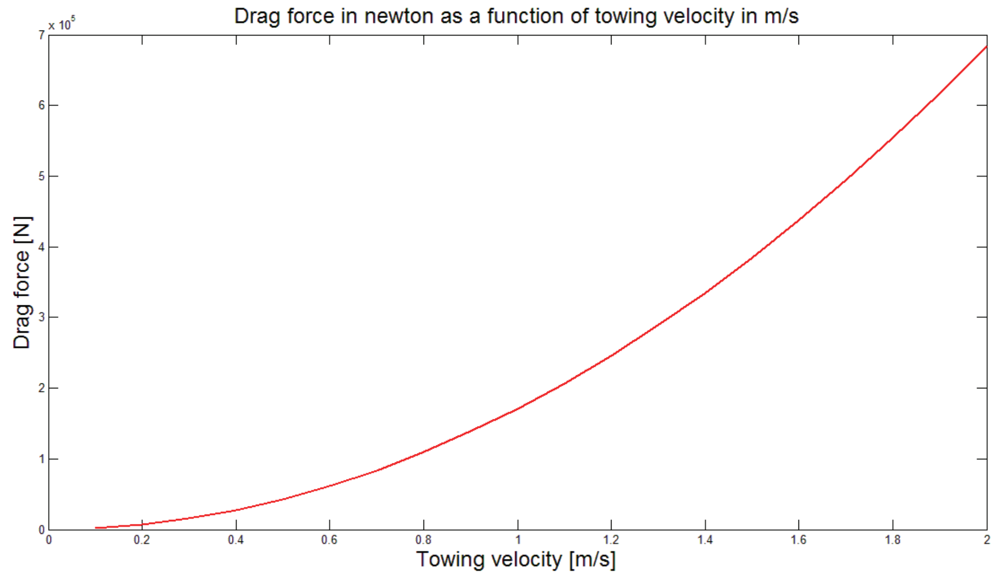


Figure 74 Drag force in Newton as a function of the towing velocity in meters per second

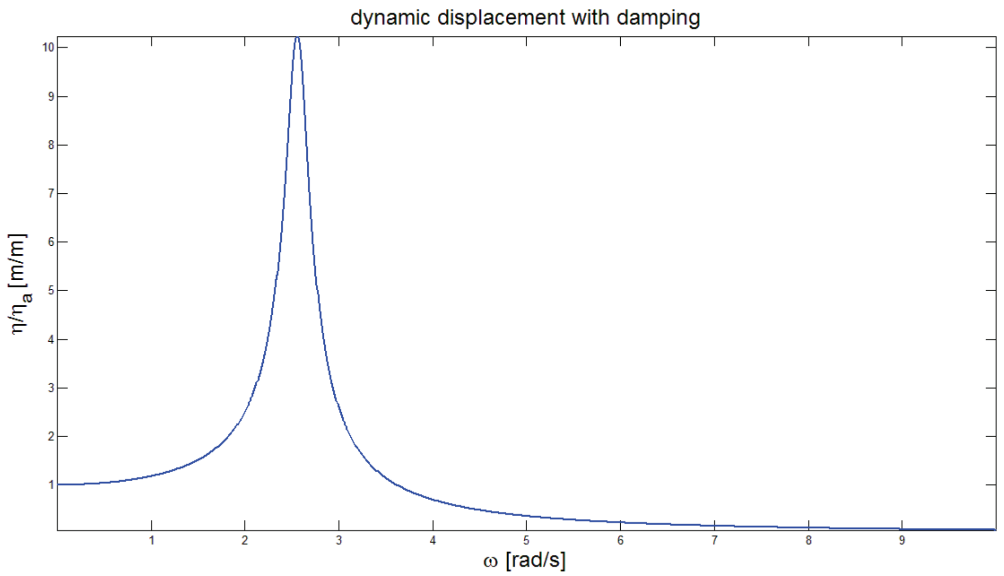


Figure 75 Dynamic displacement of the wire as a function of the frequency

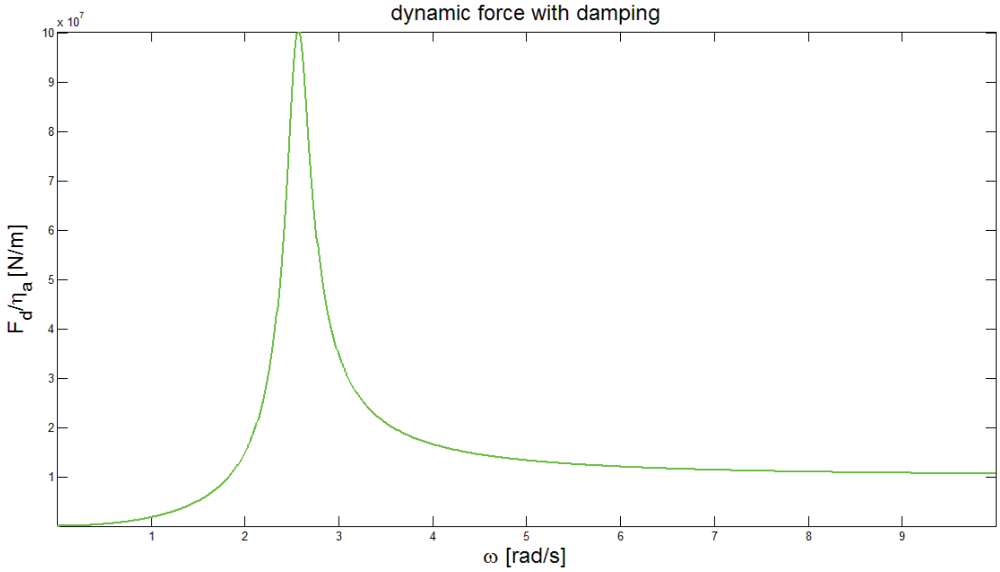


Figure 76 Dynamic force in the wire as a function of the frequency

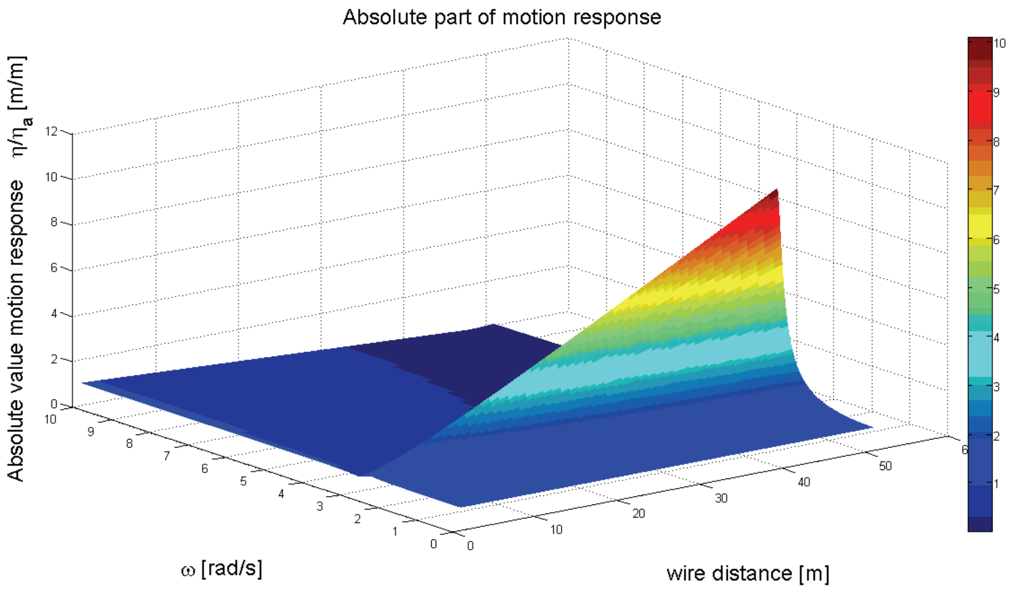


Figure 77 Absolute part of the motion response in the wire as a function of the frequency. Varying wire distance.

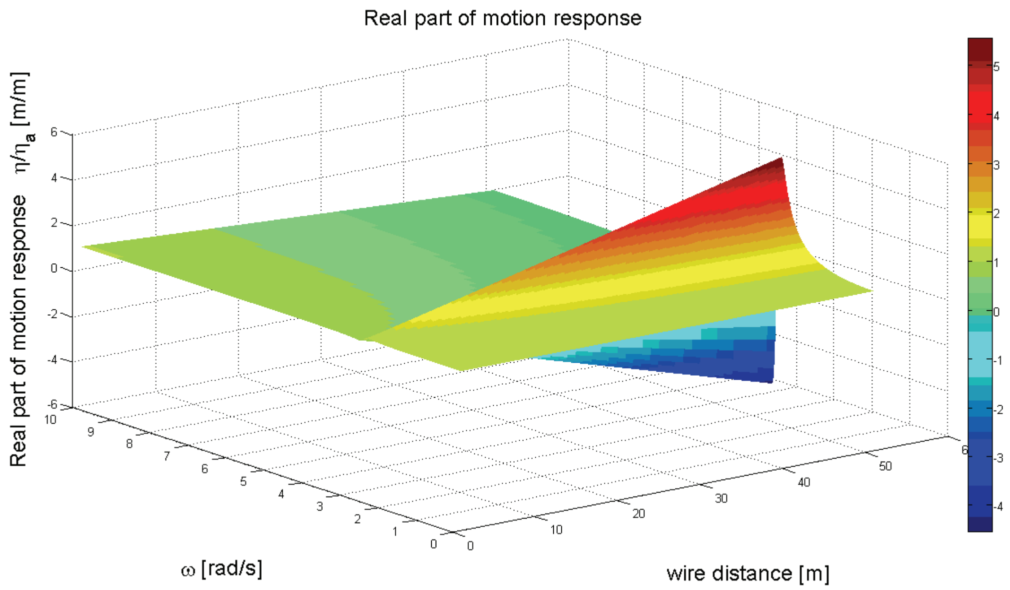


Figure 78 Real part of the motion response in the wire as a function of the frequency. Varying wire distance.

