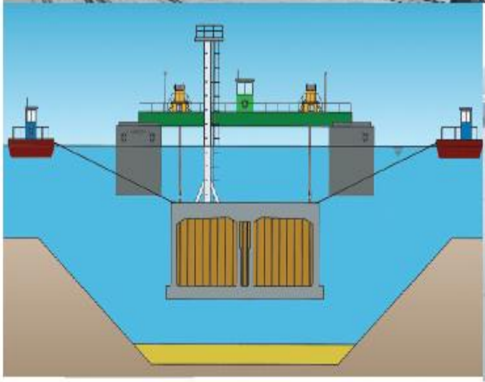
Appendices



Jawid Jelia

TU Delft

1 IMMERSION RIGS AND DESIG ASPECTS

1.1 INTRODUCTION

This part of the report contains details of the analysis carried out to determine the dimensions of the immersion rigs (pontoons) and the required amount of ballast. The pontoons discussed here will be designed such that they will be able to immerse the tunnel elements of the Fehmarnbelt Fixed Link. For the calculations only the standard tunnel elements are analyzed.

Also the loads that should be considered in the design of the immersions rigs are discussed here. In the calculations for the dimensions of tunnel element the dimensions of the illustrative design has been used as a starting point. The dimensions of the illustrative design are provided by the client. The objective of this part of the report is to determine the main dimensions of the immersion rigs (Pontoons: Catamaran and Semi-submersible). In the later stage the static stability will be checked for the determined dimensions. Also the Eigen-periods of the system and the pontoons will be determined.

1.2 ASPECTS OF INFLUENCE ON DIMENSIONS OF PONTOONS:

In the following part of the report some aspects will be explained which determine the required dimensions of the pontoons. The number of the aspects is far from complete. The mentioned aspects play an important role in determining the main dimensions of the pontoons. It doesn’t matter which type of pontoon is selected, it´s first necessary to consider the following aspects:

- Dimensions tunnel elements (only the standard elements are considered in this study)
- Weight of the tunnel elements ‘which depends on:
  1. Amount of the reinforcement
  2. Density of concrete
  3. Construction inaccuracies
  4. Weight for the extra facilities needed for the immersion.
- Salinity of the water (difference in water density in time and depth)
- Amount of the required Ballast ‘which depends on:’
  1. Dynamic wave force
  2. Difference in weight (of the TE)
  3. Safety against uplift (after the TE is immersed)
  4. Difference in salinity

The above mentioned aspects are explained in the following paragraphs and the explanation is given for the chosen values for the modelling purposes.

1.3 WEIGHT OF THE TUNNEL ELEMENT

The buoyancy characteristics of the tunnel elements are very important for the transportation and immersion of the tunnel. Therefore the whole structure needs to be considered. Not just the typical cross section, but also the discrete loads which are not included in the weight of the concrete area. The weight of the bulkheads, immersion chambers, ballast tanks and the equipment needed for the immersion in the element should be considered. The weight of the tunnel elements is one of the most critical loads on the pontoon system. It has influence on the floatability of the pontoons and the required amount of ballast during the immersion. Therefore the following aspects should be considered in the design of the pontoons:

- Concrete density which depends on:
  1. Density unreinforced concrete
  2. Amount of reinforcement
  3. Construction inaccuracies

- Weight Bulkheads
- Weight trimming concrete (can also be used as ballast concrete)
- Weight immersion equipment
- Weight ballast tanks
- Weight immersion chamber

The floating behaviour of the element is important when the tunnel element has been transported to the construction area. The element must be able to float. When the element is placed on the bottom of the ocean it must stay on the place and not float up. In the next part of the report the above mentioned aspects should be explained in a quantitative way.

Weight of concrete

The tunnel elements of the fixed link are primarily built from concrete. The quality and other linked parameters of concrete are of paramount importance for this project. In order to determine the dimension of the immersion rigs the weight of the tunnel element plays an important role. The density properties of the used concrete are more significant for floating structures like the tunnel elements than for other type of concrete structures. The concrete density is one of the factors, which determines the floatability of the tunnel elements. It is very important to understand the density values used in design and that these are achieved in construction. Therefore also the concrete characteristics have to be known.

It's assumed that the concrete is mainly intended to provide the primary water tightness of the tunnel elements, great attention must be paid in achieving a certain quality. Concrete mix design is hence an important part of design and construction for all immersed elements. Mixture proportioning refers to the process of determining the quantities of concrete ingredients, using local materials, to achieve the specified characteristics of the concrete. A properly proportioned concrete mix should possess these qualities:

- Durability, strength, and uniform appearance of the hardened concrete
- Acceptable workability of the freshly mixed concrete
- Economy

It's the concrete technologist job to combine the criteria arising from both the design and the implementation into a suitable composition. Here in this thesis a concrete mix design will be calculated based on the criteria and environmental conditions for the Fehmarbelt Fixed Link project. The property of different concrete mix are explained in paragraph 2.9.2 for the detailed calculations one is referred to appendix 10.

For the calculations of the concrete weight the dimensions of the illustrative design of the TE has been used.

L_{TE} =	217 m	The TE has been assumed as a rectangular box
B_{TE} =	41,2 m	
M_{TE} =	8,9 m	

Simplifications

1) The toe will not be taken into account in the dynamic and static analysis (The toe is taken into account for the calculations for the trimming concrete and determining amount of ballast).
2) Ventilation rich is not taken not into account
3) Shape of the boxes is rectangular.

First the position of the COG of the tunnel element has been determined. If we take the coordinate system at the bottom of the element, the coordinates of the COG are (x,y,z) = (108,5 ; -0,10 ; 4,47)

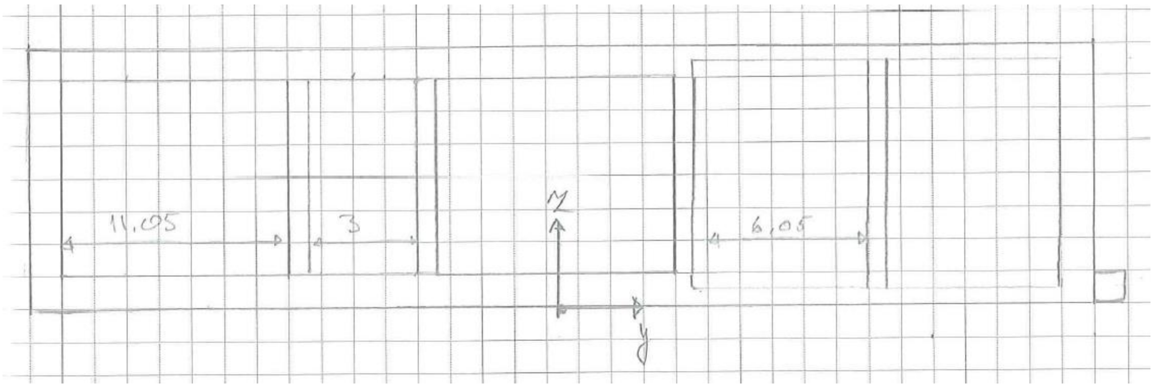


Figure 1 Determination of the COG of the TE

General data parameters:

- A_{con} calculated from the given dimensions of the illustrative design: (A_{con} = 137,3 m^2)

- Concrete volume tolerances ± 0,45 %. Compared to average cross section of TE

$$A_{con(min)} \; = \; A_{con} - \left(A_{con} \cdot \frac{tolerance}{100}\right) = 136{,}68 \, m^2 \tag{1}$$

$$A_{con(max)} = A_{con} + \left(A_{con} \cdot \frac{tolerance}{100}\right) = 137{,}92 m^2 \tag{2}$$

Weight TE without equipment:

$$G_{(min)} = A_{con(min)} \cdot \gamma_{con(min)} \cdot L_{TE} \tag{3}$$

$$G_{(max)} = A_{con(max)} \cdot \gamma_{con(max)} \cdot L_{TE} \tag{4}$$

$$\gamma_{con(min)} = \frac{\omega_{(min)}}{100} \cdot \gamma_{steel} + \left(1 - \frac{\omega_{(min)}}{100}\right) \cdot \gamma_{con;un(min)} \tag{5}$$

$$\gamma_{con(max)} = \frac{\omega_{(max)}}{100} \cdot \gamma_{steel} + \left(1 - \frac{\omega_{(max)}}{100}\right) \cdot \gamma_{con;un(max)} \tag{6}$$

Where

$\gamma_{con;un}$ = unit weight of the unenforced concrete

ω = reinforcement percentage (deduced from the reinforcement weight)

The following values are calculated:

$\gamma_{con(min)}$	= 23,53	[kN/m^3]		
$\gamma_{con(max)}$	= 24,85	[kN/m^3]		
$G_{(min)}$	= 697889,5	[kN]	Or	71141 [ton]
$G_{(max)}$	= 743726,8	[kN]	Or	75183 [ton]

Weight Bulkheads

Bulkheads are watertight partition used to generate compartments. Bulkheads are temporary structures which are provided at the ends of the tunnel element to keep it watertight during the transportation and immersion. The Bulkheads can be made from different materials. Three kind of materials has been used in the immersion projects in the past to make Bulkheads:

- Concrete
- Steel
- Wood

All the three materials have their own advantages and disadvantages. But the wooden bulkheads are not that common to apply anymore. Steel bulkheads are structures build from a steel plate with vertical steel columns. The steel plate is welded to the steel columns. Along the perimeter is the steel plate welded to a cast in steel angle to get a watertight bulkhead.

The concrete bulkheads are built from the vertical reinforced concrete wall. The concrete bulkheads are built with the same principle as the steel bulkheads. Both kind bulkheads has to be demolished and removed after the immersion is been completed to get access to the tunnel elements.

In the case of the Fehmarnbelt tunnel there are 78 standard elements which have to be immersed. It quite interesting if demountable bulkheads can be applied so that the bulkheads can be reused as this was done for the Øresund Tunnel. In this case the Bulkheads will be built from the modular systems instead of the flat conventional steel plate. The advantage of this kind of system is that the bulkheads can be installed and removed quite easily. And the bulkheads can be reused for many elements. The advantage of this system is that the chance of making mistakes will be reduced if the system is implemented proper.

The last mentioned system seems to be more advantageous because of the scale of the project. Also in the past is the same system has been used and there is already experience with this kind of system. In this stage of the study it is been assumed that this kind of system would be applied to the Fehmanrbelt tunnel. For the bulkheads weight a value of 8 KN/m^2 is been used for the calculations.
The weight of the bulkheads depends on the $A_{bulkhea}$ and the unity weight Bulkheads.

$$A_{bulkhead} \quad = 2 \cdot 11{,}05 \cdot 6{,}17 + 2 \cdot 6{,}9 \cdot 6{,}05 \quad = 232{,}2 \ [m^2]$$

$$G_{bulkhead} \quad = Abulkheads \ \cdot \gamma bulkheads = 232.2 \cdot 8 \quad = 1857{,}6 \ [kN] \ or \ 190 \ [\text{ton}]$$

In total the weight of the bulkheads will be:

$$G_{bulkheads} \quad = 2 \cdot A_{\ bulkheads} \cdot \gamma_{bulkheads} \quad = 380 \ [\text{ton}]$$

Weight trimming concrete.

The tunnel elements of the Fehmarnbelt Fixed Link are not symmetrical as it can be seen Figure 1and Figure 2. The elements on the road side are heavier that the tunnel on the rail side. But the weight of the toe on the rail side provide extra moment and weight. Extra weight is needed to trim the tunnel element in y direction. Otherwise it can cause extra difficulties during the transportation.

From the practical considerations is more practical to use concrete as trimming material. Because in the final stage there has concrete to be added to the tunnel element as ballast to prevent that element will be lifted up by the water force. That’s why it’s convenient to add extra concrete also as trimming material.

The rotational moment which will be caused by the eccentricity of the tunnel elements weight and the weight of the temporary equipment can be expressed as:

$$M_{ex} = \{(L_{TE} * A_{con} * \gamma_{con}) + G_{bulkheads} + G_{ballast\ tanks} + G_{immersed\ equipment}\} * e_y$$

The eccentricity moment is equal to 48495 KNm. To compensate this moment there must 149 m^3 ballast concrete be placed on the railway side. To compensate the asymmetry of the TE, concrete blocks will be placed on the rail side see next figure.

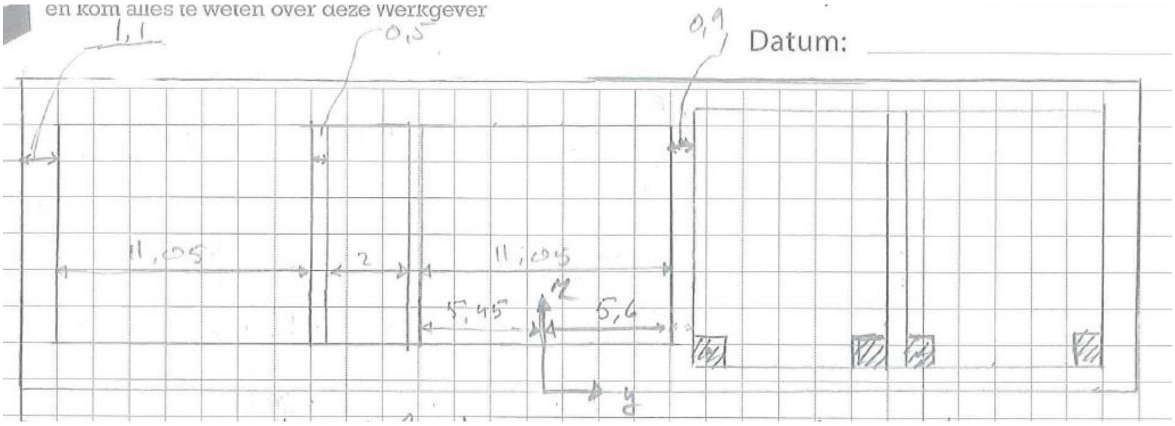


Figure 2 Trimming concrete blocks

The concrete blocks will provide extra moment and prevent the TE from heeling to rail-side. The amount of the concrete is calculated by the aid of defining first the rolling moment due to eccentricity and then determining the needed amount of concrete and the associated moment.

$$M_{roll(max)} = G_{TE(max)} \cdot e_y = 750478{,}735 \cdot 0{,}10 = 5047{,}9[\ kNm] \qquad (7)$$

$$G_{trim} = 598\ [ton]$$

Weight immersion equipment

Before the immersion different temporary equipment will be installed on or in the tunnel element for the transportations and immersion. The weight of this temporary equipment is been estimated as 50 ton. The immersion equipment consist mainly from:

- Required bollards and lifting points
- Generators
- Leakage water pumps
- Communication Services
- Lighting systems.
- Fenders
- Frame for the push boats
- Cables
- Materials and equipment for emergency repairs
- Spare Parts

After the transportation and the immersion the temporary equipment will be removed. So the weight of this equipment has not to be taken into account in the final situations.