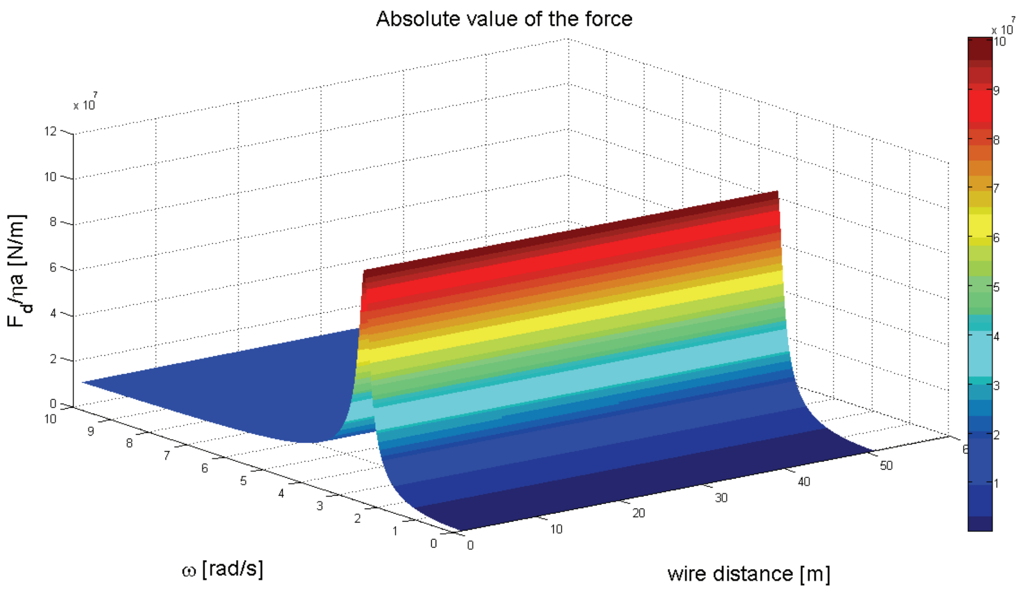


Figure 79 Absolute value of the force in the wire as a function of the frequency. Varying wire distance.

12.4 Surface elevation, vessel- and template motion in regular waves with an artificial damping

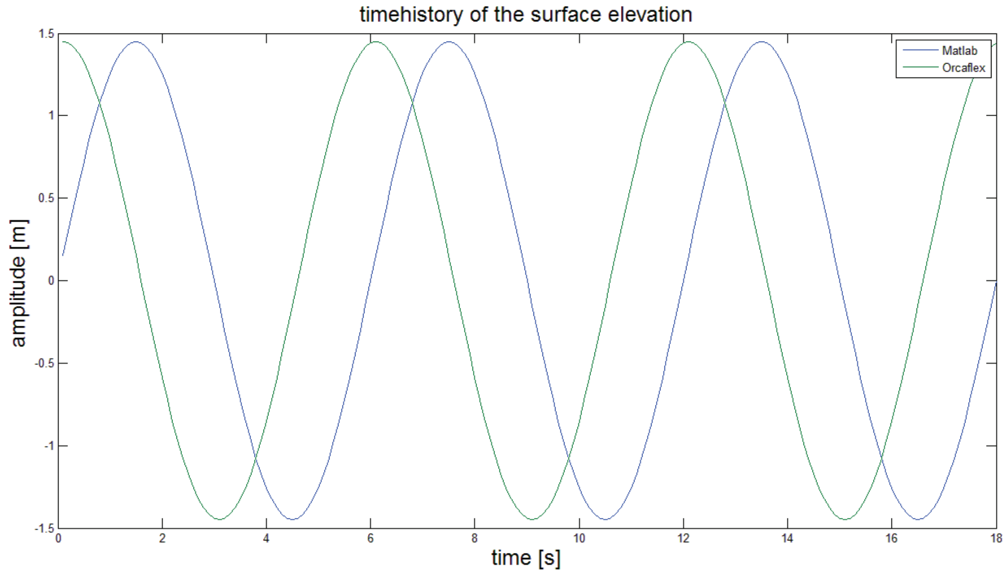


Figure 80 Time history of the surface elevation after introducing an artificial damping in the 1-DOF program

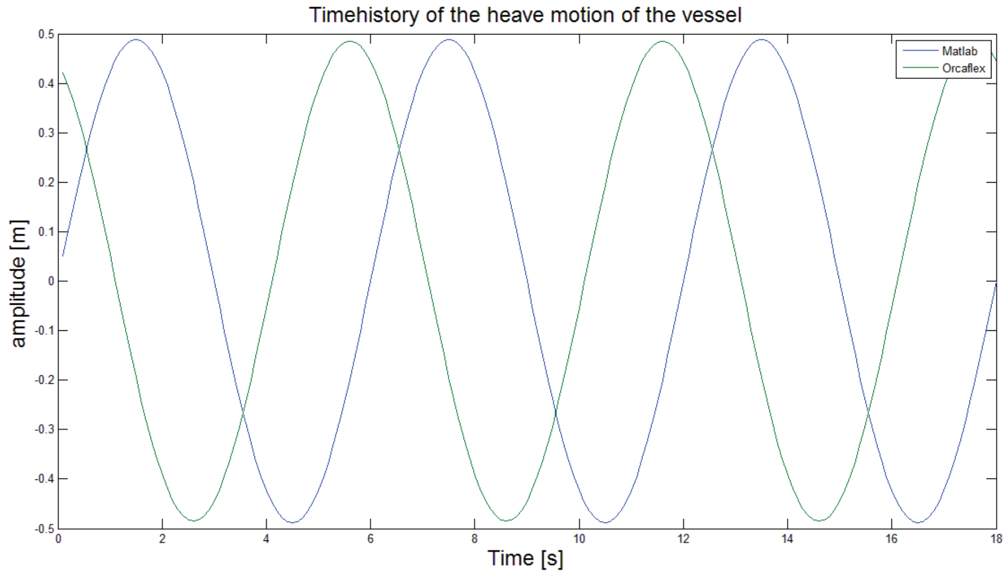


Figure 81 Time history of the vessel motion in heave after introducing an artificial damping in the 1-DOF program

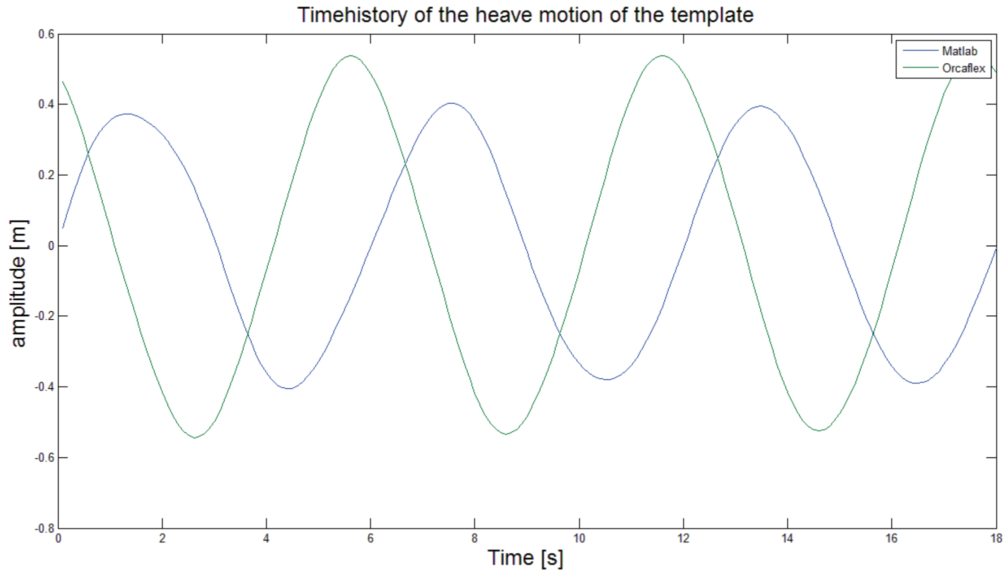


Figure 82 Time history of the template motion in heave after introducing an artificial damping in the 1-DOF program

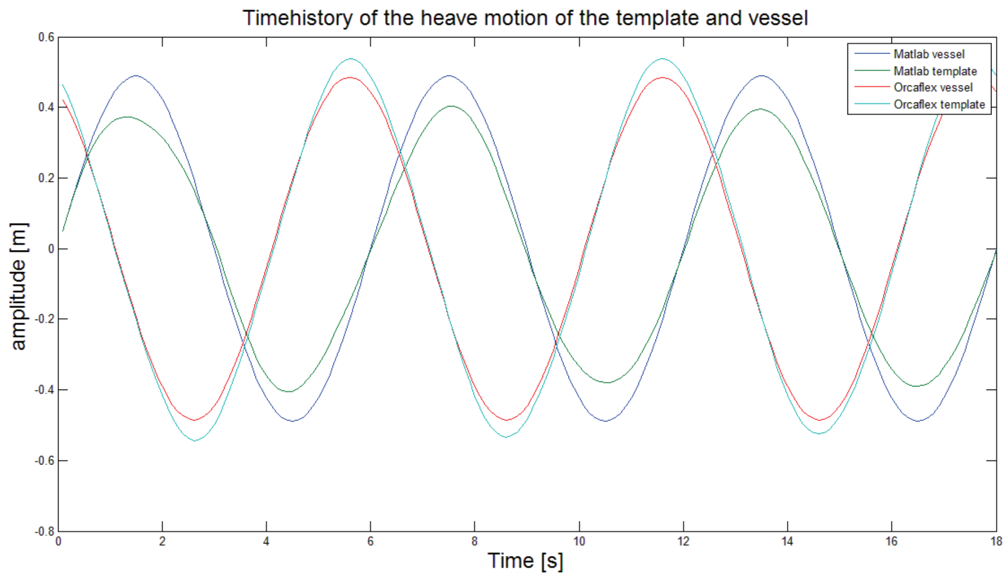


Figure 83 Time history of both the vessel and template motion in heave after introducing an artificial damping in the 1-DOF program

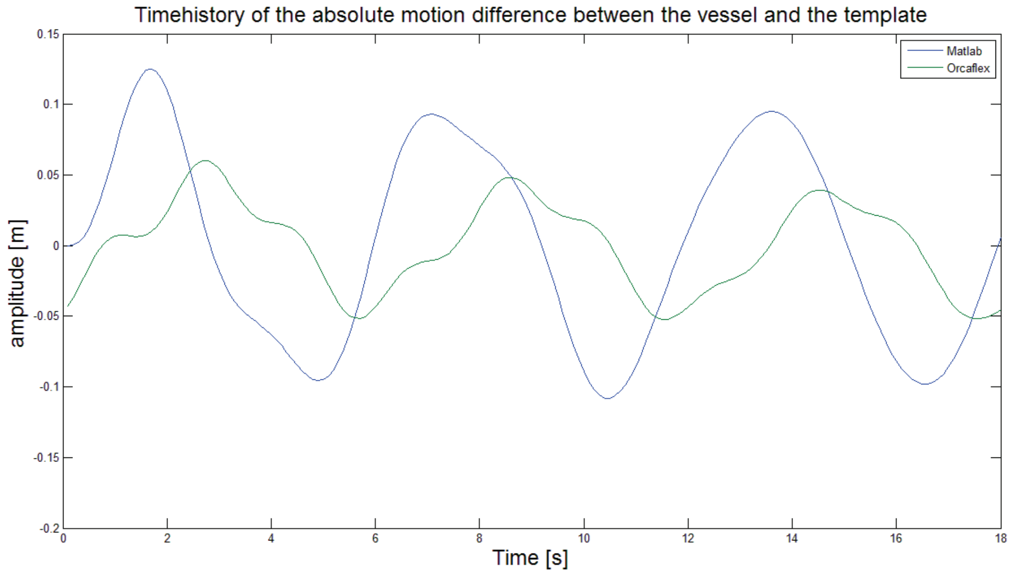


Figure 84 Time history of the motion difference between the vessel and the template after introducing an artificial damping in the 1-DOF program

12.5 FFT plots of the surface elevation, vessel- and template motion

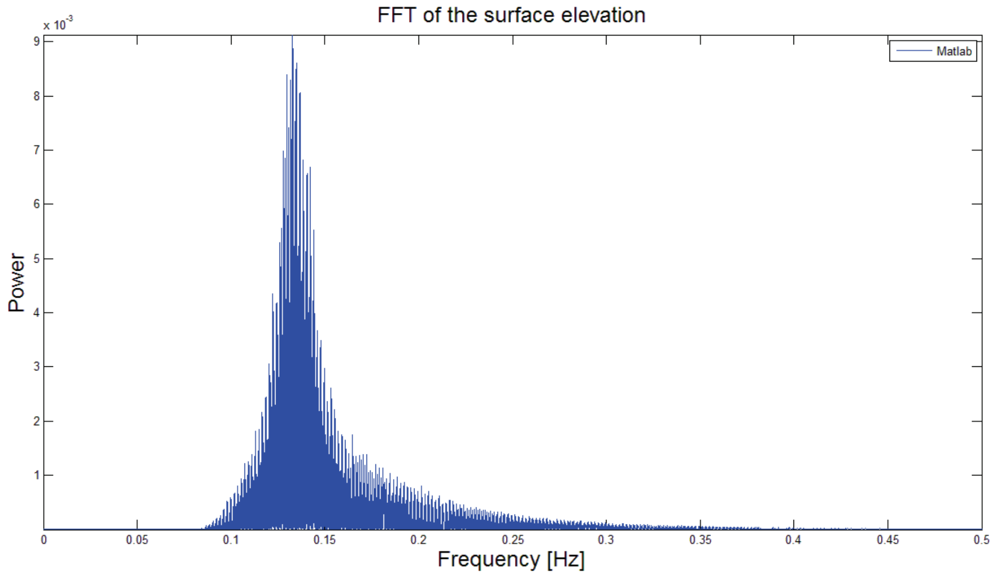


Figure 85 FFT of the surface elevation in the 3 hour 1-DOF analysis.

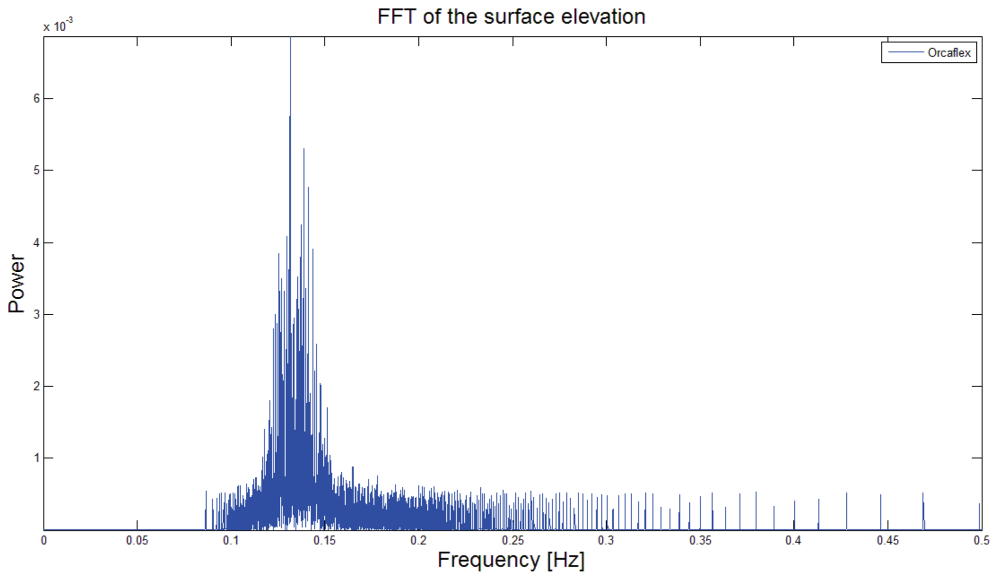


Figure 86 FFT of the surface elevation in the 3 hour multiple-DOF analysis.

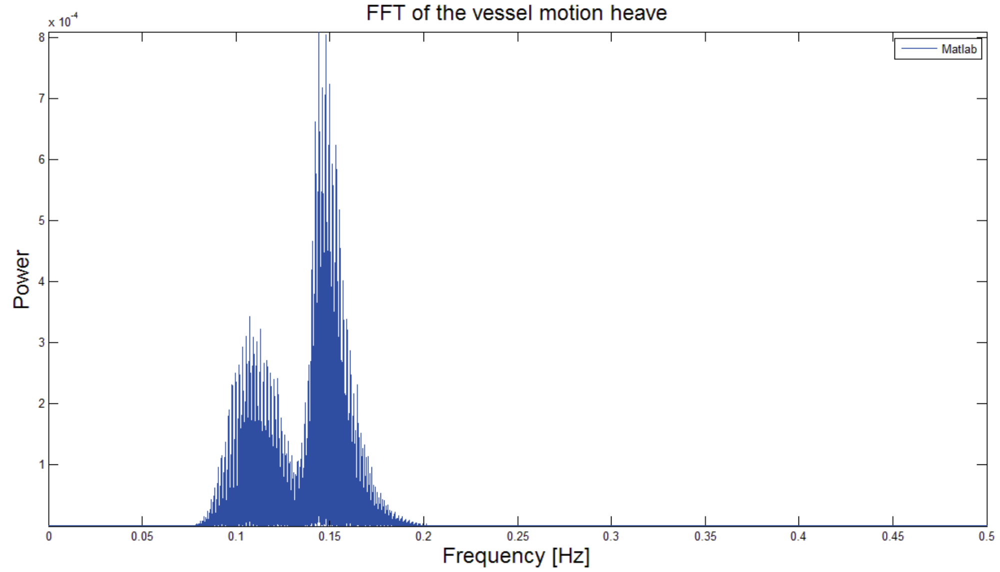


Figure 87 FFT of the vessel motion in heave in the 3 hour 1-DOF anlaysis.

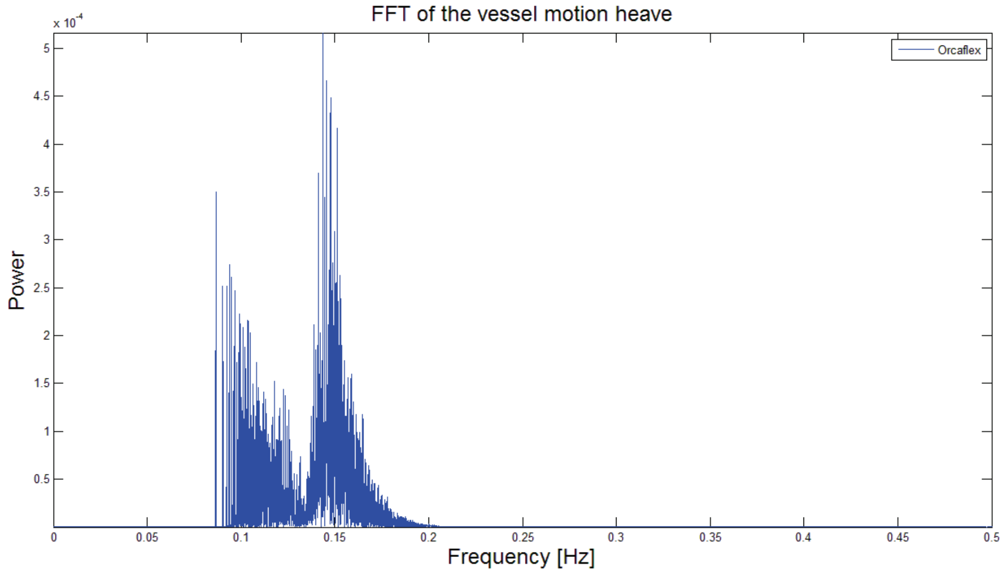


Figure 88 FFT of the vessel motion in heave in the 3 hour multiple-DOF analysis.

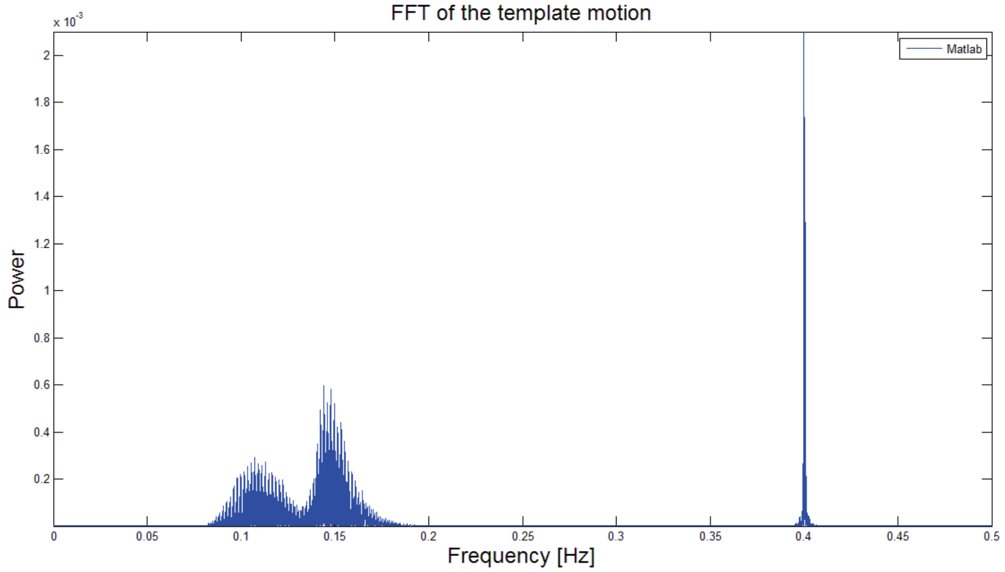


Figure 89 FFT of the vessel motion in heave in the 3 hour 1-DOF analysis.