Weight ballast tanks

Before the transportation and immersion of the tunnel elements can take place, an internal ballasting system will be installed. The ballast tanks are needed to immerse the tunnel element in a controlled way. If the water will be let into the tunnel element freely to give the tunnel element a negative buoyancy, the element will become uncontrollable during the immersion. It's needed to contain the water ballast in a known en required position within the element.

In this phase of the study it's assumed that for the immersion of a standard tunnel element 4 a 6 ballast tanks will be needed. The construction of the tanks is assumed to be very simple. In the horizontal direction the tanks are supported by the tunnel walls. The access through the tube will be blocked and walkways over the tanks are required. The ballast tanks are assumed to be composited from a PVC liner backed by the plywood panels with the timber wailings and steel column system. The advantage of this system is that this can be dismantled and reused for other elements. The amount of water can be controlled by the pumps in the tunnel element. In this stage of the study the weight of the tanks is assumed to be 250 ton.

Weight immersion chamber

One of the important loads for the weight analysis is the weight of water within the perimeter of the tunnel element at the bulkheads. The length of the perimeter is assumed to be one meter and the thickness of the tunnel element roof slab and the tunnel element base slab are both assumed to be ½ m. The length of the chamber is assumed to be 1 m. At each side of the element, there is one chamber. For the total weight two times the chamber weight has to be taken into account.

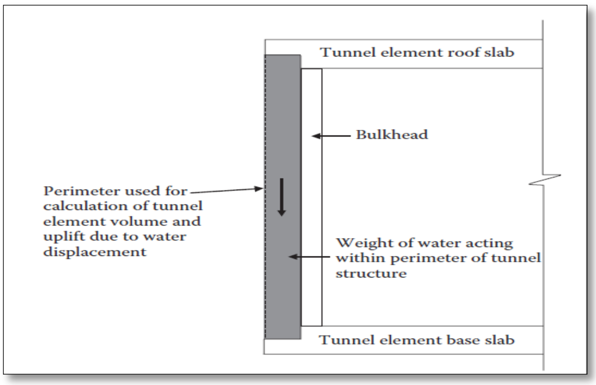

Tunnel element roof slab
Bulkhead
Perimeter used for calculation of tunnel element volume and uplift due to water displacement
Weight of water acting within perimeter of tunnel structure
Tunnel element base slab

Figure 3 Water in immersion chamber

The weight of water within the perimeter of the TE should also be taken as extra weight. At this stage a conservative value for this weight has been taken into account.

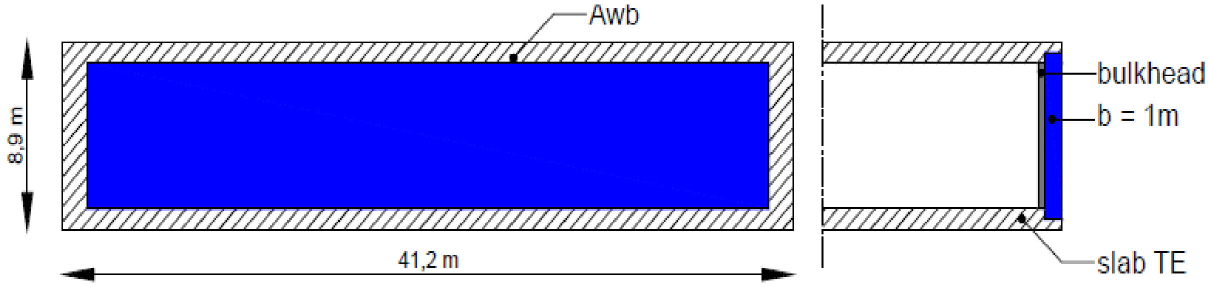


Figure 4 Water in immersion chamber

A_{wb} = Area before bulkheads

B = with/length the bulkheads = 1,0 m (assumed).

$$A_{wb} = (8{,}9 - 2 \cdot 0{,}5) \cdot 41{,}2 - 2 \cdot 1{,}1 = 308.1 \text{ m}^2$$

Volume water in the perimeter = 308,1 m^3

Weight water in perimeter:

$$G_{wb} = 308{,}1 \cdot \gamma_w$$

$$G_{wb(min)} = 308{,}1 \cdot \gamma_{w(min)} = 308{,}1 \cdot \frac{1007 \cdot 9{,}8}{1000}$$

$$G_{wb(min)} = 308{,}1 \cdot 9{,}879 = 3043.6\ [kN]$$

$$= 3043{,}6\ kN\ or\ 310\ [ton]$$

$$G_{wb(total)} = 310 \cdot 2 = 620\ [ton]$$

$$G_{wb(max)} = 308{,}1 \cdot \gamma_{w(max)} = 308.1 \cdot 10.16 = 3131.3\ [kN]$$

$$= 3131{,}3\ [kN]\ or\ 319{,}2\ [ton]$$

$$G_{wb(max)}\ in\ total = 319{,}2\ \cdot\ 2\ = 638{,}4\ [ton]$$

The extra weight in the chamber which is been taken into account is:

- Weight water in the chamber (max) 631 ton
- Weight water in the chamber (min) 616 ton

Total weight

The total weight of the tunnel element during the transportation and immersion is dependent on different aspects as mentioned in the previous paragraphs. It can be concluded that for the transportation and immersion of the elements the minimum and maximum possible weight are of interest.

In evaluating the buoyant behaviour of the tunnel element the nominal values of the weights and the water densities should be considered. No load factors should be applied to this calculations. Two conditions will be assed, namely:

- Light weight condition (Considering the maximum possible water density and the minimum element weight)
- Heavy weight Condition (Considering the minimum possible water density and the maximal element weight)

When we consider the Light weight condition the freeboard of the tunnel element will be maximal. Due to the maximum water density and minimum structural weight and the buoyancy force on the tunnel element more ballast must has to be applied. The Heavy weight condition is the opposite of the Light weight condition. In this condition one consider the minimum water density and the heaviest structural weight. The heavy condition determines the minimum freeboard of the element during the transportation.

Nr.	Weight component	$G_{(max)}$ [ton]	$G_{(min)}$ [ton]
1.	Structural concrete	75183	71141
2.	Bulkheads	380	380
3.	Immersion equipment	50	50
4.	Ballast tanks	250	250
5.	Parameters immersion chambers	638,4	620
6.	Trimming concrete	598	565
	Total weight TE	77100	73006

Table 1 Total weight TE

1.4 HYDROSTATIC FORCE

The main principle of the immersed tunnels is that the elements can float. That's why the elements of the Fehmarnbelt Fixed Link will be produced in a fabric. After the completion they will be transported to the tunnel site by towing the elements. In the tunnel trench on the sea bottom after the immersion, the elements has to be ballasted such that a minimum safety factor against the uplift is achieved in the temporary conditions and the final conditions. This means that enough ballast water in the temporary conditions has to be pumped in the ballast tanks to prevent the uplifting of the elements. The buoyancy force is dependent on the displaced volume water and the water density. The uplift force on the tunnel elements is equal to:

$$F_b = V_{displaced} * \rho_{water} \tag{8}$$

During the immersion the displaced volume water will not vary. But the water density will be variable with the depth. Also seasonal variation are to be expected in the project area. To ensure the tunnel elements behave as expected during the transportation and immersion, it is therefore important to know the water density range in the project area. Fehmarnbelt is an area where estuarial conditions are valid. The interface between the saline and the fresh water will vary with the tides and flow. Also seasonal conditions determine the change of the water density. The values of salinity are given in paragraph 2.8. and 3.8. In the next subparagraph the design values of the water density will be determined.

Further, for determination of the buoyancy force 3 situations are considered:

1) Phase 1: When the element is floated up in the construction area, the freeboard is dependent on the weight of the element and water density.
2) Phase 2: When a floating element will have a freeboard of 0,2 m in fresh water after trimming
3) Phase 3: When the entirely element is submerged.

Variation in density of the water

Water densities play an important role in both the temporary conditions and the permanent conditions as mentioned before. In the temporary conditions during the immersion the salinity rate determine the amount of ballast which has to be added. In the permanent conditions the water density determine the water pressure on the bottom of the tunnel element. For the Fixed Link project the salinity levels are quite predictable. For the Fehmarnbelt area there is sufficient amount of data to predict the salinity (see also the appendix 2). But it still quite challenging to predict salinity levels because of Estuarial conditions.

The Fehmarnbelt is a transitional area between the Baltic Sea and the North Sea. The stratification in the Fehmarn Belt is strongly related to water exchange between the North Sea and Baltic Sea. The surface water flows from Baltic Sea with low salinity. The lower layer of the water column is manly water from North Sea with height salinity rate. The salinity of water is to be expected increase with the incoming tide. And the salinity will reduce when the tide recedes and the saline water is flushed out by the river water. Also seasonal variation in salinity occurs.

It's important to know the salinity variation over the depth. Because the variation of salinity determine the amount of ballast water which has to be pumped into the ballast tanks to still have negative buoyancy during the immersion process. Also there is tendency that the saline water will accumulate in the trench. So before the tunnel element is uncoupled there must be enough overweight to overcome the buoyancy force. It is dependent on the choice of the execution method. One can put the extra ballast when the tunnel element is near the surface. That will lead to extra capacity of the floating pontoons and the winches and the suspension cables. It's also possible to measure the salinity during immersion across the water depth and when increase of the salinity will be analyzed then extra ballast water will be pumped into the ballast tanks.

When the tunnel element will be uncoupled from the pontoons it will be quite sensitive to the uplift forces. Because no extra material is putted on the roof of the tunnel. Also the ballast concrete will be not present in sufficient amount. That's why an extra amount of water will be pumped into the tunnel element to get extra vertical stability. The tunnel will be ballasted with an overweight of 2.5%. The salinity rate on the bottom influences the safety against the uplift in the temporary and final conditions. The salinity rate of the surface water influences the floatability of the tunnel elements. In saline water the freeboard of the tunnel element will be greater than in fresh water.

The water column is stratified most of the time in the Fehmanrbelt. That´s why for the calculations a stratified water column is assumed. The precise shape of the stratification is difficult to predict because it is also time dependent. Generally it can be noted the transitional area in the vertical direction is around 15 m (approximately). The associated water density fluctuations in the vertical of 5 kg/m^3 (maximum) can be observed from the available data. Normally the water density fluctuation are smaller.

During a period of 19 years the salinity rate has been measured at the Fehmarnbelt light vessel (located approximately east of the fixed link corridor). The monthly variation are originally reported and discussed in (FEHY, 2012a). *Those 19 years of data were considered representative for the Fixed Link* (Metocean Conditions 2,2013).

Also during the period 1931-1993 the salinity of the surface water has been measured at Rødbyhavn. The monthly extreme values of the variations are presented in the appendix 2. The annual mean salinity is measured was 12.2 psu. There is large span in the variation of the extreme values of salinity. In the 62 years measured data which are available for the project locations the following extreme values are obtained:

- Minimum measured salinity 6 psu
- Maximum messured salinity 27 psu

In order to determine the extreme values of the water density the salinity rate values are translated to the water density. The calculation principle is given here below.

$$M_{salt} = \frac{27}{1000} * 2150 = 58.05 \frac{kg}{m^3} \qquad (9)$$

$$V_{salt} = \frac{M_{salt}}{\rho_{salt}} = \frac{58.05}{2150} = 0.027\, m^3 \qquad (10)$$

$$V_{water} = 1 - V_{salt} = 0.973\, m^3 \qquad (11)$$

$$M_{water} = 0.973 * 1000 = 973 \frac{kg}{m^3} \qquad (12)$$

$$\rho_{water(\max)} = 973 + 58.05 = 1031 \frac{kg}{m^3} \qquad (13)$$

The same approach is followed to determine water density for other given salinity rates. The minimum water density measured in the 62 years is: 1007 kg/m^3. But as mentioned before the water density at the bottom is approximately 5 kg/m^3 more than the water density at the surface. With the given data the following values are obtained for the extreme values of the water density.

Water density	Symbol	Value	Units
Minimum water density at the surface	$\rho_{water(\min)}$	1007	$\frac{kg}{m^3}$
Maximum water density at the surface	$\rho_{water(\max)}$	1031	$\frac{kg}{m^3}$
Minimum water density at the bottom	$\rho_{water(\min)}$	1012	$\frac{kg}{m^3}$
Maximum water density at the bottom	$\rho_{water(\max)}$	1036	$\frac{kg}{m^3}$

Table 2 Water density

Phase 1

When the element is floating, the buoyancy force is dependent on the freeboard and curvature of the element. For the calculations, it's assumed that the element is straight with a zero curvature. Therefore, freeboard along the whole element is equal.

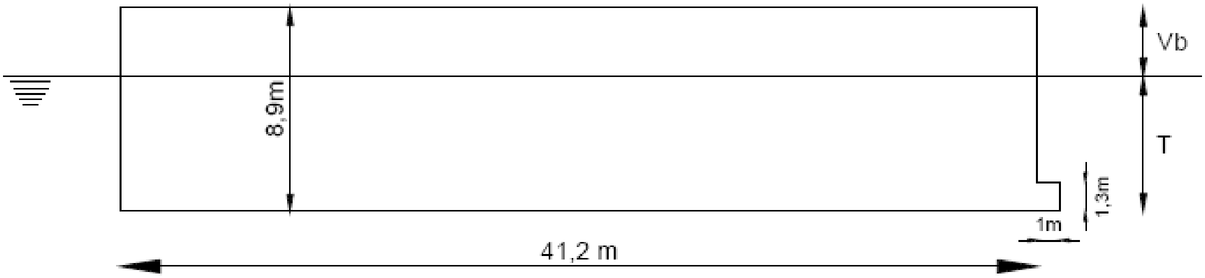


Figure 5 Floating tunnel element

Area TE under water:

$$A_{uw} = (41{,}2 + 1) \cdot 1{,}3 + (T - 1{,}3) \cdot 41{,}2 \tag{14}$$

Buoyancy forces:

$$F_b = A_{uw} \cdot S_w \cdot L_{TE} \tag{15}$$

In phase 1 the following equilibrium is applied

$$\begin{gathered} F_b = G_{TE} \\ A_{uw} \cdot \rho_w \cdot L_{TE} = G_{TE} \\ (42{,}2 \cdot 1{,}3 + (T - 1{,}3) \cdot 42{,}2) \cdot L_{TE} = \frac{G_{TE}}{\rho_w} \\ 11904{,}62 + 8940{,}4\,T - 11622{,}5 = \frac{G_{TE}}{\rho_w} \\ T = \left(\frac{G_{TE}}{S_w} - 282{,}10\right) \cdot \frac{1}{8940{,}4} \end{gathered} \tag{16}$$

When G_{TE} is maximal and $\rho_w = \rho_{fresh}$ → $T_{max} = 8.6\ m$

When G_{TE} is minimal and $\rho_w = \rho_{salt}$ → $T_{min} = 7.89\ m$

Phase 2

In this phase the TE will be trimmed such that the freeboard will be 0,2 m. So T will be 8,7 m. The exact amount of ballast concrete is dependent on the final structural weight of TE and water density.

$$F_b = A_{uw} \cdot \rho_w \cdot g \cdot\ L_{TE} = (42{,}2 \cdot 1{,}3) + \big((8{,}7 - 1{,}3) \cdot 41{,}2\big) \cdot 217 \cdot \gamma_w = 78063{,}6 \cdot\ \gamma_w$$

$$F_{b(max)}\ = 78063{,}6 \cdot 10{,}11 = 7{,}89 \cdot 10^5\ [kN] \quad \text{Or} \quad 80450{,}8\ [ton]$$

$$F_{b(min)}\ = 78063{,}6 \cdot 9{,}88 = 7{,}71 \cdot 10^5\ [kN] \quad \text{Or} \quad 78620{,}61\ [ton]$$

Phase 3

During phase 3 the TE will be submerged and the buoyancy force depends on the position under free surface (actually the buoyancy force depends on water density)

$$F_b = A_{uw} \cdot\ L_{TE} \cdot\ \gamma_w = 79851.4 \cdot\ \gamma_w$$

$$F_{b(max)}\ = 8{,}11 \cdot 10^5\ [kN] \quad \text{Or} \quad 82726{,}3\ [ton]$$

(This force works on the TE when it's nearby the tunnel trench and ρ_w= 1036kg/m^3)

$$F_{b(min)}\ = 7{,}89 \cdot 10^5\ [kN] \quad \text{Or} \quad 78620{,}61\ [ton]$$

(This force works on the TE when the water density is ρ_w= 1007 [kg/m^3], this is when the element nearby the water surface).