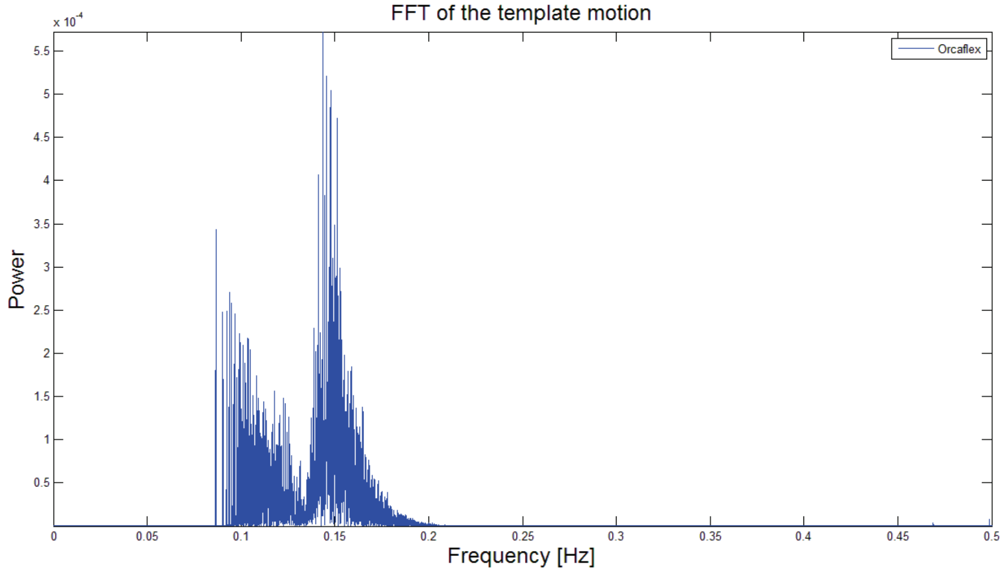


Figure 90 FFT of the vessel motion in heave in the 3 hour multiple-DOF analysis.

12.6 FFT plots of the surface elevation, vessel- and template motion with an artificial damping.

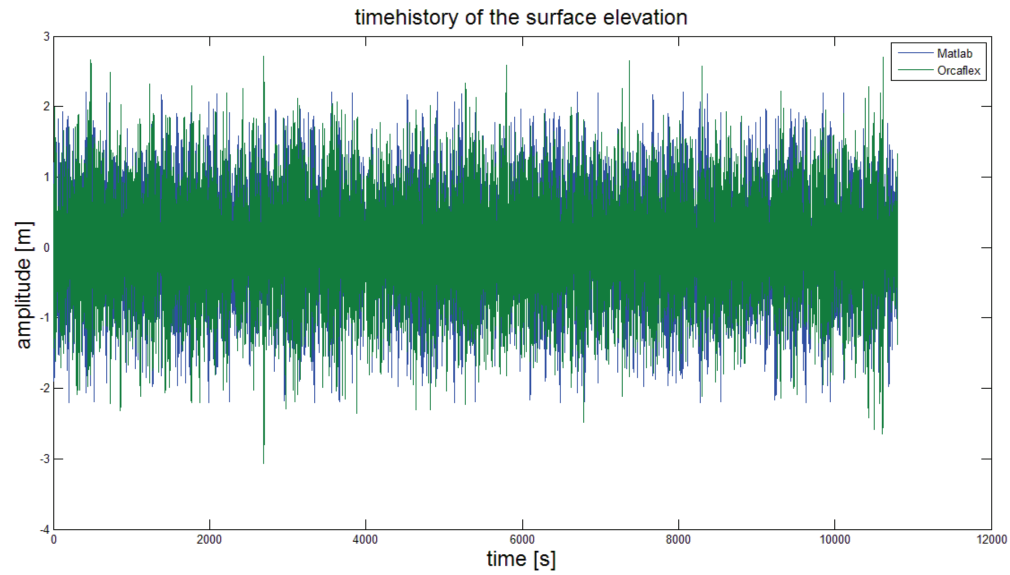


Figure 91 Time history of the surface elevation after introducing an artificial damping in the 1-DOF analysis.

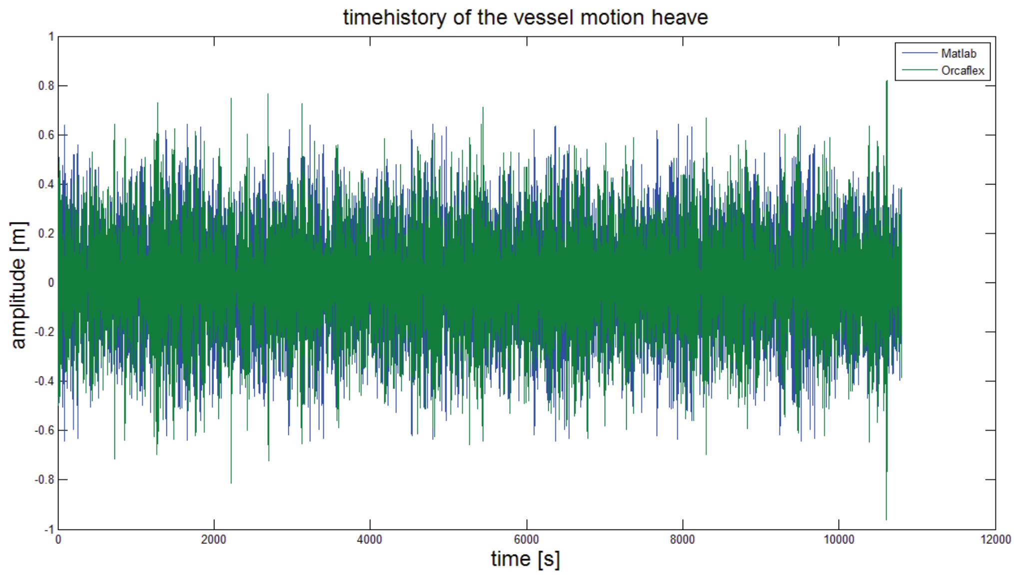


Figure 92 Time history of the vessel motion in heave after introducing an artificial damping in the 1-DOF analysis.

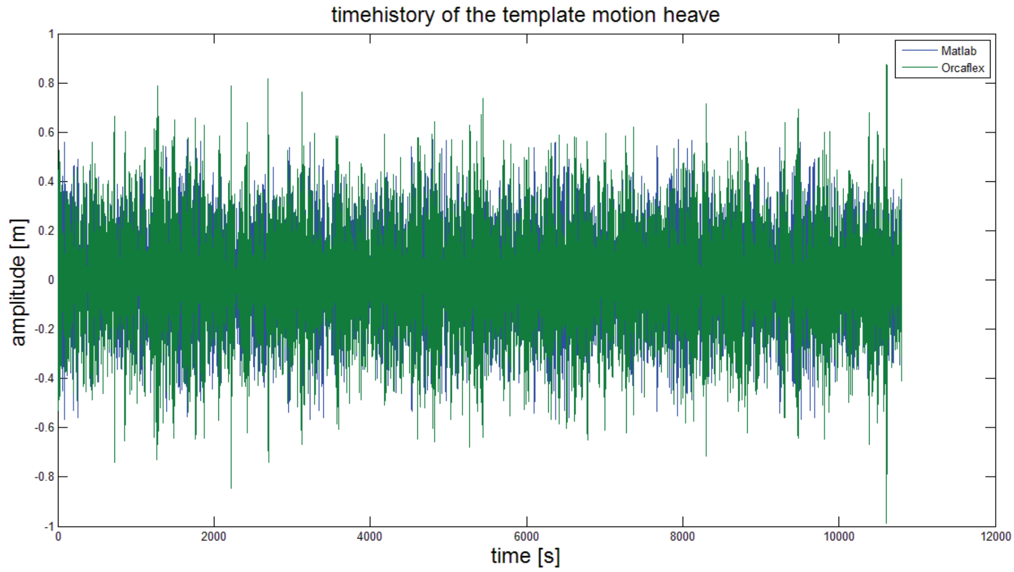


Figure 93 Time history of the template motion in heave after introducing an artificial damping in the 1-DOF analysis.

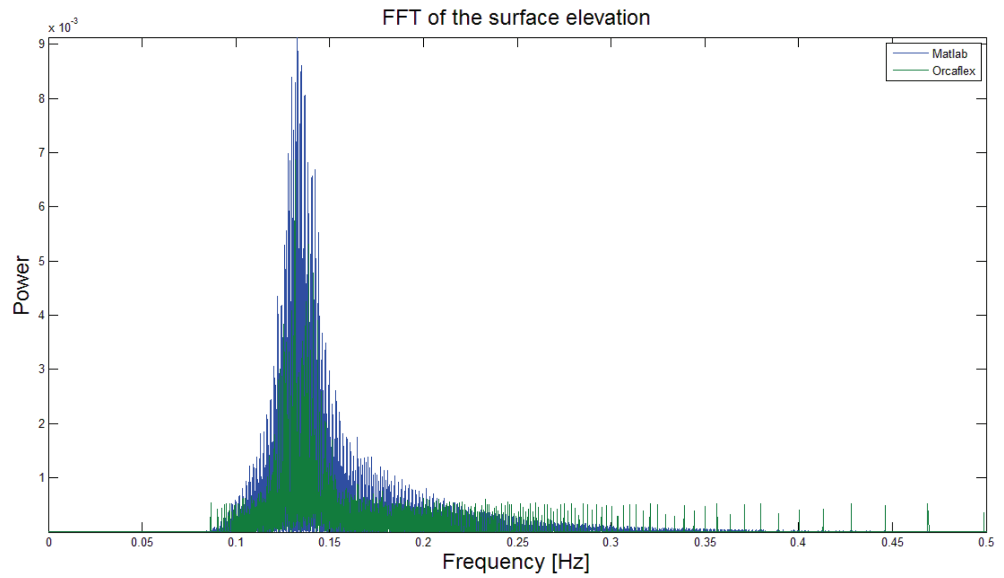


Figure 94 FFT of the surface elevation in after introducing an artificial damping in the 1-DOF analysis.

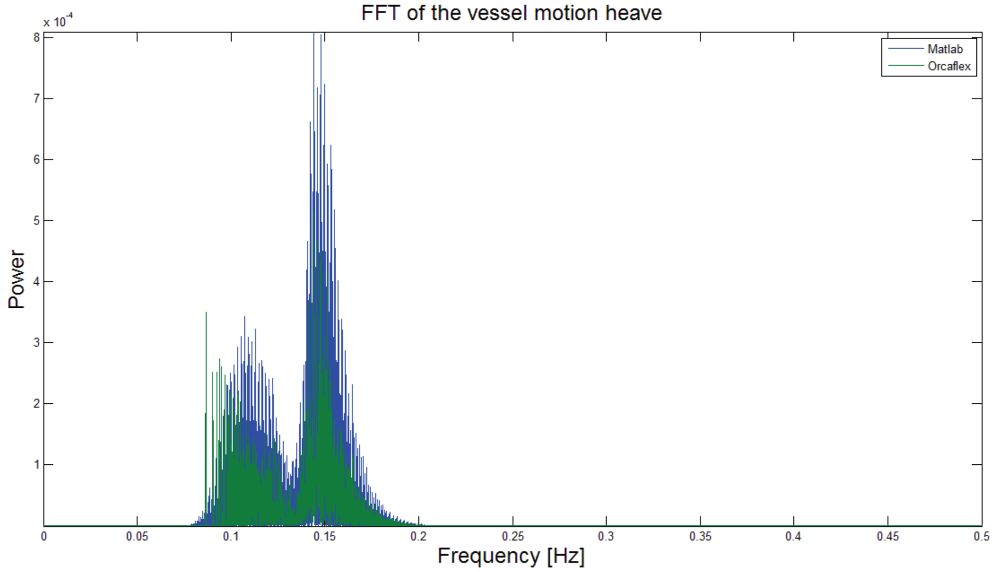


Figure 95 FFT of the vessel motion in heave after introducing an artificial damping in the 1-DOF analysis.