Amount ballast:

In order to overcome the buoyancy force, it has been looked to the force on TE in Phase 2 which means that the weight of TE will be also equal to the buoyancy force.

$$F_b\ =\ G_{TE} \tag{17}$$

Also it has been looked to the F_b during phase 3 when the buoyancy force bigger than in phase 2.

$$F_b = 82726{,}3 - 80450{,}8\ =\ 2275{,}5\ [ton]$$

$$F_b = 79851{,}7 - 78620{,}61 =\ \ 1231\ [ton]$$

In the calculations it's been taken into account that $F_{b(min)}$ in phase 2 and $F_{b(max)}$ in phase 3 can't occur together, because the greatest difference in salinity measured over the depth is 5 kg/m^3. The difference in the maximum buoyancy force is determinative for the design of the capacity of the ballast tanks.

In phase 2 also the element should be trimmed, such that it can have a freeboard of 0,2 m. In order to get the desired freeboard of 0,2 m the element has to be ballasted:

$$G_{(ballast);(min)} = G_{(max)} - F_{b(min)} = 77100 - 78621 = 1521\ [ton]$$
$$G_{(ballast);(max)} = G_{(min)} - F_{b(max)} = 73006 - 80450{,}8 = 7445\ [ton]$$

In the extreme case there should be 7445 Ton of ballast concrete added to TE in phase 2. The extra ballast should be added in the at the factory. It has to be mentioned that the advantage of this method is that at the construction site limited amount of ballast will be needed. The construction factory will be near by the construction site. It's been assumed that the tunnel will be immersed direct after arriving at the construction site. For the immersion moderate wave climate is needed. That's why also during the transportation the wave climate will be moderate.

Force due to density variation

The variation of density with depth due to stratified condition in the Fehmarnbelt has been taken into account. The density variation over the vertical has been assumed to be 5kg/m^3 . If the tunnel element is lowered from a layer with lower density to a layer with higher density, the buoyancy force will be increased. The increase in the buoyancy force can be given as:

$$\Delta F_b \ = \ L_{TE} \cdot \ B_{TE} \cdot \ \Delta \mathrm{h} \cdot \ \Delta \mathrm{\rho_w} = 438.5 \ \cdot \ \Delta \mathrm{h} \ \ [\mathrm{kN/m}] \tag{18}$$

Where $\Delta\mathrm{h}$ is the draught in layer with higher density. In the case of Fehmarnbelt, the water density changes approximately at a depth of 15 m under the free surface. During the immersion the TE should be ballasted with more water approximately at a depth of 15 m in order to overcome the extra buoyancy force.

In the case of Fehmarnbelt, it's possible that the density profile over the trench could well reach a wider density range. Once the saline bottom water from North sea with relatively a higher density has filled the trench, at least it will tend to stay there. When the tunnel element has been immersed in relatively brackish water, one should encounter a density increase of 1036 – 1007= 29kg/m^3.

The total increase in buoyancy force is:

$$\Delta F_b \ = \ L_{TE} \cdot \ B_{TE} \cdot \ \Delta \mathrm{h} \cdot \ \Delta \mathrm{\rho_w} = 2543 \ \cdot \ \Delta \mathrm{h} \ \ [\mathrm{kN/m}] \tag{19}$$

Of course, this buoyancy force does not appear suddenly, but builds up gradually. In reality the density change over the depth is much less. The maximum change over depth is measured in time record is 5 kg/m^3.

In order to guaranty enough safety in the design stage the above mentioned value of $\Delta F_b = 438.5 \ \left[\frac{kN}{m}\right]$ has been taken into account in the design of the pontoons.

$$\Delta F_b \ = \ L_{TE} \cdot \ B_{TE} \cdot \ \Delta \mathrm{h} \cdot \ \Delta \mathrm{\rho_w} = 2543 \ \cdot \ \Delta \mathrm{h} \ \ \left[\frac{\mathrm{kN}}{\mathrm{m}}\right] = 438.5 \cdot 8{,}9 = 3903 \ [kN] \ \text{ or } 398 \ [ton] \tag{20}$$

Another extreme situation which could occur is that half of the element length is submerged in water with higher density and another half with lower water density. This could happen if a rather sharp front passes or when a passing ship in the Fehmarnbelt causes internal waves in the two layered system. If we assume a density difference between the layers of 5 kg/m^3 the increase of the buoyance force over the length can be given as:

$$\Delta F_b \ = \ B_{TE} \cdot \ H_{TE} \cdot \ \Delta \rho_w \cdot \ \Delta \mathrm{L} = 41{,}2 \cdot 8{,}9 \cdot 5 \cdot \frac{9{,}8}{1000} \cdot \ \Delta \mathrm{L} = 18 \ [\frac{\mathrm{kN}}{\mathrm{m}}] \tag{21}$$

ΔF_b over the length will be then:

$$\Delta F_b \ = 3903 \ [kN] \ (or \ 398 \ [ton]) \tag{22}$$

The change of tension in the suspension cables in this situation is dependent on the arrangement of pontoon. If we assume that the suspension cables will be spanned on $\frac{1}{5} \cdot \ L_{TE}$ from the ends of the element, then is the distance from end and suspension cable:

$$distance \ from \ the \ edge = \frac{1}{5} \cdot 217.8 = 43{,}56 \, m \tag{23}$$

The distance between suspension cables on pontoons:

$$distance \ between \ the \ suspension \ cables = 217.8 - 43{,}56 \cdot 2 = 130{,}2 \, m \tag{24}$$

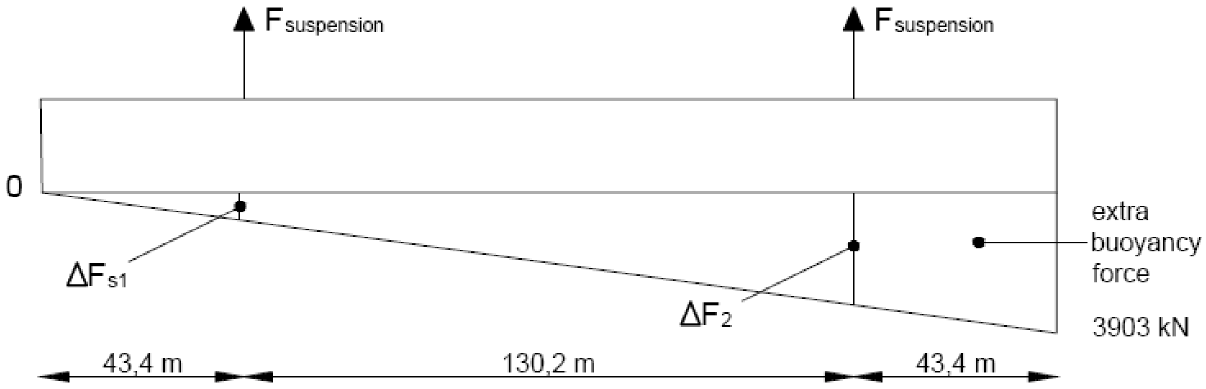


Figure 6 Floating tunnel element

$$\Delta F_{s1} = 43{,}56 \cdot 18 = 784\ [kN]\ (or\ 80[ton]) \tag{25}$$

$$\Delta F_{s2} = 3121{,}8\ kN\ (or\ 318{,}2\ [ton]) \tag{26}$$

F_s Denotes the total force in the suspension cables. (27)

In order to overcome the extra buoyancy forces due to the stratification in longitudinal and vertical directions some extra ballast will be applied during the immersion phase. The extra ballast could be able to compensate the mentioned extra buoyancy forces.

$$F_{extra\ buoyancy} = 318.2 + 398 = 716.2\ [ton] \tag{28}$$

Dynamic wave load:

The response of the element is determined by the natural periods of the element (will be calculated later on). During the immersion process the stability of the element is determined by the stability of pontoons. In no circumstances the suspension cables may be slacken. The suspension cables will be ballasted such that in no circumstances may not become tensionless. The variation of the pretension force in the suspension cables may be calculated by assuming that the pontoons staying in position and that the displaced volume of the pontoons varies with the wave surface. The variation of displaced volume is the design wave height multiplied by water plane area of the pontoons and the force variation is calculated by multiplying the displaced volume by ρ_w. The dynamic wave force can be expressed as:

$$F_w = H_d \cdot A_p \cdot \rho_w \cdot g \tag{29}$$

Because in this stage the water plane area of the pontoons is unknown. The calculations of wave force will be taken in the iterative manner in the calculations.

1.5 DIMENSIONS PONTOONS:

Floating capacity pontoons:

In order to determine the needed dimensions of the immersion rigs and needed equipment it's needed to know the expected loads and equipment properties. For the determination of the pontoons dimensions the floating capacity of the pontoons is taken as strength parameter and the loads are different weights which are acting on the pontoons during the immersion. Also current forces and wave loads which are acting on the pontoons will be taken into account. The structural design of the pontoons and the tunnel element is not considered in the model. The design rules for the required floating capacity are taken into consideration. The design rules refer to the failure modes. For the determination of the floating capacity of the pontoons and related pontoon dimensions the following approach has been followed:

The applicable design situations are chosen such that the pontoons can fulfill its function during the immersion operation. The design situations considered here refer to temporary conditions in which one of the tunnel elements has to be immersed.

Situation 1
In this situation, the following loads are considered:

1) Self-weight of the pontoons and immersion equipment (G_{pon})
2) Ballast weight needed to compensate the dynamic wave load (F_w)

Situation 2
The following loads are considered in this situation:

1) The self-weight of the pontoons and immersion equipment (G_{pon})
2) Weight ballast needed to overcome the extra buoyancy force.
3) Ballast needed to overcome the dynamic wave load (F_w)

Situation 3
In this stage is the load considered from damaged situation of the pontoon. If one of the compartments of the pontoon will be flooded, the weight of the extra water has to be taken into account. The dimensions of one compartment has been assumed as:

$$\left((L \cdot B \cdot H)(9 \cdot 3 \cdot 5) = 135\ m^3\right)$$

The following loads are considered:

1) The self-weight of the pontoons and immersion equipment (G_{pon})
2) Ballast needed to overcome the dynamic wave load (F_w)
3) Weight of the flooded compartment

In order to get the design loads, the load factors given in table 1 are used.

Dimensions Catamaran pontoon

The following result are observed for different load situations:

Design situation	Determinative Load	Units
Situation 1	4557	[ton]
Situation 2	5273	[ton]
Situation 3	4714	[ton]

Table 3 Forces on Catamaran pontoon

From observing different load situations it's obvious that situation nr.2 provide the largest load on the pontoons. For the immersion, two pontoons will be used and each pontoons consist from 2 floaters. The determinative floating capacity is for each floater:

$$Floating\ capacity\ one\ floater = \frac{5273}{4} = 1319\ m^3 \tag{30}$$

For determining the floating capacity of one floater, the freshwater density is used for the calculations. The pontoon should also have freeboard to prevent that wave will pass over the pontoon. Instead the height of the floater. The draft T will be used in the calculations.

Volume floater can be given as:

$$V_f = B_f \cdot L_f \cdot T = 1319\ m^3$$

The following dimensions has been chosen

$$L * B * H = (38 * 7 * 5)$$

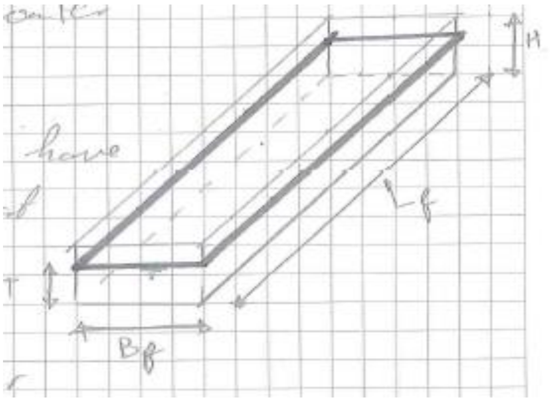


Figure 7 Capacity of a floater

The pontoons should have also a free board. The significant wave height during the normal conditions is: $H_s = 0{,}57\ m$. The design wave height has been assumed to be 2,1 · H_s = H_d

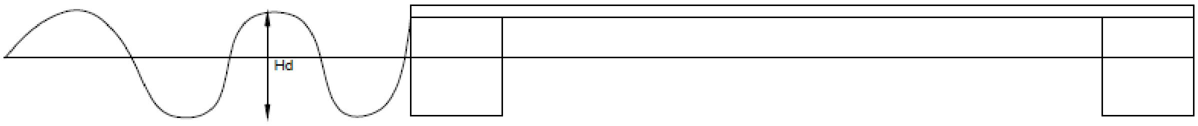


Figure 8 Scattering waves

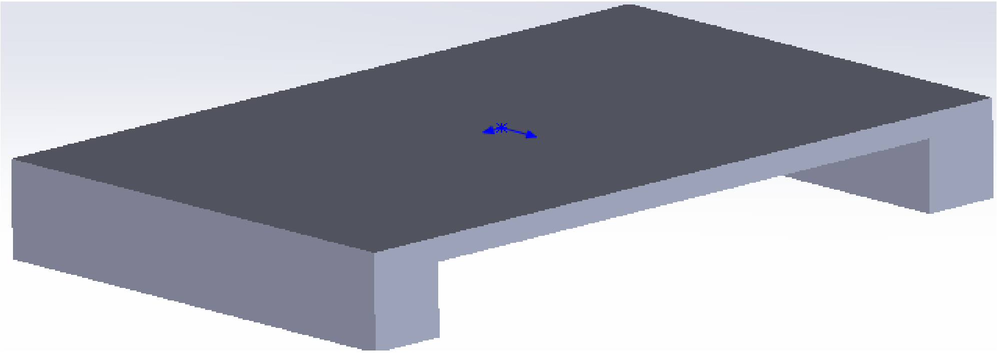


Figure 9 3D sketch of the catamaran pontoon

If we assume that the wave will be fully reflected by the pontoons then the required freeboard is 2 · H$_d$. which means freeboard = 4 · 0,57 = 2,28m. Instead of practical reasons the freeboard is chosen 2,5 m so the total height of pontoon becomes 7.5 m.

This gives the following pontoon dimensions.

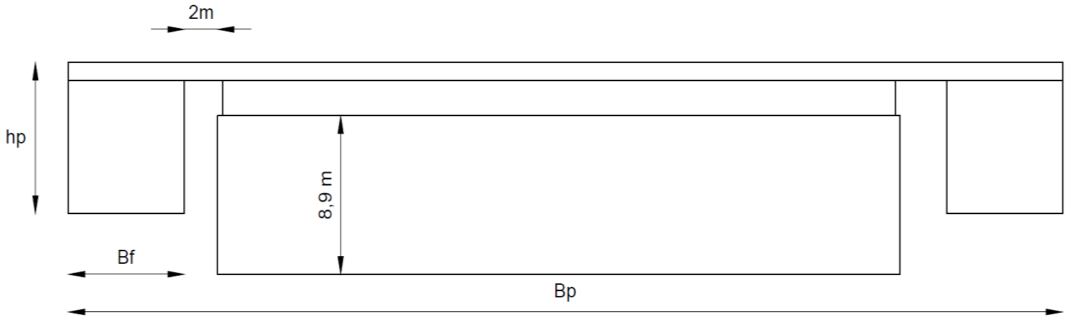


Figure 10 catamaran pontoon

Dimension	Value	Units
With pontoon B_p	60	m
Length pontoon L_p	38	m
Height pontoon h_p	8.5	m
With floater B_f	7	m
Length floater L_f	38	m
Height floater h_f	7.5	m
With pontoon deck B_D	60	m
Length pontoon deck L_D	38	m
Height pontoon deck h_d	1	m

Table 4 Overall dimensions of the Catamaran pontoon

Dimensions semi-submersible type pontoon

The considered load situations for the semisubmersible type of pontoon is the same as for the catamaran pontoons. But the dynamic wave load for the semi-submersible is considerably less than for the catamaran. This is because water plane area A_p of this type of pontoon is considerably less than for the catamaran.

It's assumed that each semi-submersible pontoon will have 4 columns and A_{col} of each pontoon is: $A_{col} = 4 \cdot 4 = 16\ [m^2]$

The total water plan area of this pontoon is

$$A_p = 16 \cdot 4 = 64m^2$$

$$F_{dyn} = 64 \cdot 2{,}1 \cdot 0{,}57 \cdot \frac{1031}{1000} \cdot 9{,}81 \cdot 2 = 1549\ kN\ (or\ 158\ Ton)$$

The loads in each situation is:

Design situation	Determinative Load	Units
Situation 1	2113	[ton]
Situation 2	2830	[ton]
Situation 3	2263	[ton]

Table 5 Forces on Semi-submersible pontoon

Again is situation 2 determinative. The needed capacity of each floater is:

$$Floating\ capacity\ one\ floater = \frac{2830}{4} = 708\ m^3$$

This gives the following pontoon dimensions.

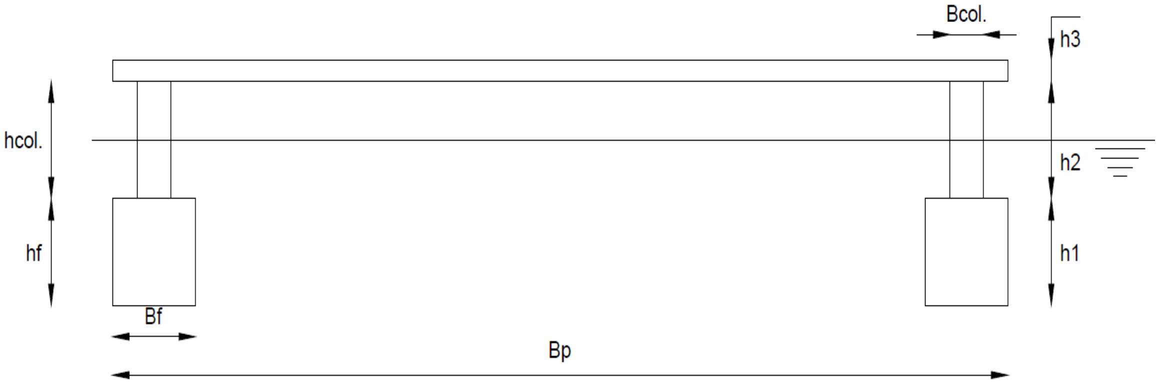


Figure 11 Semi-submersible pontoon

Dimension	Value	Units
With pontoon B_p	54	m
Length pontoon L_p	40	m
Height pontoon h_p	10.5	m
With floater B_f	4	m
Length floater L_f	40	m
Height floater h_f	4	m
With pontoon deck B_D	54	m
Length pontoon deck L_D	30	m
Height pontoon deck h_d	1	m
Columns	(L·B·H) → (4 x 4 x 5.5)	m^3

Table 6 Overall dimensions of the Semi-submersible pontoon

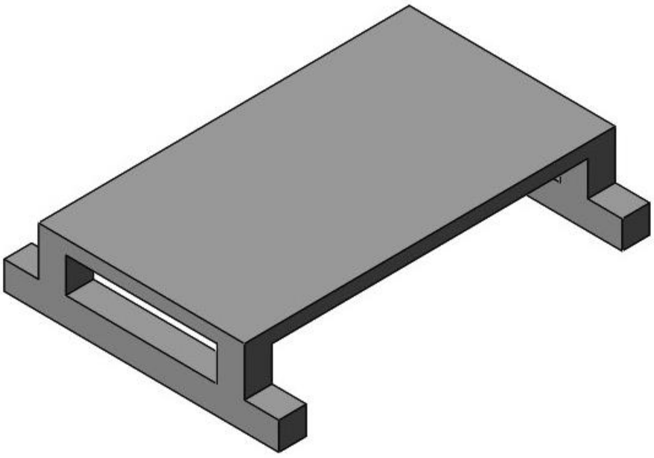


Figure 12 3D sketch of the semi-submersible pontoon

1.6 LIMITING CONDITIONS

The limiting conditions are determined by the loading regime and the functional requirements. If these conditions are exceeded then the tunnel element cannot be immersed safely. It should be noted that a qualitative and quantitative application of the expected loading regime is a fundamental prerequisite for the analysis and the design of the system. For the immersion system consisting mainly from tunnel element and the two pontoons the hydrostatic, dynamic loadings and the responses in different degrees of freedom are all interlinked and they cannot be considered separately. Nevertheless in this analysis they are separately treated so far. In this part of the analysis the limiting condition will be determined in such a way that they will interlink the different loading regimes and responses.

For the analysis only the floating conditions for the three structures (tunnel element and two floating pontoons) are considered. The physical parameters such as wave load and current load which govern the response of each of the three floating structure have a wide range of values. Choices should be made what are the acceptable conditions, in which the system can operate and what is economically optimal. It should be noted that the economically optimization is not considered in this analysis. But the results presented here could be used as input for the economically optimization. Also the different load types make an important contribution to the overall response of the system. Responses at higher and lower frequencies are likely to occur. It should be analyzed what are the conditions to minimize this responses. The system component could be designed such that is more robust in different environmental conditions, or the workable conditions should be chosen such that the responses are in acceptable range.

In order to be able to analyze the dynamical behavior and the workability of the two different types of the pontoons some limiting conditions are determined for both pontoon types. The limiting conditions are subdivided in two types. Type 1 are the operational conditions, if they are exceed the system cannot operate properly and the operation should be interrupted. This kind of limiting conditions are linked to safety and operability of the personnel and equipment on the pontoons. Type 2 operational conditions are linked to the failure of the system. If this type conditions are exceeded than the operation will be failed. This type of conditions are linked to the structural capacity of the components.

From analytical point of view the total load working on the system is subdivided in two components, namely: static and dynamic load components. By so far dividing these two components and interlinking them, the main dimensions of the pontoons and the amount of the required ballast has been determined. The required amount ballast on its turn determine also the pretension in the cables during the immersion

Gravity load or better said the weight of the different components have resulted in the main dimensions of the pontoons. By lacking of further information, a payload (the weight of the equipment) of 400 ton has been assumed. The hydrostatic load is equally important as the gravity load. By choosing the proper dimensions for the pontoons the hydrostatic load is equal to the gravity load. On other hand the distribution between the gravity and hydrostatic loads determine the static stability. And in previous section we have already seen that the static stability on its turn is sensitive for the force fluctuations in the suspension cables. This all indicated the interlinked character of the design parameters.

There are many complicated relationships between the different components. In this analysis, only the components which determine the dynamic behavior of the system are observed. Figure 13 describes the analysis scheme. The approach described in this scheme has been used to determine the dimensions of the immersion rigs and the cable dimensions. The method which has been used for the determining the dimension of the different components of the immersion system is a deterministic and a semi probabilistic which controls that the required strength has to be greater than the expected load in a certain period of time. This scheme could be also used as design check.

In this analysis the failure mechanism is described as:

- A way in which the pontoon structure is no longer able to fulfil its function. (guide the tunnel element during the immersion process → not having enough floating capacity to carry the weight of the TE and other temporary or permanent equipment).

- Also the operation will be failed when one of the suspension cables will be broken.
- Or the movements of the system will exceed the allowable values which will result in the failure of the system.

Not being able to fulfil its function relates to a temporary or a permanent situation in which the tunnel element can't be immersed on a safe way. The consequence of this situation will be that the tunnel element will be damaged or the operation should be interrupted

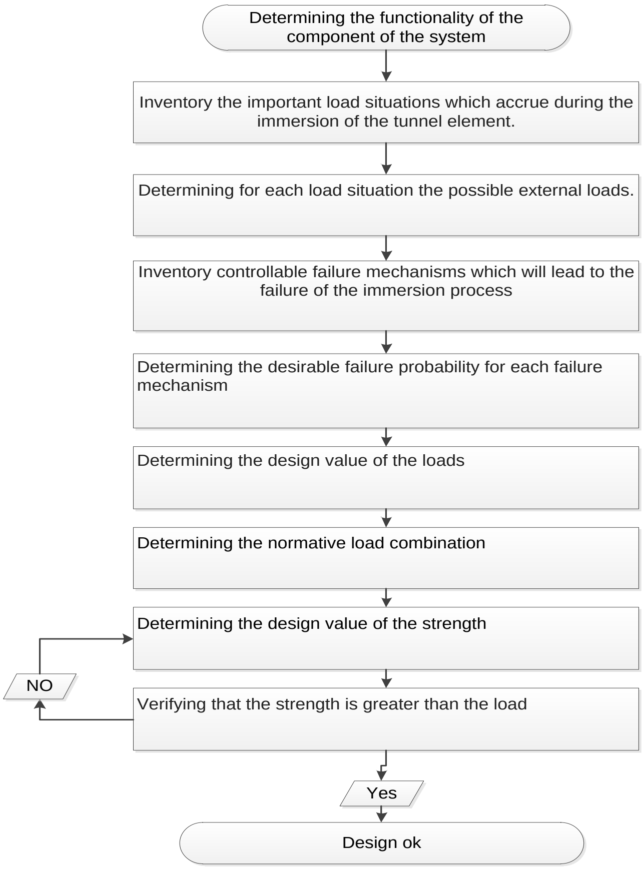

Determining the functionality of the component of the system
Inventory the important load situations which accrue during the immersion of the tunnel element.
Determining for each load situation the possible external loads.
Inventory controllable failure mechanisms which will lead to the failure of the immersion process
Determining the desirable failure probability for each failure mechanism
Determining the design value of the loads
Determining the normative load combination
Determining the design value of the strength
NO
Verifying that the strength is greater than the load
Yes
Design ok

Figure 13 Design scheme followed in this analysis

1.6.1 LIMIT STATES

Limit states are conditions which appears before the failure mechanism. For the floating system (the pontoons and the tunnel element) two limit states are distinguished.

- Ultimate limit state (ULS)
- Serviceability limit state (SLS)

The above two mentioned limit states linked with design situations. The tunnel element can be immersed as long as the operational limits are not exceeded. The operational limits are related to:

a) Allowable line tensions (suspension cables and mooing lines)
b) Capacity of the winches
c) Allowable motional envelopes
d) Floating capacity of the pontoons

Alternatively the operational limits can also subdivided into two types of limits:

a) Structural limits (exeedance of this kind of limits will result in damage to one of the components in the system or the system as whole) → ULS
b) Availability limits (exeedance of this kind of limits will result in interruption of the immersion process) → SLS

Serviceability limit state (SLS):

The Serviceability limit states are the boundary conditions relating to the functioning of the structure or parts thereof under normal use. The serviceability limit in the modelled case here is described as the disruption of the immersion process. As mentioned before the exeedance of this kind of limits will result in interruption of the immersion process. This is translated into the motional behaviour of the system. The motional degrees of freedom are given in Figure 14.

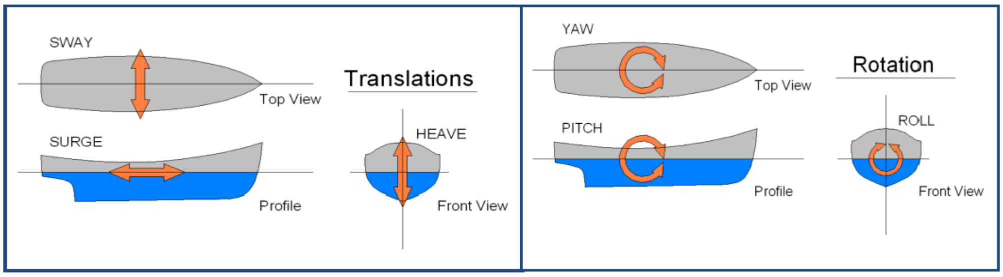


Figure 14 six degrees of freedom of a floating structure

For the SLS the limiting conditions are defined such if one of motional amplitudes is exceeding the maximum allowable value, the immersion operation has to be interrupted. Or when it appears in the analysis that in given environmental condition the exceedance is likely to occure then those conditions will be classified as not workable conditions.