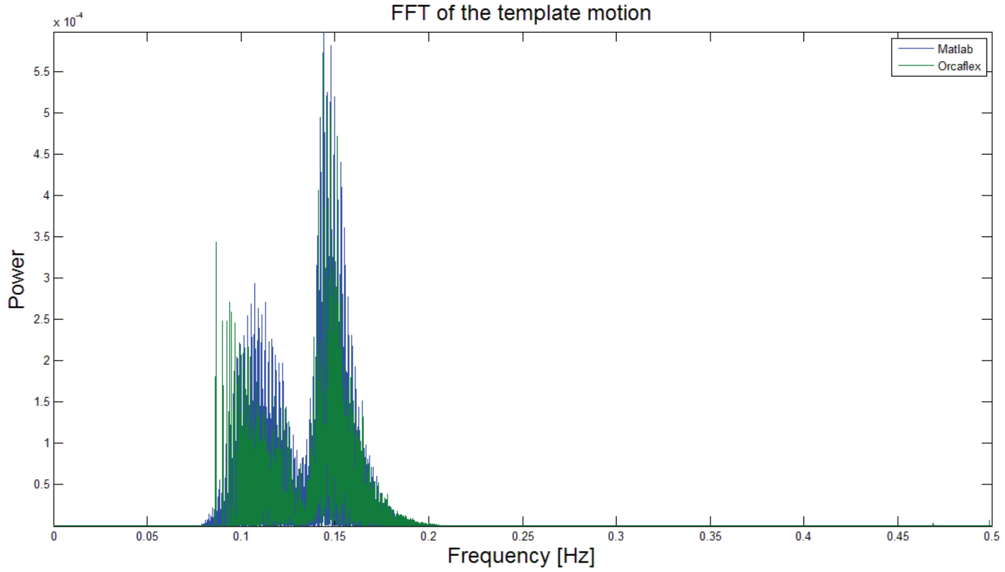


Figure 96 FFT of the template motion in heave after introducing an artificial damping in the 1-DOF analysis.

12.7 FFT plots of 1-DOF and multiple-DOF time histories. Parametric study.

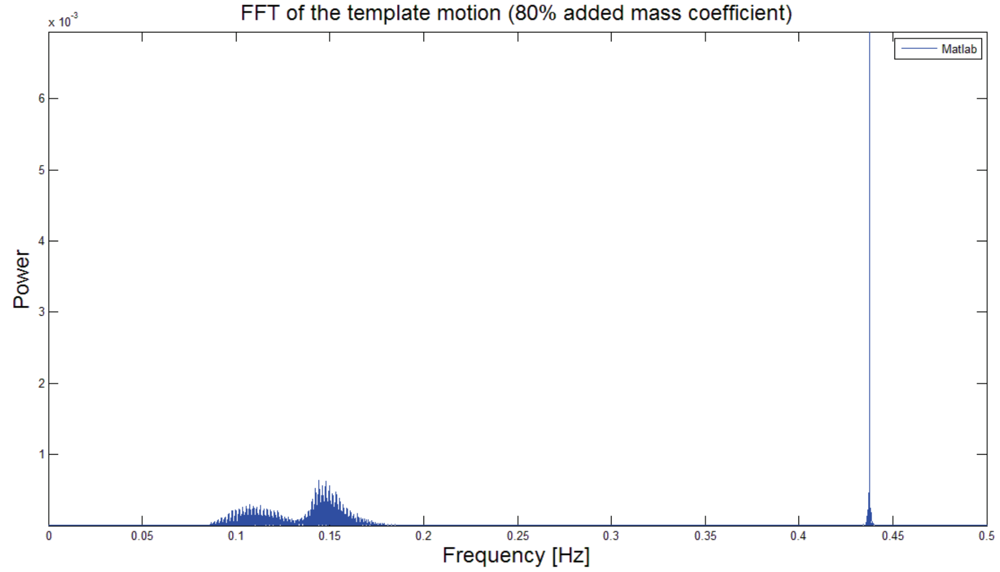


Figure 97 FFT of the template motion in the 1-DOF analysis. 80% of the added mass coefficient.

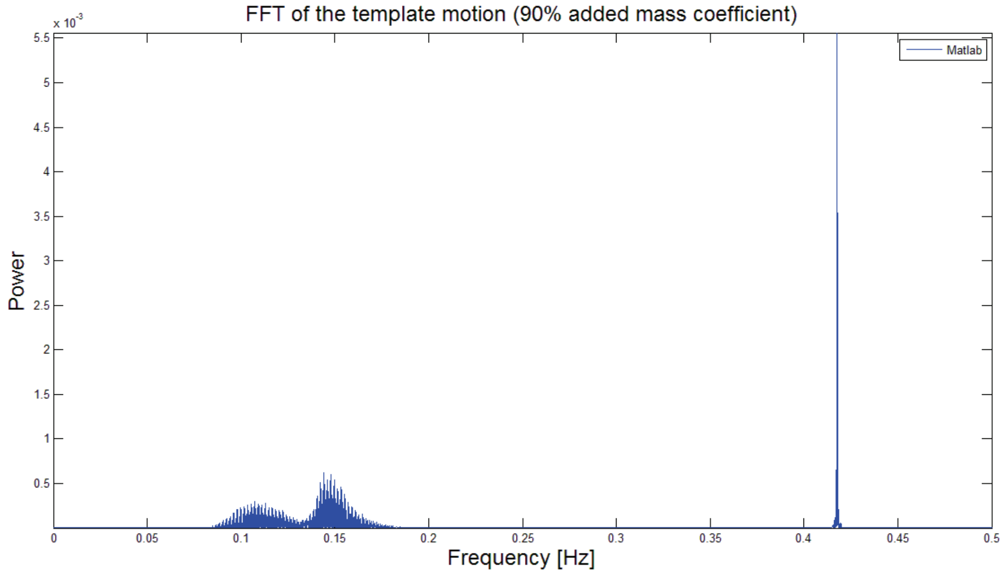


Figure 98 FFT of the template motion in the 1-DOF analysis. 90% of the added mass coefficient.

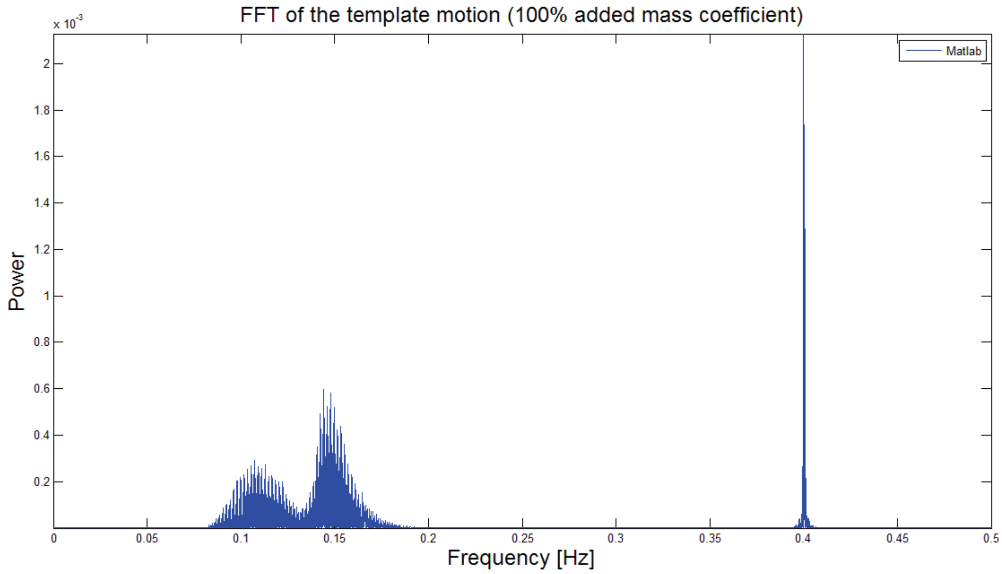


Figure 99 FFT of the template motion in the 1-DOF analysis. 100% of the added mass coefficient.

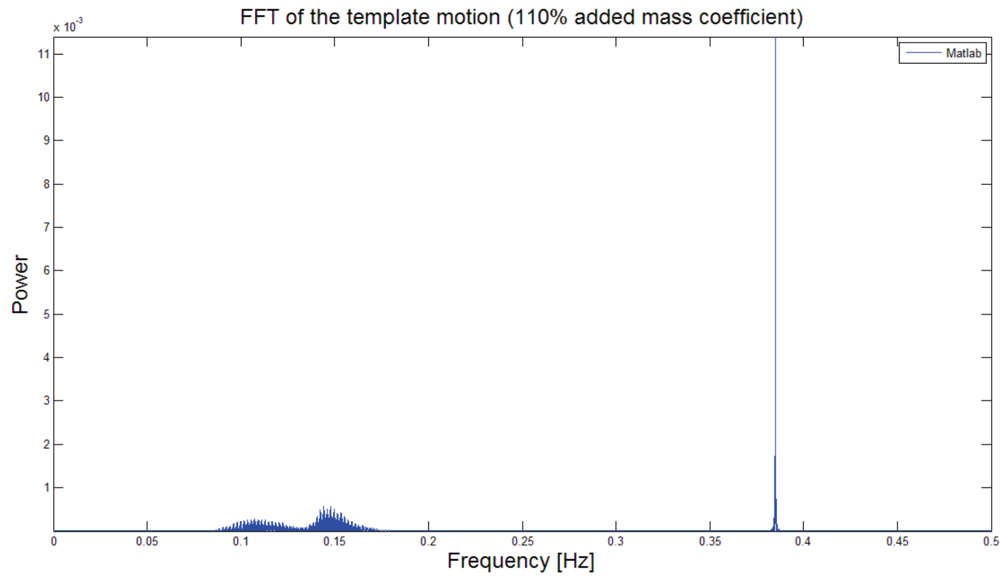


Figure 100 FFT of the template motion in the 1-DOF analysis. 110% of the added mass coefficient.