In the SLS velocities and accelerations of tunnel element and pontoons are important parameters. In the SLS the comfort for the personnel and the operability of the system is considered to be determinative. For the comfort of the personnel and for functioning of the electrical devises on the pontoon some general operability limiting criteria for ships has been used to check the conditions in SLS. The criteria are copied from [J.M.J. Journée and W.W. Massie; OFFSHORE HYDRODYNAMICS; January 2001]. The values are given in . This limits can be applied as motional constrains in the workability analysis of the pontoons.

	Motion/acceleration
Maximum roll angle	3°
Maximum pitch angle	3°
Surge (acceleration/velocity)	0.05g
Sway (acceleration/velocity)	0.05g
Heave (acceleration/velocity)	0.10g

Table 7 Limiting conditions in SLS for the pontoons

Ultimate limit state (ULS):

In general the boundary conditions in ULS are related to the safety of persons and/or the safety of the structure. In this analysis only the structural safety is considered. When the ULS conditions are exceeded it indicates that the immersion operation will be failed. The result of this kind of exceedance is much more dangerous than exceedance of the SLS limit conditions.

The exceedance of this type of boundary conditions indicates that the tunnel system or a component will undertake serious damage. The ULS conditions for this case study are translated to the following:

- The floating capacity of one of the pontoons will be less than required. This will lead that the pontoon will sink.
- The pontoon will be capsized during the immersion (statically not stable)
- The suspension cables will be slackening due to the dynamic wave force (zero force in one of the suspension cables).
- Allowable line tensions will be exceeded (suspension cables, mooring lines and contraction cables)
- The force in the cables will be more that the capacity of the winch, which will lead to the failure of the winch.

– The motions and the related velocities of the tunnel element will exceed the allowable values which will cause that the guide beam/catch will be damaged. Or that the tunnel element will clash to the already installed tunnel element.

Some of the above mentioned boundary conditions will be explained here below. It is obvious that the suspension cable will be slacken when the total force in the cables is zero. If this happens than the pontoons will rise up and the tunnel element will become uncontrollable.

1.6.2 CAPACITY OF THE MOORING SYSTEM

The function of the mooring lines is to keep the pontoons in position. For determination of the mooring forces the second order wave forces and the current drag force has to be taken into account. The mooring lines are modeled as taut mooring lines.

Figure 15 indicates the main configuration for mooring lines. The mooring lines are given in red and the contraction wires are given in blue. The angle of the mooring lines with the horizontal plane is assumed to be Ɣ=10°. And for the calculation of the capacity of the contraction wires ß is assumed to be 15°.

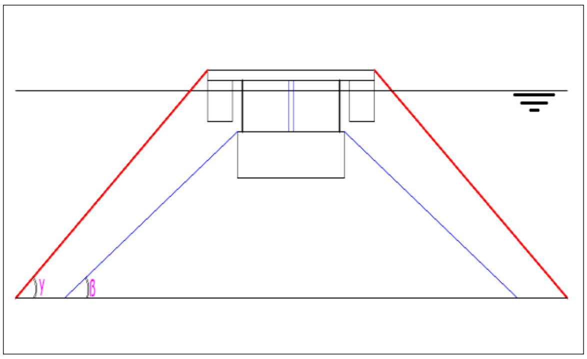


Figure 15 Mooring line configuration for transversal direction

The capacity of the mooring system is mainly determined by the horizontal drag force and wave Drift force. The calculation principle is the same as it is given for the tunnel element in chapter 5. For the calculations the width of one floater has been considered as the width of the structure. Again the skin friction is neglected, the approximate value of the skin friction is expected to be less than 5% of the total drag force. During the immersion the pontoons will have an angle of 90° with the current flow which will lead to an area perpendicular to the flow direction A_c of (L_f·T).

$$F_{mooring} = F_{Drift} + F_{drag} \tag{31}$$

The results of the calculated drag force for both types pontoons are given in Table 8.

Type Pontoon	ρ_w [kg/m^3]	A_c [m^2]	C_D	λ	$Ɣ_Q$	V_r [m/s]	F_D [kN]
Catamaran	1031	190	2.0	0.6	1.55	2.5	1140
Semi-Submersible	1031	165	2.0	0.6	1.55	2.5	990

Table 8 Drag Force on the two types Pontoons during immersion

Wave drift forces on the pontoons are also calculated according the same principle as for the tunnel element. For the calculation beam waves (angle 90°) are assumed. For the calculations significant wave height (Hs=0.57m) in the middle of the water corridor has been used. With exceedance probability of 5% the design wave height can be expressed as H_d=2.11 ·$H_{s.}$ The calculated wave drift forces for the two types of pontoons are given Table 9.

Type Pontoon	ρ_w [kg/m^3]	L_f [m]	h_f or T [m]	H_D [m]	H_s [m]	F_{Drift} [kN]
Catamaran	1031	38	5	2.11·H_s	0.57	174
Semi-Submersible	1031	40	4	2.11·H_s	0.55	147

Table 9 Wave Drift Force on the two types Pontoons during immersion

For the determination of the total mooring capacity of each pontoon the total wave drift force and the total drag forces are summed up. Each pontoon will have 4 mooring lines. The capacity of each mooring line is the total force divided by 4. The winches should also have the same capacity as the mooring lines. The result are summarized in Table 10 for both pontoon types.

Type Pontoon	Total Force	Capacity mooring line [kN]	Capacity mooring winch [kN]
Catamaran	1314	365	365
Semi-Submersible	1137	315	315

Table 10 Capacity mooring system

1.6.3 RESONANCE OF PONTOONS DUE TO CURRENT EXCITATION

Also the vortex shedding period of the pontoon has been determined by the same principle as explained for the tunnel elements in chapter 5. It appears the smallest vortex shedding period of the pontoon is approximately 19.5 s for the current velocities of 2.5 [m/s]. For the increasing flow velocity the vortex shedding period decreases exponentially.

For the vortex shedding period it appears that the width of one floater is determinative resulting in smaller Vortex shedding period. That’s why for the determination of the Strouhal number and calculation of the vortex shedding period the width of one floater has been used. The results are given Figure 16, the red line indicated the upper bound of the Strouhal number resulting in higher vortex shedding periods and the purple line indicates the lower bound representing lower vortex shedding period. It is more likely that the actual periods will be in between the given lines. But as limiting condition the purple line can be maintained.

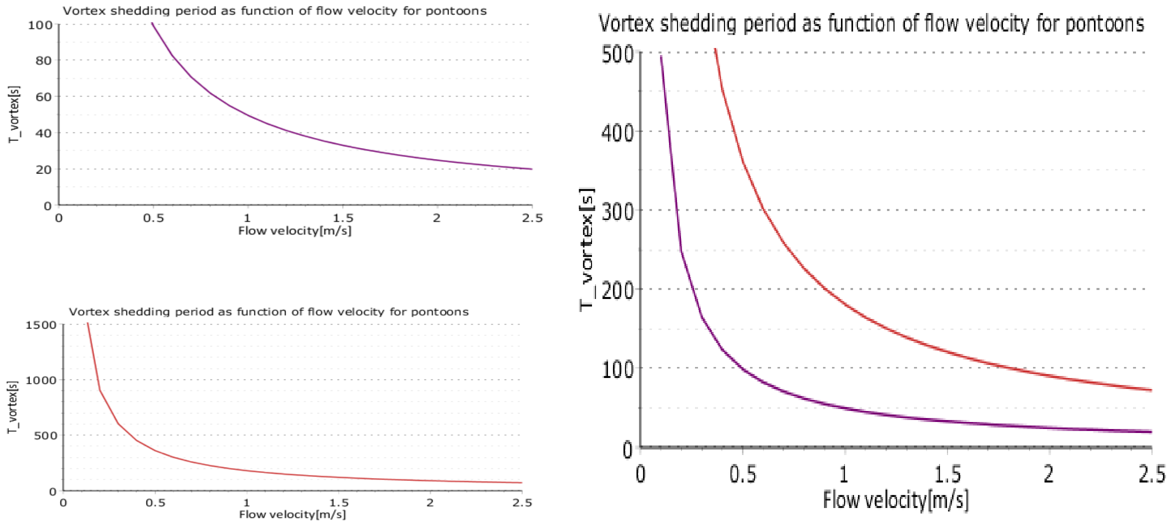


Figure 16 Vortex Shedding Period of the pontoons

1.6.4 CAPACITY CONTRACTION LINE

The function of the contraction lines is to control the motions of the tunnel element during the immersion. The contraction lines are connected to the pontoons, tunnel element and sea bottom. The contraction lines guided from pontoons through the pad eyes on the tunnel element and fixed to the bottom of the sea. The main function is to prevent undesired transversal motions of the tunnel element during the immersion.

The transversal motion of the tunnel element during the immersion will be caused by the current drag force and the wave drift forces.

The angle of the tunnel element during the immersion with current flow and wave attack is assumed to be 90°. The design valued of the current drag force and the wave drift forces can be expressed as:

$$F_{drag} = C_D \cdot \frac{1}{2} \cdot \rho_w \cdot V_r^2 \cdot A_c \cdot \gamma_Q \qquad (32)$$

$$\overset{\equiv}{F}_{drift} = \frac{1}{16} \cdot \rho \cdot g \cdot L_{TE} \cdot H_d^2 \cdot H_{TE} \quad [kN] \qquad (33)$$

The values calculated with the above mentioned expressions are :

- Drag force F_{drag} = 1502 [kN]
- Wave Drift force F_{drift} = 3276 [kN]

The total transversal force on the tunnel element is the sum of two mentioned forces. The transversal force on the tunnel element during the immersion is: $F_{transversal}$ =4252 [kN]

During the immersion 4 contraction wires should be used, 2 from each pontoon. The capacity of each wire should be at least 819 [kN]. For the calculations of the total drift force is H_d used instead of the H_s. This means that the exceedance probability of the force is 5 % and for the calculations of the drag force a load factor of 1.55 is applied to calculate the design value. The final results are given Table 11.

	Capacity contraction wire	Capacity winch
Transversal forces on tunnel element	850 [kN]	850 [kN]

Table 11 Capacity contraction mooring system

1.6.5 CAPACITY SUSPENSION CABLES

The suspension cables should carry the extra weight of the tunnel element during the immersion. The strength of these cables is of essential importance. Because if one of the suspension cables will break, then the tunnel element will hang unstable in the remaining cables. It is even possible that the pontoons will capsize and the whole immersion operation will fail resulting in great damage. The extra weight which should be carried by the cables is different for the two pontoons. Two configurations are analysed for the stability calculations.

Namely a pontoon with two suspension cables resulting in that the tunnel element will hang on 4 cables from two pontoons and a configuration with 4 cables from one pontoon resulting in 8 cables from two pontoons. By the determination of the pontoons dimensions the following values are calculated as the weight which should be carried by the suspension cables.

	Total extra weight [ton]	Force per cable (2 cables per pontoon) [ton]	Force per cable (4 cables per pontoon) [ton]
Catamaran	984	492	246
Semi-Submersible	480	240	120

Table 12 Force in suspension cables

During the immersion due to motions of the pontoons and the tunnel element the forces in the cables will change. The force fluctuations will be determined in the dynamic analysis. For the design purpose the maximal fluctuation for the 4 cable system assumed to be 40 ton per cable and 80 for the 2 cable system. Therefore the capacity of the system should at least must be:

	Total capacity of the system [kN]	Force per cable (2 cables per pontoon) [kN]	Force per cable (4 cables per pontoon) [kN]
Catamaran	$1{,}12 \cdot 10^4$	$5{,}6 \cdot 10^3$	$2{,}8 \cdot 10^3$
Semi-Submersible	$6{,}27 \cdot 10^3$	$3{,}14 \cdot 10^3$	$1{,}57 \cdot 10^3$

Table 13 Capacity suspension cable system

1.6.6 TYPE MOORING LINES:

In order to analyse the workability the main dimensions of the mooring system and the characteristics should be known. Different mooring types are possible to apply for the station keeping of the system. Traditionally the taut spread moorings are used during the immersion. It is possible to use different materials for mooring lines as well different combinations of mooring lines. For the analysis purposes it is assumed that for the mooring lines, contraction wires and for suspension cables wire rope will be used.

Three rope types are considered which can be used, namely, six strand, spiral strand and multi strand. For mobile floating structures the six strand independent wire rope core (IWRC) is most commonly used due to its lateral flexibility and relative cheapness. The spiral strand has greater longitudinal stiffness, torque balance and lower spinning loss. It is more suitable for long term installation. For example it is more commonly used in offshore industry for the floating production systems. Multi strand is not commonly used in the offshore conditions. For the analysis it is assumed that all the cables will be six strand.

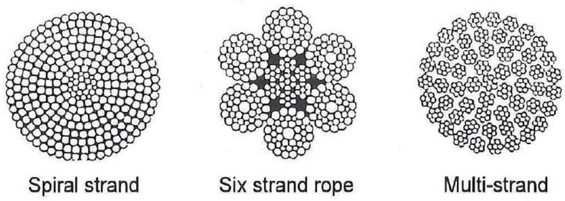


Figure 17 Typical wire ropes (Floating Structures:, 1998)

1.6.7 WEIGHT AND STIFFNESS

The weight and stiffness depend not only on type of the wire rope but it is also dependent on the manufacturer and the fabrication process and the pre-stretching of the rope during the fabrication process. Thus after selecting the rope type and the manufacturer the precise values can be obtained. For the analysis purposes only estimated values given in Table 14 are used. It should be mentioned that these values are applicable for the new steel wire ropes. After being used the wire rope tends to decrease its stiffness with age. Thus there are some uncertainties by using these values. For the final design purposes the precise data should be gathered and used instead of the given values in Table 14.

Construction	Submerged weight/length	Stiffness/length (A·E)
Six strand (IWRC)	$0.034 \cdot d^2 [N] (\text{d in mm})$	$45000 \cdot d^2 [N] (\text{d in mm})$
Spiral strand	$0.043 \cdot d^2 [N] (\text{d in mm})$	$90000 \cdot d^2 [N] (\text{d in mm})$

Table 14 Wire rope weight and stiffness properties (© Centre for Marine and Petroleum Technology)

The steel used for the wire rope applications has a very high strength. The breaking strength of the wire rope is dependent on the fabrication process as well as the grade of steel used. After ordering specific type of wire rope the final breaking strength will be tested by the manufacturer and a certificate will be provided. The test certificate will provide the accurate strength of the rope. For the offshore industry API-Spec 9A provides a specification for the minimum breaking strength of the wire ropes. These values can be used as conservative estimate of the breaking load. For the estimation of the required diameter of the cables the values in Table 15 are used.