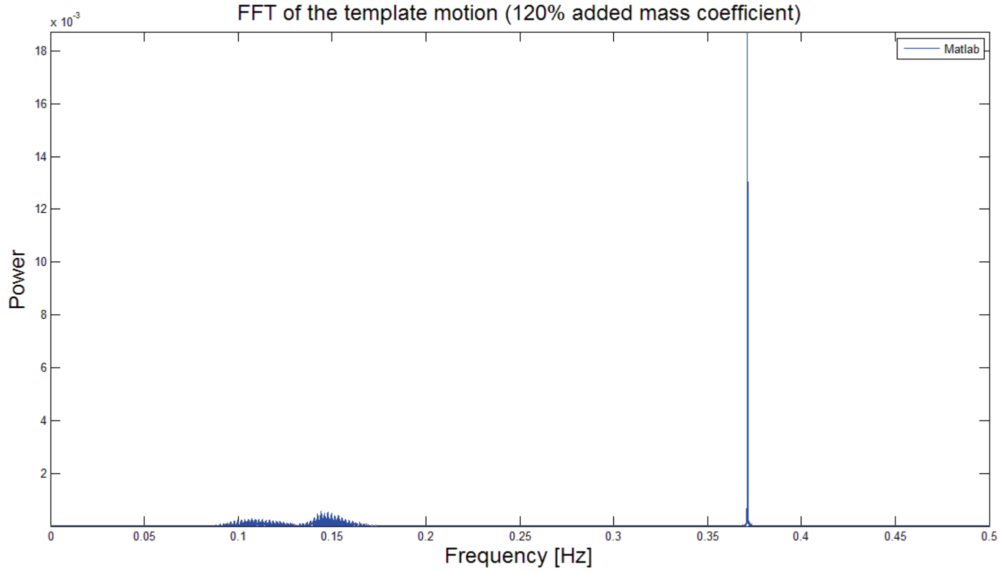


Figure 101 FFT of the template motion in the 1-DOF analysis. 120% of the added mass coefficient.

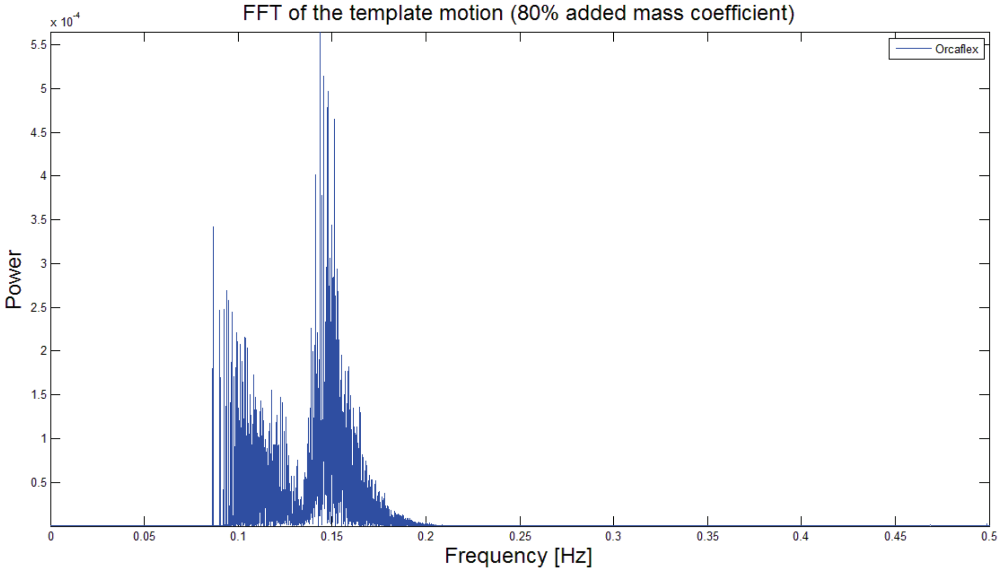


Figure 102 FFT of the template motion in the multiple-DOF analysis. 80% of the added mass coefficient.

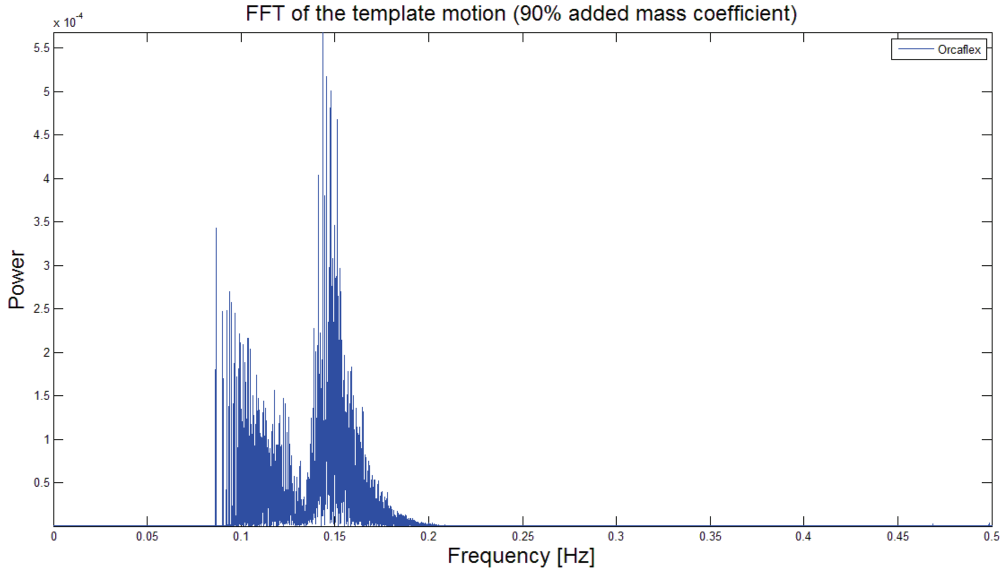


Figure 103 FFT of the template motion in the multiple-DOF analysis. 90% of the added mass coefficient.

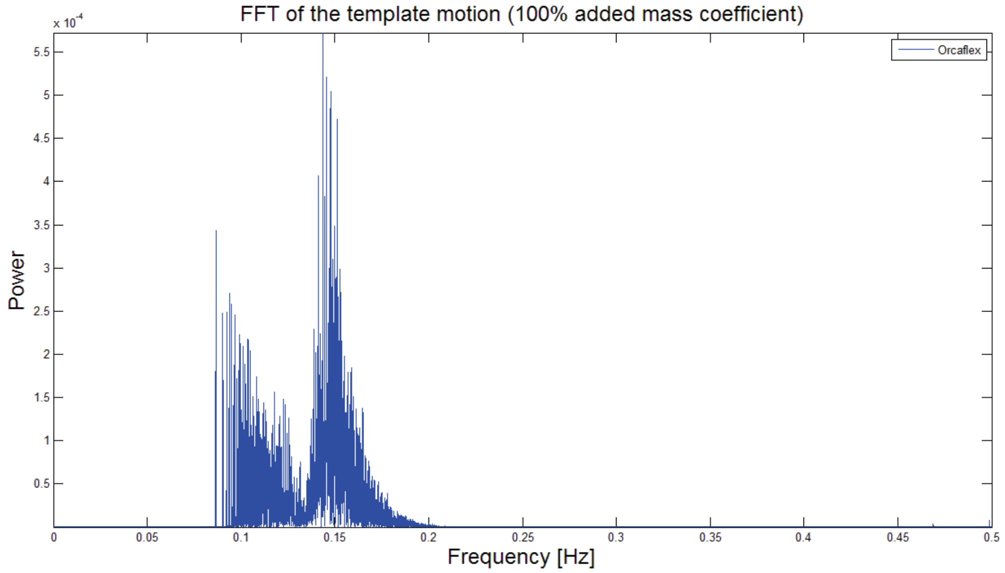


Figure 104 FFT of the template motion in the multiple-DOF analysis. 100% of the added mass coefficient.

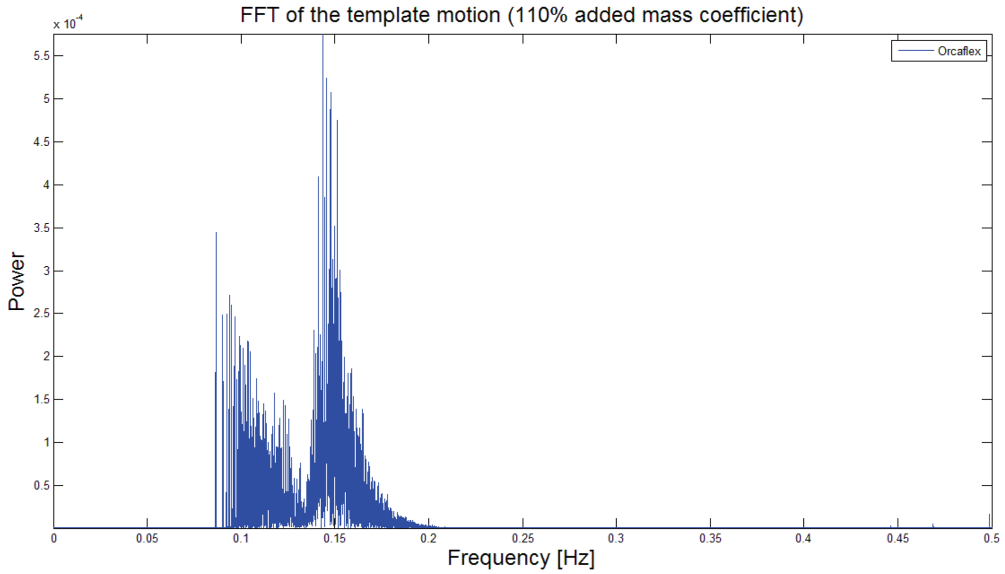


Figure 105 FFT of the template motion in the multiple-DOF analysis. 110% of the added mass coefficient.

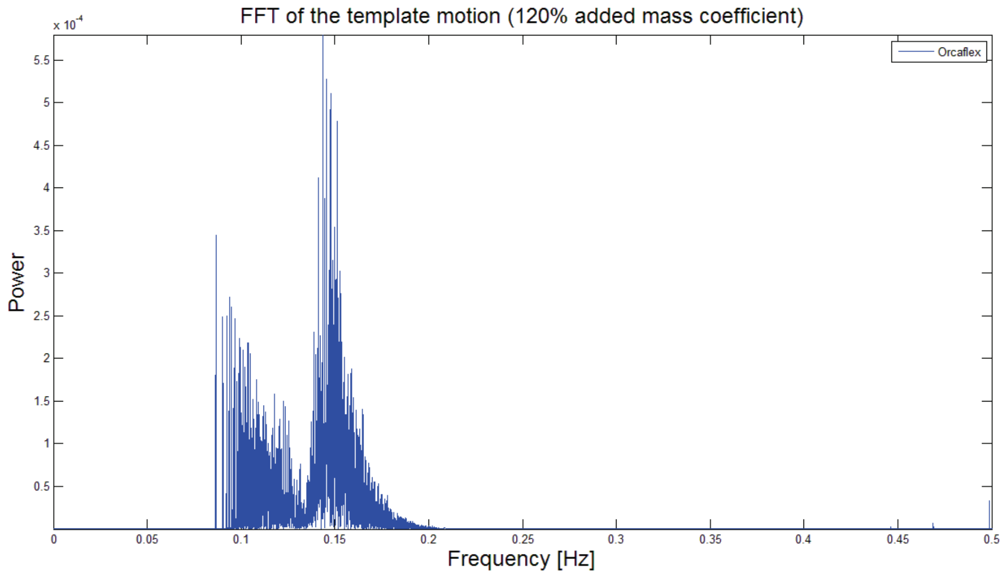


Figure 106 FFT of the template motion in the multiple-DOF analysis. 120% of the added mass coefficient.

12.8 FFT plots of the 1-DOF and multiple-DOF time histories. Parametric study with artificial damping.

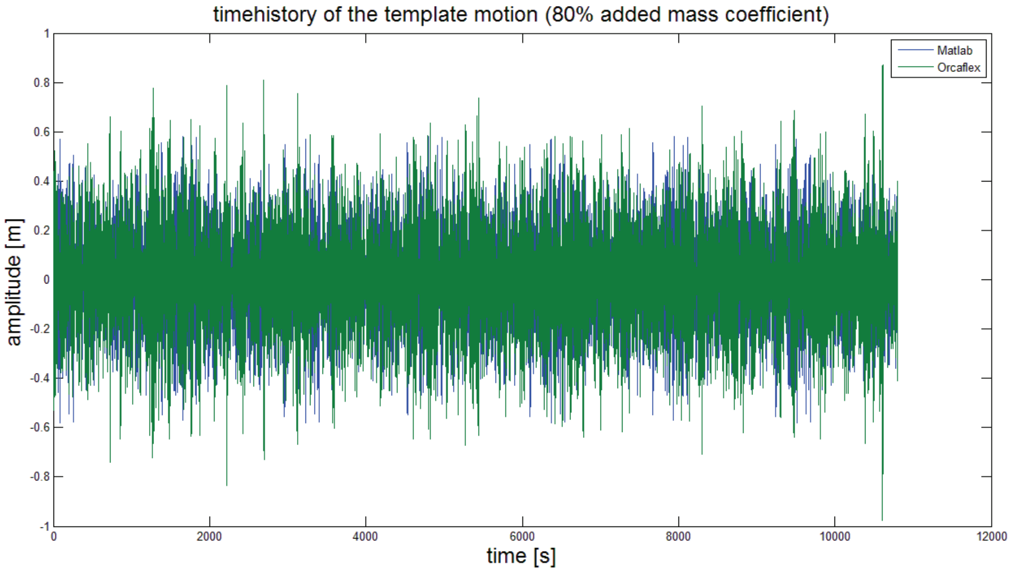


Figure 107 Time history of the surface elevation in the 1-DOF and multiple-DOF analysis. With 80% of the added mass coefficient and artificial damping in the 1-DOF program.

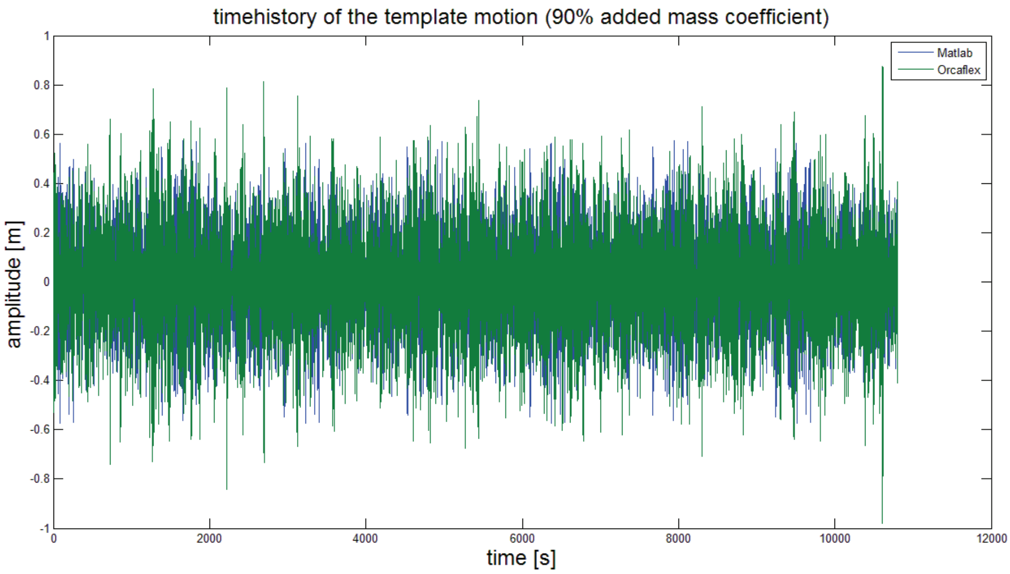


Figure 108 Time history of the surface elevation in the 1-DOF and multiple-DOF analysis. With 90% of the added mass coefficient and artificial damping in the 1-DOF program

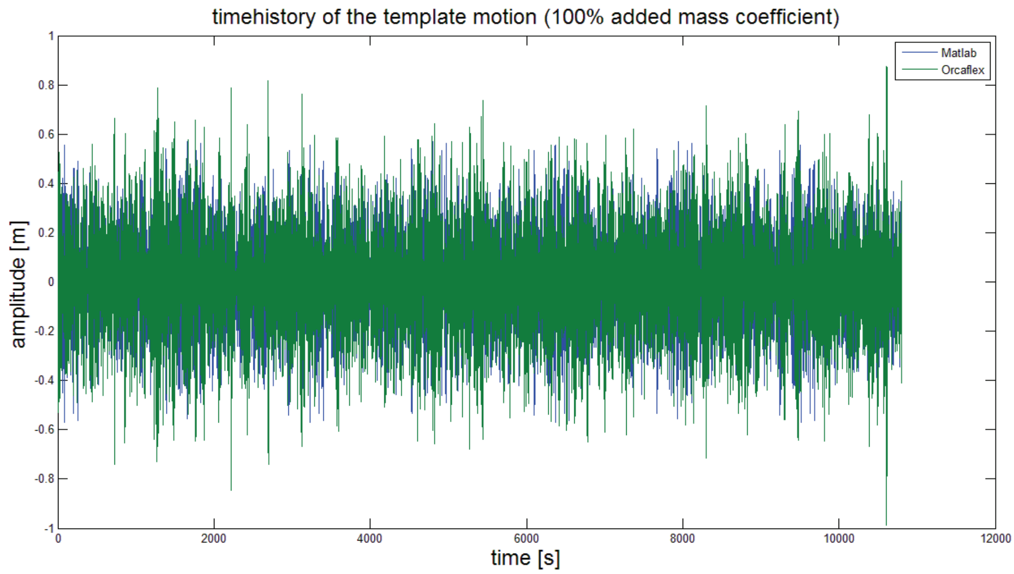


Figure 109 Time history of the surface elevation in the 1-DOF and multiple-DOF analysis. With 100% of the added mass coefficient and artificial damping in the 1-DOF program

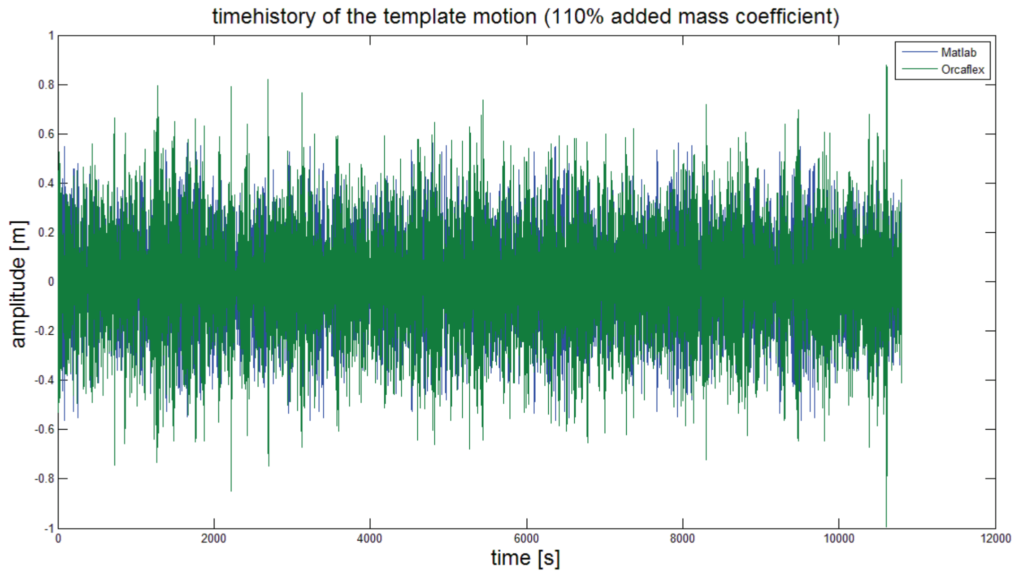


Figure 110 Time history of the surface elevation in the 1-DOF and multiple-DOF analysis. With 110% of the added mass coefficient and artificial damping in the 1-DOF program

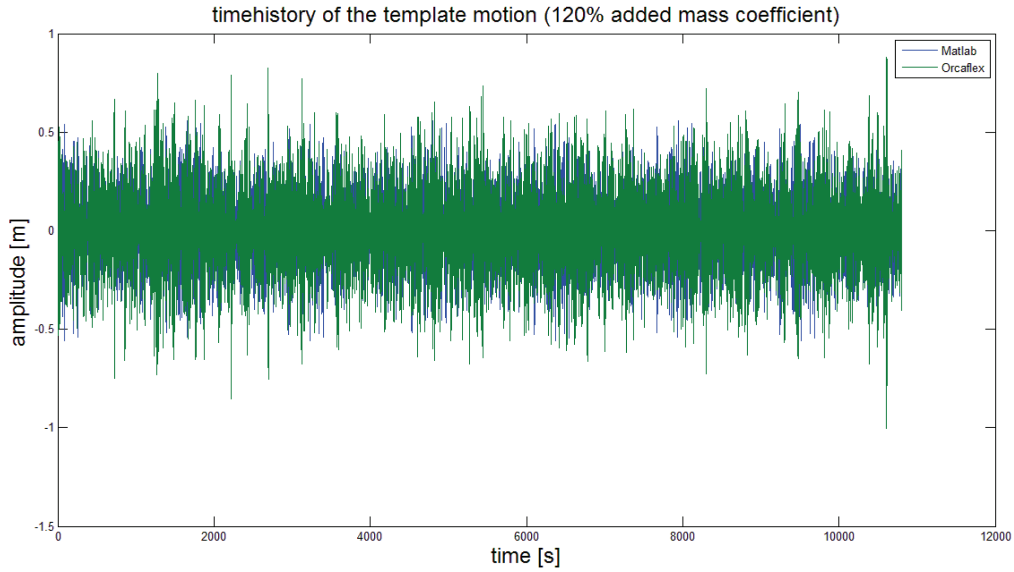


Figure 111 Time history of the surface elevation in the 1-DOF and multiple-DOF analysis. With 120% of the added mass coefficient and artificial damping in the 1-DOF program