Type	Ultimate Tensile Stress $[N/mm^2]$	Breaking Strength $[N]$
Six strand (IWRC)	1770	$525 \cdot d^2 \ (d \ in \ mm)$
Six strand (IWRC)	1860	$600 \cdot d^2 \ (d \ in \ mm)$
Spiral strand	1570	$900 \cdot d^2 \ (d \ in \ mm)$

Table 15 Breaking strength cable

It is assumed that the ultimate tensile strength of the cables will be 1860 $[\mathbf{N/mm^2}]$ and the six strand (IWRC) wire ropes will be applied for all the three types of cables. The required capacity of the cables has already been determined. The following diameters has been determined for the three different cables used in the system.

	Diameter mooring line [mm]	Diameter contraction wire [mm]	Diameter suspension cable [mm]
Catamaran	25	38	97
Semi-Submersible	23	38	72

Table 16 Cable diameters (pontoon configuration with 2 suspension cables)

	Diameter mooring line [mm]	Diameter contraction wire [mm]	Diameter suspension cable [mm]
Catamaran	25	38	68
Semi-Submersible	23	38	51

Table 17 Cable diameters (pontoon configuration with 4 suspension cables)

The elongation of the wire rope is not linear with the stress, therefor the wire rope extension curve is not linear and the elasticity modulus is not constant. It depends on the tensile stress. Furthermore, a distinction is made between permanent extension, elastic stretch and extending through wear. For the calculated cable dimension it should be checked if the cable stress remains in the elastic range for relevant design situations. And the maximal allowed stretch has been determined for the calculations of the maximal force in the cables.

In order to check if the stresses remains in the elastic range the procedure described in (NEN-EN 1993-1-11+C1:2011) has been applied. In the ultimate limit state the ratio between the design value of the axial force and the design value of the tension resistance should be smaller than 1 “equation (34)”.

The design value of the tension resistance of the wire rope is expressed by equation (35).

The value of the partial factor is dependent on the measures applied at the ends to reduce the bending stiffness. If no measures are taken into account, then the value is 1.0. For the calculation it is assumed that no measures are applied. The design value of the breaking strength is given by equation (36).

The value for f is determined from table 2.2 of the (NEN-EN 1993-1-11+C1:2011) and it is equal to 0.56 then the value of K is calculated as 0.44.

The characteristic value of the breaking stress are expressed by (39).

The stress limit for the ULS is given by equation (41).

For the short term situations the product of the partial factor is equal to 1.10. For the SLS the limit stress is expressed by equation (42).

For the SLS the product of the partial factor is equal to 1.48, then the limit stress for the SLS can be expressed as in equation (43).

$$\frac{F_{Ed}}{F_{Rd}} \leq 1 \qquad (34)$$

$$F_{Rd} = min\left\{\frac{F_{uk}}{1.5 \cdot \gamma_R}; \frac{F_k}{\gamma_R}\right\} \qquad (35)$$

$$F_{uk} = F_{min} \cdot k_e \qquad (36)$$

$$F_{min} = \frac{K \cdot d^2 \cdot R_r}{1000} \quad [kN] \qquad (37)$$

$$K = \frac{\pi \cdot f \cdot k_e}{4} \qquad (38)$$

$$\sigma_{uk} = \frac{F_{uk}}{A_m} \qquad (39)$$

$$A_m = \frac{1}{4}\pi d^2 \cdot f \qquad (40)$$

$$f_{const} = \frac{0.66 \cdot \sigma_{uk}}{\gamma_R \cdot \gamma_F} \qquad (41)$$

$$f_{SLS} = \frac{0.66 \cdot \sigma_{uk}}{\gamma_R \cdot \gamma_F} \qquad (42)$$

$$f_{SLS} = 0.45 \cdot \sigma_{uk} \qquad (43)$$

Where	
F_{Ed}	design value of the axial rope force
F_{Rd}	design value of the tension resistance
F_{uk}	characteristic value of the breaking strength
F_k	characteristic value of the proof strength of the tension component
γ_R	partial factor.
F_{min}	minimum breaking force factor taking account of the spinning loss
k_e	spinning loss factor
K	minimum breaking force factor taking account of the spinning loss
d	nominal diameter of the cable
R_r	rope grade in $[\mathrm{N/mm^2}]$
f	fill factor

Based on the above given principle, calculation has been performed for all the three types of cables and two different types of the pontoons. The results are given in Table 18. In order to specify the parameters for the two types pontoons, the subscript C and S are used for Catamaran pontoon and subsequently for the Semi-submersible pontoon. From the calculations it was clear that the chosen diameters for the mooring lines and contractions wires did not met the requirement of the unity check. Also it did not have any extra capacity for the force fluctuation due to motions of the system. That is why the chosen diameter has been adjusted such that it met the requirements for the unity check and that it would have some extra capacity for the force fluctuation due to motions of the pontoon. Also the stretch of the cables for in the elastic region for the two limit states has been calculated. For the calculations a value of $105 \cdot 10^3$ has been used for the modulus of elasticity E. Also the maximum allowable elongation of the cables has been calculated.

Parameter	Mooring line	Contraction cable	Suspension cable (2 cables)	Suspension cable (4 cables)
d_C	30	45	100	75
d_S	30	45	75	55
A_{m-C}	396	891	4398	2474
A_{m-S}	396	891	2474	1331
F_{min-C}	736	1656	8180	4602
F_{min-S}	736	1656	4602	2475
F_{Ed-C}	365	850	4827	2413
F_{Ed-S}	315	850	2354	1177
F_{Rd-C}	491	1104	5454	3068
F_{Rd-S}	491	1101	3068	1650
Unity check$_{-C}$	0.7436	0.7697	0.8850	0.7867
Unity check$_{-S}$	0.6412	0.7697	0.7674	0.7114
σ_{uk-C}	1860			
f_{const}	1116			
f_{SLS}	837			
ε_{const}	0.01062857			
ε_{SLS}	0.00797143			
Δl_{C-ULS}	2.3236	1.19	0.06377	0.06377
Δl_{S-ULS}	2.4921	1.19	0.08503	0.08503
Δl_{C-SLS}	1.7427	0.893	0.0478	0.0478
Δl_{S-SLS}	1.8690	0.893	0.06377	0.06378

Table 18 Cable Forces, cable dimensions, cable stretch and design check

1.6.8 STRUCTURAL LIMITING CONDITIONS:

The total operability of the immersion process is based on the limited motion and the capacity of the system and the duration of the immersion process. Regardless the pontoon configuration the limits are applied for both pontoon types. If in specific environmental conditions the limiting motions are exceeded, then the immersion process has to be interrupted. In the practice it means, before the immersion process is initiated the hydraulic conditions has to be checked. And if the limiting conditions are exceeded than no immersion will take place in those conditions.

So far the main parameters for the dynamic analysis has been determined. In order to determine the workability and to compare it for the two pontoons, structural limiting conditions will be determined for the analysis purposes. First the maximum allowable forces in the cables will be determined. Subsequently the allowable forces will be translated to the maximum allowable transversal motions and rotations of the pontoons. Off course for the detailed design more parameters has to be determined. But for the purpose of this analysis the required parameters are sufficient to perform the dynamic analysis and check the workability. The main dynamical characteristics of the system and the dynamic analysis will be described in the next part of the report. The focus of this part is to determine the maximum allowable motions of the system for the chosen structural dimensions. The workability of the system will be analized in beam waves. Therefore the main motion of the system will occur in three degrees of freedom, namely:

- Sway
- Heave
- Roll

For the derivation of maximal elongation is the same principle has been used as for derivation of the stiffness matrix. The maximum elongation of a cable for the two pontoon types is given in Table 18. The maximum force fluctuation in a cable due to motion in ith direction can be expressed as:

$$dT_i = \frac{EA \cdot \Delta l_i}{l} \tag{44}$$

Where dT_i represents the force fluctuation in the cable and *l* is the length of the cable. By substituting the values from Table 18 in the above mentioned equation the maximum force in elastic deformation region can be determined for each cable. The calculated values are presented in Table 19. Distinction is been made between the ULS and SLS. The upper two rows of the table indicate the maximum forces for the ULS and the lower two rows indicates the values for the SLS.

Parameter	Mooring line [kN]	Contraction cable [kN]	Suspension cable [kN]	Suspension cable (4 cables) [kN]
$\mathbf{dT_{i\text{-}C\text{-}ULS}}$	441.6	994	4908.4	2761
$\mathbf{dT_{i\text{-}S\text{-}ULS}}$	473.8	994	3681.3	1484.8
$\mathbf{dT_{i\text{-}C\text{-}SLS}}$	331.3	745	3681.3	2070.7
$\mathbf{dT_{i\text{-}S\text{-}SLS}}$	331.3	745	2070.7	1113.6

Table 19 Maximum allowable Cable Forces (for ULS and SLS)

The maximum forces can be translated to the maximum displacements and rotations by determining the force and displacement relationship. In the dynamic analysis effect of the mooring lines and contraction wires will be not taken into account. That's why the maximal allowable forces in the suspension cables are translated in the maximal motions of the pontoons. For the determination of the maximum displacement, the following relations are used for the motions in sway and roll (see also Figure 106 and Figure 108 of the main report). For heave the maximum displacement is equal to the maximum elongation (see also Figure 107 of the main report).

$$dT_2 = \left(\sqrt{x_2^2 + l^2} - l\right) * \frac{AE}{l} \tag{45}$$

$$dT_4 = \frac{AE}{l} \cdot \left(\frac{B_p}{2} - B_f - tolerance - \frac{W_1}{2}\right) \cdot \cos(\varphi) \cdot \varphi \tag{46}$$

Where	
dT_2	Force Fluctuation in each cable due to sway motion
x_2	Displacement in sway degree of freedom
l	Length of the suspension cable
AE	Axial stiffness of the cable
B_p	Width of the pontoon
B_f	Width of the floater
$tolerance$	Transversal distance between the pontoon and the tunnel element(=2m)
W_1	Wall thickness of the tunnel element of the outer wall
φ	Rotation in Roll degree of freedom.

The maximal motion of the system are given in Table 20 for the ULS for two or 4 suspension cables applied. The SLS condition are not considered further because these conditions are meant to limit strains such that the corrosions control measures, cracking of the sheaths and hard fillers are not damaged. Due to temporary character of the immersion operation it is most probable that these conditions are not limitative for this operation. The subscript 2,3,4 in the results indicated the motions in sway, heave and roll degrees of freedom. Again the subscripts C and S indicate the Catamaran and Semi-Submersible pontoons.

Limit Motions	Value when 2 suspension cable are applied		Valued when 4 suspension cables are applied	
		Catamaran pontoon		
$\mathbf{x_{2\text{-}C}}$	0.8771	[m]	0.7693	[m]
$\mathbf{x_{3\text{-}C}}$	0.06377	[m]	0.0638	[m]
$\mathbf{x_{4\text{-}C}}$	0.003149	[rad]	0.003149	[rad]
		Semi-submersible pontoon		
$\mathbf{x_{2\text{-}S}}$	1.3516	[m]	1.1695	[m]
$\mathbf{x_{3\text{-}S}}$	0.08503	[m]	0.08503	[m]
$\mathbf{x_{4\text{-}S}}$	0.00556	[rad]	0.00420	[rad]

Table 20 Limit motions pontoons for the ULS

2 BOUNDARY CONDIIONS

In this part the boundary conditions will be descried. The data present here below have been derived from the investigations carried out by the owner of the project. For the calculations in this master thesis relevant threshold values will be used. The provided environmental data by the client could be divided into two groups.

- The operational conditions (to support the planning of the work)
- The design conditions with a return period of 1 to 10000 years. (to support the design purposes)

Science the scope of this thesis work is the dynamical behavior of the immersion system during the immersion process, only the normal conditions will be used for the modeling purposes. The relevant data for the immersion activities are:

- Wave data
- Wind speed data
- Water levels
- Temperature
- Precipitation
- Visibility
- Sea ice

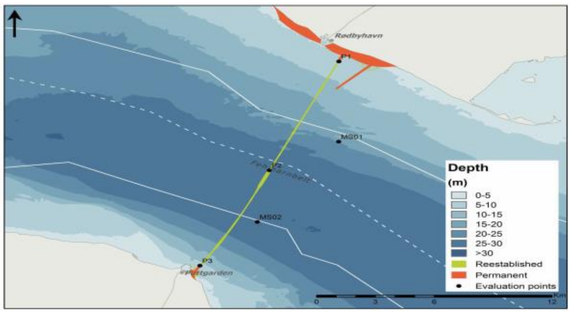


Figure 18 Location of the measurement points

As it will be described in the description of the model only the effect of the waves on the immersion system is taken into account. That’s why only the boundary condition in relation with wave data will be described here. Also the water level and the salinity condition are explained here.

The data presented by the owner are the results of a 18 years study provided by the client. The study was elaborated from February 1994 till February 2012. Additionally long term observations and measurements were adopted to support the establishment of the extreme and normal conditions. The weather conditions were established across the alignment of the fixed link or in the area to get sufficient and reliable data. The measurements were performed within 5 positions in the Fehmarnbelt (for more detailed description and explanation see also Ref. 12). The position of the measurement points and the coordinates are shown in Figure 18 and Table 21. The of the reference points represents the condition across the whole length of the corridor.

Name	Easting [mUTM32]	Northing [mUTM32]	Water Depth [mDVR90]
P1 (near Rødbyhavn)	652209	6057172	7.1
P2 (middle of corridor)	648656	6049152	28.5
P3 (near Puttgarden)	645109	6042126	8.0
MS01	652199	6051263	
MS02	648039	6045345	

Table 21 Coordinates of the evaluation points

The vertical reference system which is been used is DVR90;

“The national height reference system DVR90 (Danish Vertical Reference 1990)is used as standard reference for heights above mean sea level in Denmark. This system was built upon a precision levelling survey conducted from 1982 to 1994. It uses the 1990 mean sea level measured by the Danish Meteorological Institute's sea level gauges as reference.” (Kort og Matrikelstyrelsen, 2012).

2.1 WATER LEVELS

The water level in the project area is variable due to many factors, such as: (signal generated by the local wind, tides, rotation modification of the flow and long term variability affected by changes in the depth averaged density) From the analysis of the water levels it could be concluded that there is a clear seasonal variability in the water levels. In the summer months is the variation small and in the fall and winter months is the variation large. InFigure 19 and Table 22are the monthly variation of water level in the project area are depicted.

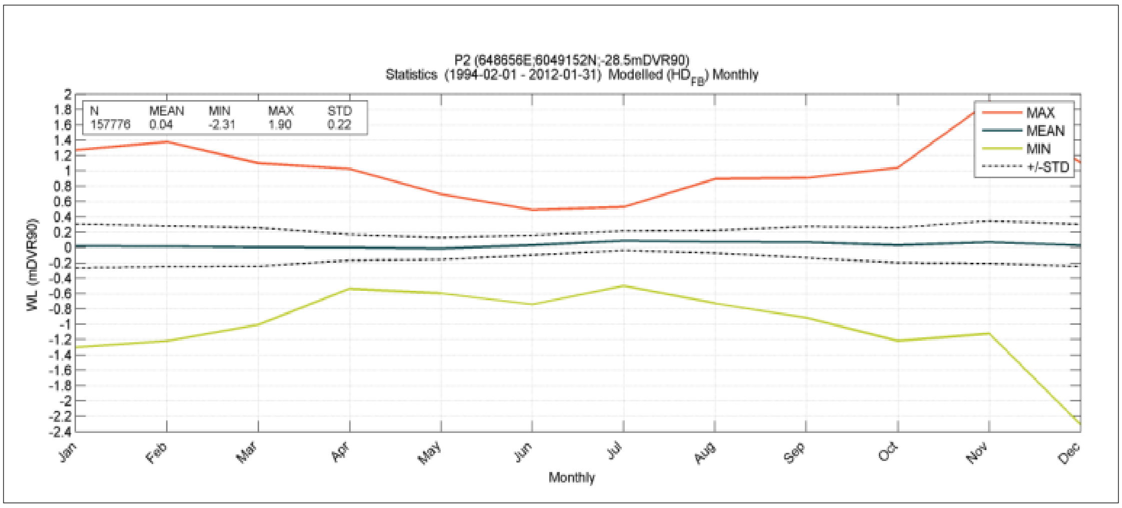


Figure 19 monthly water level variation

	All	Jan	Feb	Mar	Apr	May	Jun	June	Aug	Sep	Oct	Nov	Dec
Mean	0.04	0.02	0.02	0.01	0.00	-0.01	0.03	0.09	0.08	0.07	0.03	0.07	0.03
Max	1.90	1.27	1.38	1.10	1.03	0.70	0.49	0.53	0.90	0.91	1.04	1.90	1.10
Min	-2.30	-1.30	-1.20	-1.00	-0.65	-0.60	-.075	-0.45	-0.70	-0.90	-1.20	-1.10	-2.30

Table 22 monthly water level variation

2.2 WAVES

Waves in the project area are primarily governed by the local wind conditions and the limited fetches. The wave climate at Fehmarnbelt could occasionally be affected by the waves from the Baltic sea (south eastern: Akrona Basin). Waves in the project area are primarily governed by the local wind conditions. The fetches are limited by the surrounding lands. In general the wave climate can be considered as a mild.

From the analyzed data it can be concluded that the highest waves occur in the middle of the corridor (P2). The mean wave height near the shore is lower. The mean wave heights near the German(P3) and Danish(P1) shores are approximately 35% and 15% lower than the mean wave height in the middle (P2). The maximum wave height measured during the 18 years measurement study is 3.6 m which occurred during the severe storm in December 1999 in combination with extreme wind speed (27.2 m/s). The wave data was established by numerical modeling using the MIKE 21 Spectral Wave model by DHI. The wave model covered the Fehmarnbelt area and adjacent waters, and the data were validated against available local observations including at MS01, MS02 and Nysted. The wave data include integral wave parameters of significant wave height (H_{m0}), peak and mean wave periods (T_p, T_{01}, T_{02}, T_{-10}) and peak and mean wave directions (PWD, MWD) as well as full wave spectra. The annual statistic of the omni-directional waves are presented in Table 23 . The highest wave occur in the middle of the fixed link were the fetches and water depths are the biggest.

Position	Significant wave height, Hm0 (m)	Spectral peak wave period, Tp (s)	Mean wave period,T02 (s)
	min/mean±std/max	min/mean±std/max	min/mean±std/max
P1 (near Rødby)	<0.1/0.49±0.36/2.90	1.01/3.41±1.08/8.51	0.80/2.28±0.72/5.61
P2 (Middle of the fixed corridor)	<0.1/0.57±0.40/3.58	1.01/3.44±1.01/7.38	0.81/2.42±0.73/5.13
P3 (near Puttgarden)	<0.1/0.38±0.27/2.21	1.01/3.21±1.01/9.27	0.82/2.02±0.0.56/4.67
MS01	<0.1/0.57±0.40/3.56	1.01/3.41±1.01/7.00	0.81/2.42±0.73/5.10
MS02	<0.1/0.53±0.37/3.09	1.01/3.42±0.99/8.40	0.82/2.35±0.68/4.74

Table 23 Basic Statistics of wave model data

Swells

Occurrence of some swell components have been measured with wave periods larger than, say, 4s. However the wave height is that small ($\boldsymbol{H_{swell} < 01 - 02m}$) that it can be considered as negligible.

Wave direction

The directional information of incident waves is been provided for the 5 measurement points mentioned in Table 21. In this report only the directional information for the points P1, P2 and P3 is given. The directional wave information is been presented with the aid of annual and monthly wave roses diagrams. In this report only the annual wave roses are presented.

P1 near Rødbyhavn

The main wave directions at point P1 near the Danish coast are:

- W-WSW (approximately 40% of the time)
- S-SES (approximately 35% of the time)

The highest waves mostly occurs from WSW. This direction corresponds with the longest fetch direction at that location. Winds from these directions are frequent.

P2 in the middle of the corridor

The main wave directions in the middle of the corridor are:

- W-WNW (approximately 35% of the time)
- E-SE (approximately 25% of the time)

Winds from these directions are frequent and the distance from Fehmarnbelt to upwind land areas is also relatively large. Waves from these directions are therefore the largest and most frequent.

P3 near Puttgarden

In the southern part of the corridor near the German coast the predominant wave directions are:

- W-NW (approximately 30-40% of the time)
- E-SE (approximately 30-40% of the time)

The wave direction at this location is affected by the sheltering effect from the Western part of the Fehmarnbelt. Further it's been analyzed that the highest waves at point 3 occurs from the east. This is also the direction with the longest fetch. The annual wave roses for the three locations (P1,P2 and P3) along the corridor are presented in Figure 20.

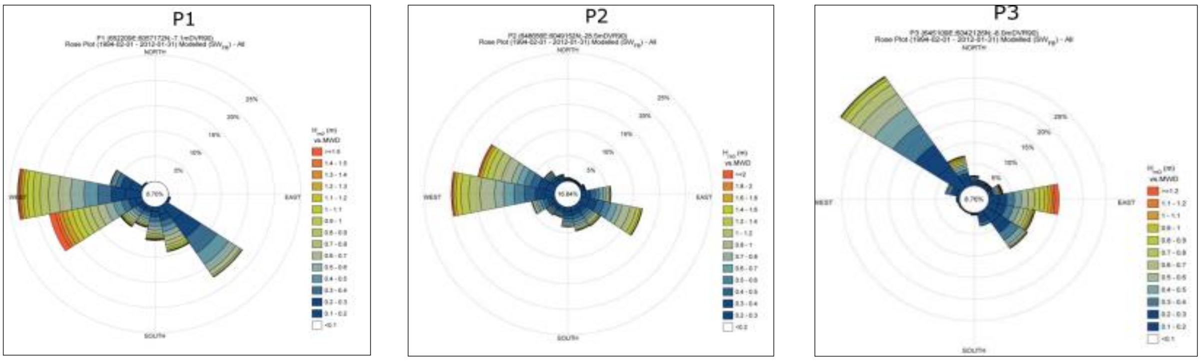


Figure 20 Annual Wave Roses at locations P1,P2 and P3

Also the monthly wave roses for each location are provided by the owner of the project. The monthly wave roses resembles the annual wave roses. And that´s why only the annual wave roses are presented here. In the annual frequency distribution of the omni-directional significant wave height for the three location are given. From it can be concluded that smaller waves are more frequent.

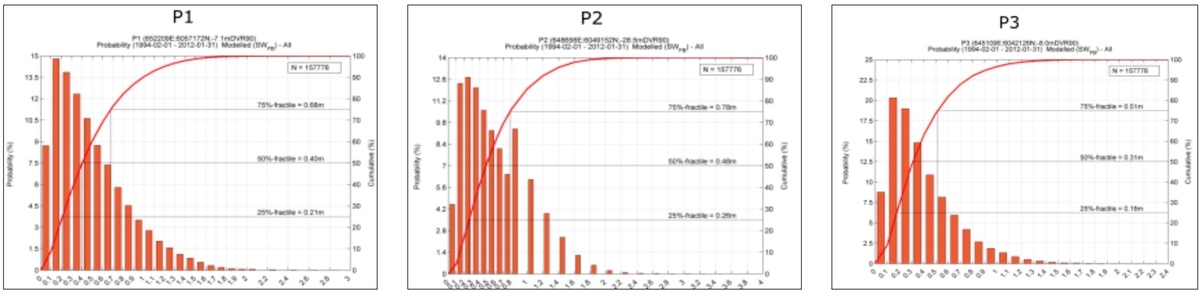
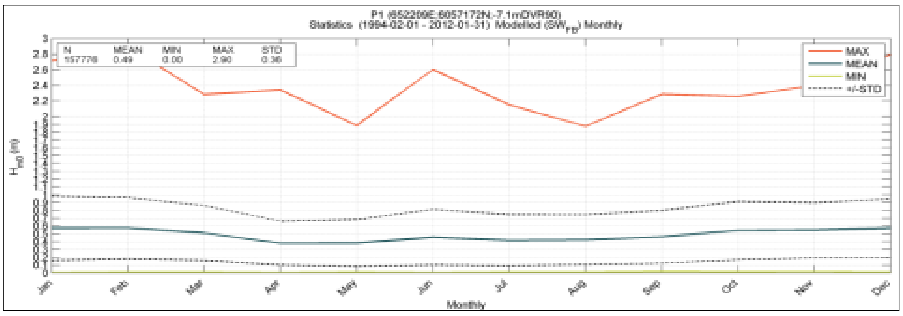
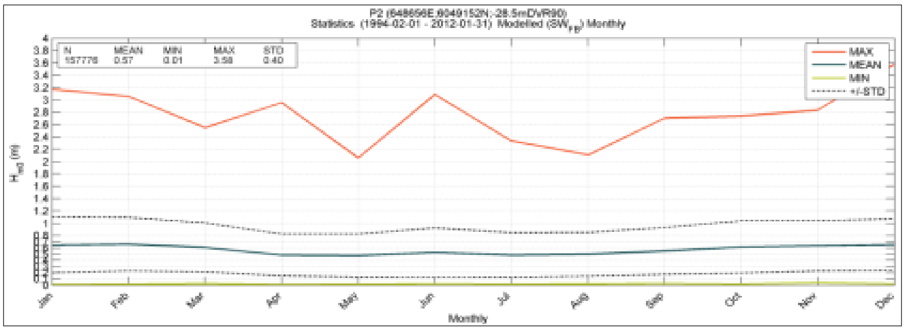


Figure 21 The annual frequency distribution of the omni-directional significant wave height

The monthly omni-direction variation of the wave height in three points P1, P2 and P3 is presented in Figure 21. From the diagrams it can be concluded that the highest waves occurs in winter months. But also during the summer period the wave height can exceed a wave height of 2 m in the northern part of the link. Particularly in June, the mean and maximum wave height is around 90% of the annual mean and maximum significant wave height caused by fairly strong westerly winds.





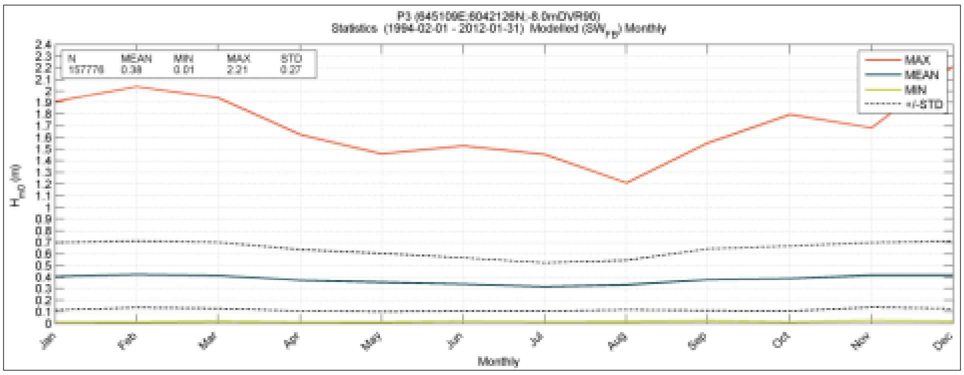


Figure 22 the monthly omni-direction variation of the wave height

Correlation between significant wave height and wave periods.

From the data plots of the wave heights and wave periods it can be concluded that there is a good correlation between wave heights and wave periods. The correlation can be described as:

$$T_p = \alpha_1 H_{m_o}^{\beta_1} \qquad \alpha_1 = 4.51 - 4.80 \quad \beta_1 = 0.35\text{-}0.37 \tag{47}$$

$$T_{02} = \alpha_2 H_{m_o}^{\beta_2} \qquad \alpha_1 = 3.23 - 3.32 \quad \beta_2 = 0.38\text{-}0.43 \tag{48}$$

Correlation between wave height and wind speed

There is a clear correlation between the significant wave height and U_{10}. The highest waves occur during the highest wind speeds. The westerly wind cause highest waves in the northern part of the link, while the easterly wind results highest waves in the southern part of the link.

The highest waves in the middle of the corridor are caused by the predominant wave field W-WNW and E-SE. The highest wave at P2 with a return period of one year is 2.5 m.

Correlation between wave height and water level

From the scatter diagrams it can be concluded that there is a trend between the wave height and water level. In the middle and the northern part of the corridor the highest waves occur during the low water level. Lower water levels are associated with westerly winds. At the southern part of the link the highest waves occur in combination with height water level. Higher water levels are associated with the easterly winds, which pushes water from Baltic sea in Fehmarnbelt.