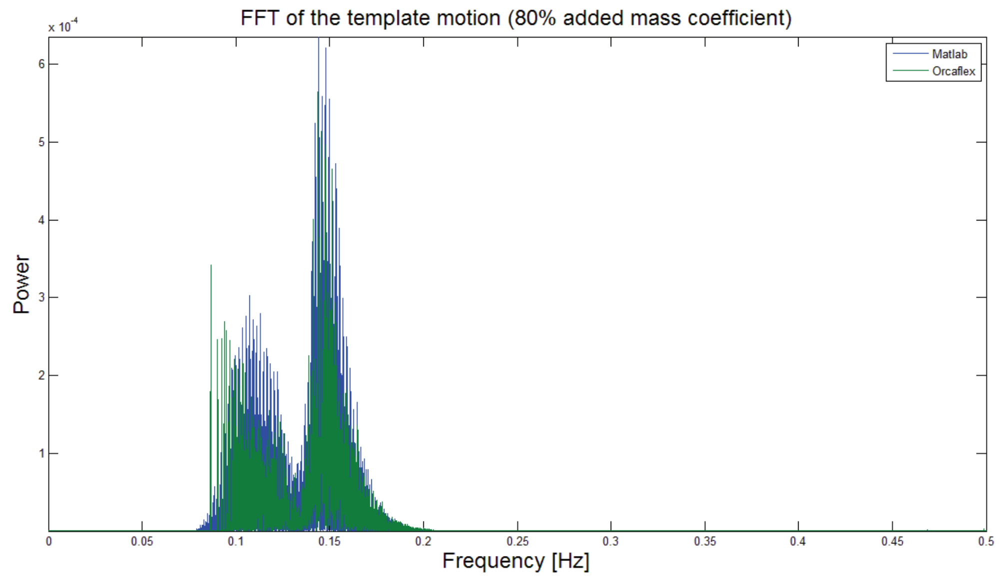


Figure 112 FFT of the surface elevation in the 1-DOF and multiple-DOF analysis. With 80% of the added mass coefficient and artificial damping in the 1-DOF program

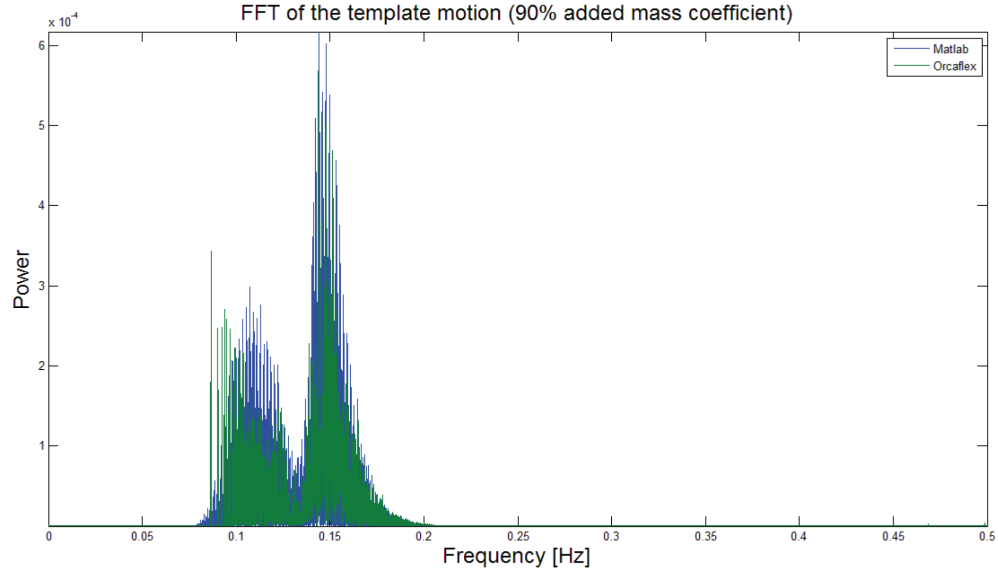


Figure 113 FFT of the surface elevation in the 1-DOF and multiple-DOF analysis. With 90% of the added mass coefficient and artificial damping in the 1-DOF program

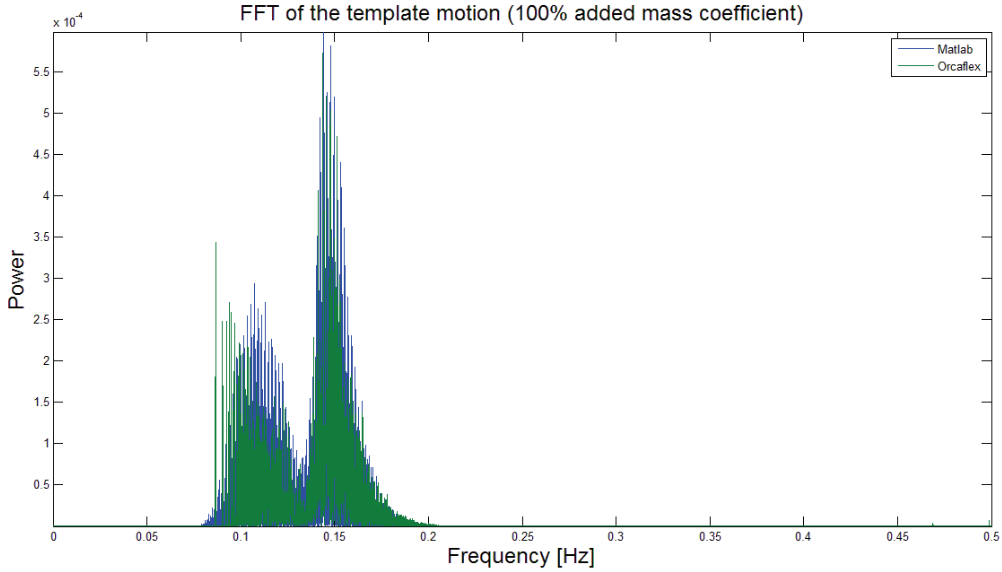


Figure 114 FFT of the surface elevation in the 1-DOF and multiple-DOF analysis. With 100% of the added mass coefficient and artificial damping in the 1-DOF program

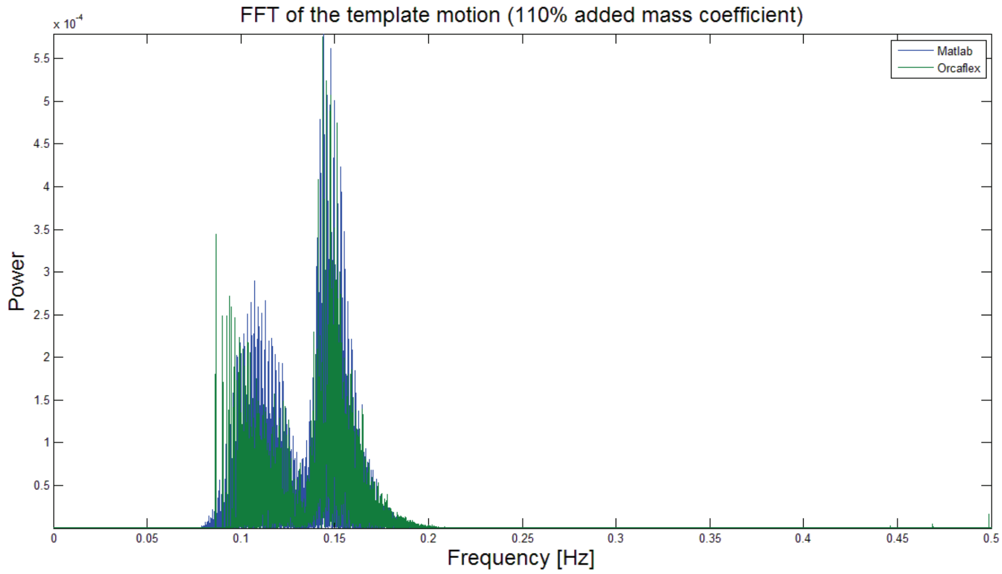


Figure 115 FFT of the surface elevation in the 1-DOF and multiple-DOF analysis. With 110% of the added mass coefficient and artificial damping in Matlabthe 1-DOF program

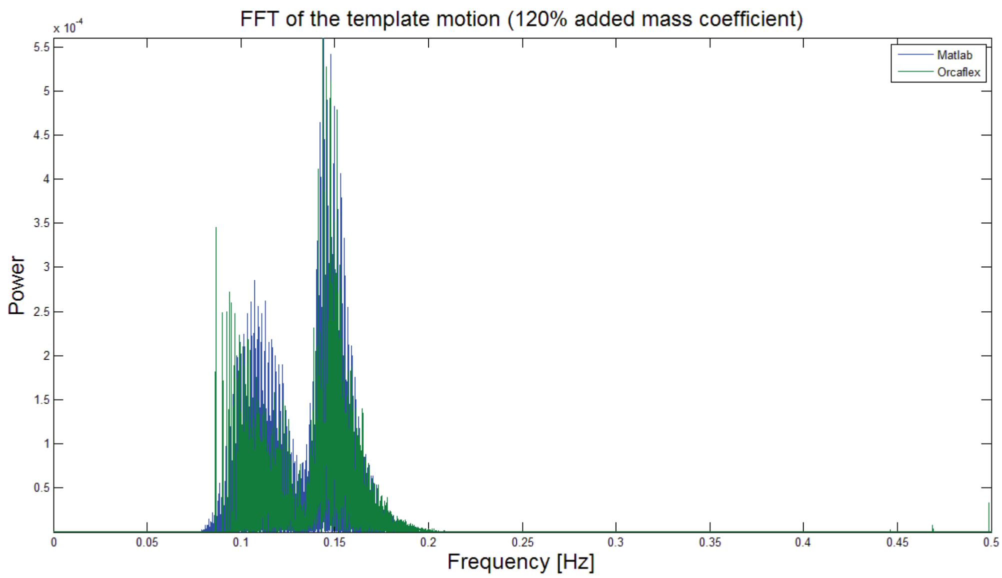


Figure 116 FFT of the surface elevation in the 1-DOF and multiple-DOF analysis. With 120% of the added mass coefficient and artificial damping in the 1-DOF program