Correlation between wave height and current velocity

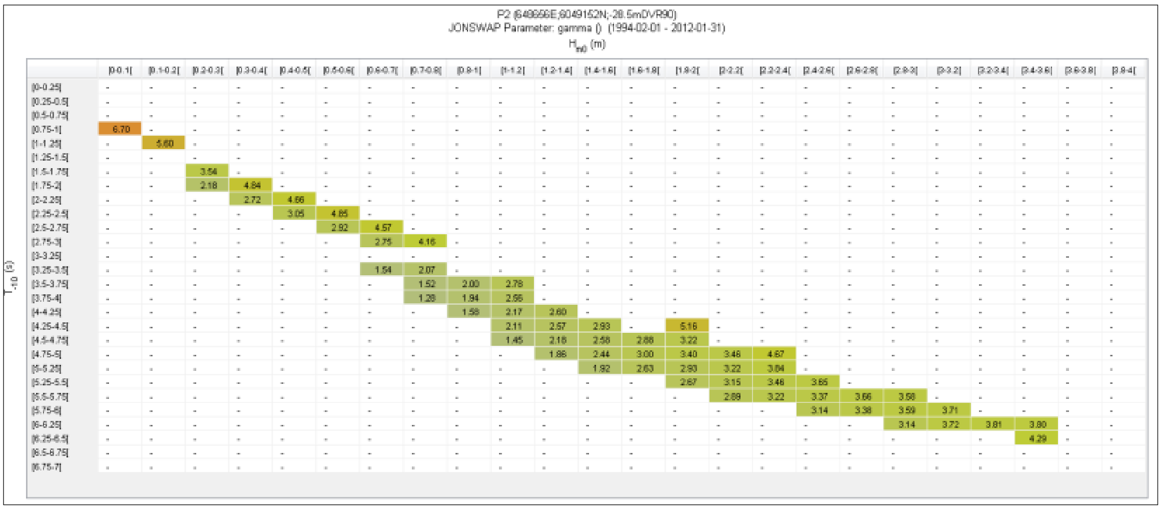
There is no clear correlation between the current speed and wave height in Fehmarnbelt. High waves were measured in conjunction with the high current speed. Both the high waves and high current speeds are related to the wind field.

Wave Spectrum

In general it can be said that sea states in confined areas are dominated by locally wind-generated waves (like the Fehmarnbelt). This areas are traditionally parameterised by the JONSWAP spectrum. It is originally a single-peaked spectrum and hence rarely representative of mixed sea (bi-modal) conditions.

For the obtained data the JONSWAP and Occhi-Hubble spectra are fitted to the average of all spectra within each interval of significant wave height (Hm0) and mean energy wave period T_{10}. The mean energy wave period was found to be more robust for characterisation of the modelled spectra (sea states) compared to other characteristic wave periods such as $T_{01}, T_{02}\ or\ T_p$. Plots of fitted spectra for all represented bins of H$_{m0}$/T$_{-10}$, and accompanying tables of the JONSWAP and Ochi-Hubble parameters, for which reasonable fits were established. Recommended parameters of the empirical wave spectra are derived based on the 18-year model data of 1 hourly values.

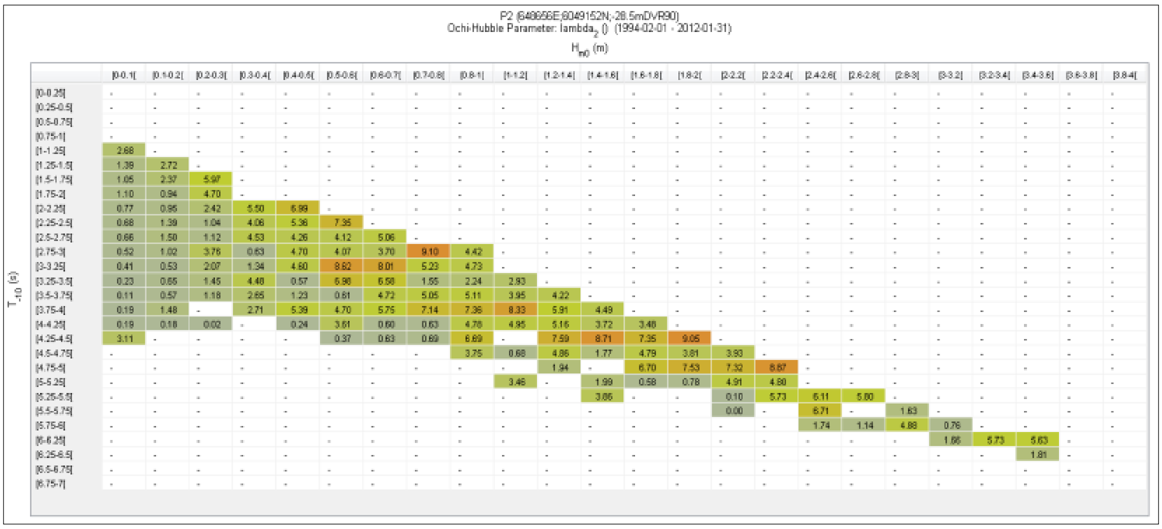
The number of occurrence of sea states in each bin of $\mathrm{H_{m0}/T_{10}}$ at the middle of the corridor are given in Table 24 and Table 25.



P2 (648656E;6049152N;-28.5mDVR90)
JONSWAP Parameter: gamma () (1994-02-01 - 2012-01-31)
H_{m0} (m)

T_{-10} (s)	[0-0.1[	[0.1-0.2[	[0.2-0.3[	[0.3-0.4[	[0.4-0.5[	[0.5-0.6[	[0.6-0.7[	[0.7-0.8[	[0.8-1[	[1-1.2[	[1.2-1.4[	[1.4-1.6[	[1.6-1.8[	[1.8-2[	[2-2.2[	[2.2-2.4[	[2.4-2.6[	[2.6-2.8[	[2.8-3[	[3-3.2[	[3.2-3.4[	[3.4-3.6[	[3.6-3.8[	[3.8-4[
[0-0.25[	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[0.25-0.5[	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[0.5-0.75[	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[0.75-1[	6.70	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[1-1.25[	-	5.60	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[1.25-1.5[	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[1.5-1.75[	-	-	3.54	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[1.75-2[	-	-	2.18	4.84	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[2-2.25[	-	-	-	2.72	4.66	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[2.25-2.5[	-	-	-	-	3.05	4.85	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[2.5-2.75[	-	-	-	-	-	2.92	4.57	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[2.75-3[	-	-	-	-	-	-	2.75	4.16	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[3-3.25[	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[3.25-3.5[	-	-	-	-	-	-	1.54	2.07	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[3.5-3.75[	-	-	-	-	-	-	-	1.52	2.00	2.78	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[3.75-4[	-	-	-	-	-	-	-	1.28	1.94	2.56	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[4-4.25[	-	-	-	-	-	-	-	-	1.58	2.17	2.60	-	-	-	-	-	-	-	-	-	-	-	-	-
[4.25-4.5[	-	-	-	-	-	-	-	-	-	2.11	2.57	2.93	-	5.16	-	-	-	-	-	-	-	-	-	-
[4.5-4.75[	-	-	-	-	-	-	-	-	-	1.45	2.18	2.58	2.88	3.22	-	-	-	-	-	-	-	-	-	-
[4.75-5[	-	-	-	-	-	-	-	-	-	-	1.86	2.44	3.00	3.40	3.48	4.67	-	-	-	-	-	-	-	-
[5-5.25[	-	-	-	-	-	-	-	-	-	-	-	1.92	2.63	2.93	3.22	3.04	-	-	-	-	-	-	-	-
[5.25-5.5[	-	-	-	-	-	-	-	-	-	-	-	-	-	2.67	3.15	3.46	3.65	-	-	-	-	-	-	-
[5.5-5.75[	-	-	-	-	-	-	-	-	-	-	-	-	-	-	2.89	3.22	3.37	3.66	3.58	-	-	-	-	-
[5.75-6[	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	3.14	3.38	3.59	3.71	-	-	-	-
[6-6.25[	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	3.14	3.72	3.81	3.80	-	-
[6.25-6.5[	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	4.29	-	-
[6.5-6.75[	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[6.75-7[	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

Table 24 Fitted JONSWAP peak enhancement factors γ at P2



P2 (648656E;6049152N;-28.5mDVR90)
Ochi-Hubble Parameter: $\mathrm{lambda_2}$ () (1994-02-01 - 2012-01-31)
H_{m0} (m)

T_{-10} (s)	[0-0.1[	[0.1-0.2[	[0.2-0.3[	[0.3-0.4[	[0.4-0.5[	[0.5-0.6[	[0.6-0.7[	[0.7-0.8[	[0.8-1[	[1-1.2[	[1.2-1.4[	[1.4-1.6[	[1.6-1.8[	[1.8-2[	[2-2.2[	[2.2-2.4[	[2.4-2.6[	[2.6-2.8[	[2.8-3[	[3-3.2[	[3.2-3.4[	[3.4-3.6[	[3.6-3.8[	[3.8-4[
[0-0.25[	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[0.25-0.5[	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[0.5-0.75[	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[0.75-1[	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[1-1.25[	2.68	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[1.25-1.5[	1.39	2.72	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[1.5-1.75[	1.05	2.37	5.97	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[1.75-2[	1.10	0.94	4.70	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[2-2.25[	0.77	0.95	2.42	5.50	6.99	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[2.25-2.5[	0.68	1.39	1.04	4.06	5.36	7.35	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[2.5-2.75[	0.66	1.50	1.12	4.53	4.26	4.12	5.06	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[2.75-3[	0.52	1.02	3.76	0.63	4.70	4.07	3.70	9.10	4.42	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[3-3.25[	0.41	0.53	2.07	1.34	4.60	8.62	8.01	5.23	4.73	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[3.25-3.5[	0.23	0.65	1.45	4.48	0.57	6.98	6.58	1.55	2.24	2.83	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[3.5-3.75[	0.11	0.57	1.18	2.65	1.23	0.61	4.72	5.05	5.11	3.95	4.22	-	-	-	-	-	-	-	-	-	-	-	-	-
[3.75-4[	0.19	1.48	-	2.71	5.39	4.70	5.75	7.14	7.36	8.33	5.91	4.49	-	-	-	-	-	-	-	-	-	-	-	-
[4-4.25[	0.19	0.18	0.02	-	0.24	3.61	0.60	0.63	4.78	4.95	5.16	3.72	3.48	-	-	-	-	-	-	-	-	-	-	-
[4.25-4.5[	3.11	-	-	-	-	0.37	0.63	0.69	6.69	-	7.59	8.71	7.35	9.05	-	-	-	-	-	-	-	-	-	-
[4.5-4.75[	-	-	-	-	-	-	-	-	3.75	0.68	4.86	1.77	4.79	3.81	3.93	-	-	-	-	-	-	-	-	-
[4.75-5[	-	-	-	-	-	-	-	-	-	-	1.94	-	6.70	7.53	7.32	6.87	-	-	-	-	-	-	-	-
[5-5.25[	-	-	-	-	-	-	-	-	-	3.46	-	1.99	0.58	0.78	4.91	4.80	-	-	-	-	-	-	-	-
[5.25-5.5[	-	-	-	-	-	-	-	-	-	-	-	3.86	-	-	0.10	5.73	6.11	5.80	-	-	-	-	-	-
[5.5-5.75[	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0.00	-	6.71	-	1.63	-	-	-	-	-
[5.75-6[	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1.74	1.14	4.88	0.76	-	-	-	-
[6-6.25[	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1.66	5.73	5.63	-	-
[6.25-6.5[	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1.81	-	-
[6.5-6.75[	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
[6.75-7[	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

Table 25 Fitted Ochi-Hubble parameters λ2 at P2

Reasonably fits for the JONSWAP spectrum were obtained approximately for half of the represented sea states. Many of the sea states without the reasonable have a limited occurrence chance (P(f)< 1 ‰). For the practical purposes this sea states can be also omitted. For other sea states with low wave height and relatively height wave periods couldn’t be described by the JONSWAP spectrum. In those cases the Ochi-Hubble wave spectrum could be used. The peak enhancement factors for both type of spectra can be found in Table 24and Table 25

Long period waves

The occurrence of long period waves(8-16 s) under normal wave conditions is very limited. The total energy of the long period waves is about 0.4 % of the total wave energy. The long period waves occur normally from easterly winds, where the fetches are longest. During the extreme conditions, say with a return period of 10 years and Hm0 > 3m the amount of long period wave energy is larger. Even longer period waves (infragravity waves, periods > 30s) may occur near the shoreline and in shallow water areas under extreme weather conditions.

	0-0.1	0.1-0.2	0.2-0.3	0.3-0.4	0.4-0.5	0.5-0.6	0.6-0.7	0.7-0.8	0.8-1.0	1.0-1.2	1.2-1.4	1.4-1.6	1.6-1.8	1.8-2.0	2.0-2.2	2.2-2.4	2.4-2.6	2.6-2.8	2.8-3.0	3.0-3.2	3.2-3.4	3.4-3.6	3.6-3.8	3.8-4.0
0-0.25																								
0.25-0.5																								
0.5-0.75																								
0.75-1	9																							
1-1.25	247	60																						
1.25-1.5	1365	675																						
1.5-1.75	2035	4015	136																					
1.75-2.0	1581	5990	2930	25																				
2.0-2.25	740	4772	7078	1473	17																			
2.25-2.50	392	2570	5784	6464	702	17																		
2.50-2.75	247	1019	2856	6895	5059	540	23																	
2.75-3.0	201	224	1071	3068	6997	4192	510	35	4															
3.0-3.25	118	86	194	899	3137	6676	3991	543	33															
3.25-3.50	52	30	27	169	666	2765	6185	3959	903	4														
3.50-3.75	61	8	2	20	117	403	1857	4663	5006	148	4													
3.75-4.00	34	7		1	16	50	205	902	6924	1832	42	1												
4.00-4.25	18	2	1		1	4	23	70	1835	5395	732	33	1											
4.25-4.5	7					1	2	4	107	2190	3713	368	15	1										
4.5-4.75									4	70	1643	2412	241	4	1									
4.75-5.00											46	911	1340	216	7	1								
5.00-5.25										1		12	309	615	135	7								
5.25-5.50												1		51	205	102	11	1						
5.50-5.75															2	32	37	18	3					
5.75-6.00																	3	17	11	6				
6.00-6.25																			1	3	1	2		
6.25-6.50																						1		
6.50-6.75																								

Table 26 Wave Scatter diagram

3 DETERMINISTIC DESIGN AND PARTIAL FACTORS

For determining the dimensions of the pontoons the required strength (floating capacity) has to have a design value larger than the design value of the load (different weights working on the pontoons during the immersion). In formula form this can be given as:

$$R_d > S_d \tag{49}$$

This leads to the following definition:

$$\frac{R_{rep}}{\gamma_R} > \gamma_s * S_{rep} \tag{50}$$

In which:

R_{rep}	Representative value for the strength
S_{rep}	Representative value for the loads
γ_R	Partial safety factor for the strength
γ_S	Partial safety factor for the load
R_d	Design value of the strength
S_S	Design value for the load

The representative values are related to the characteristic values of the loads and strength. The characteristics values are exceeded by the 95 % of the strength and 95% not exceeded of the time by the load. The idea behind of this approach is that when multiplying the upper limits of the load and the under limits of the strength by the load and material factors a very small failure probability will be acquired. The principle of this approach is shown in Figure 23.

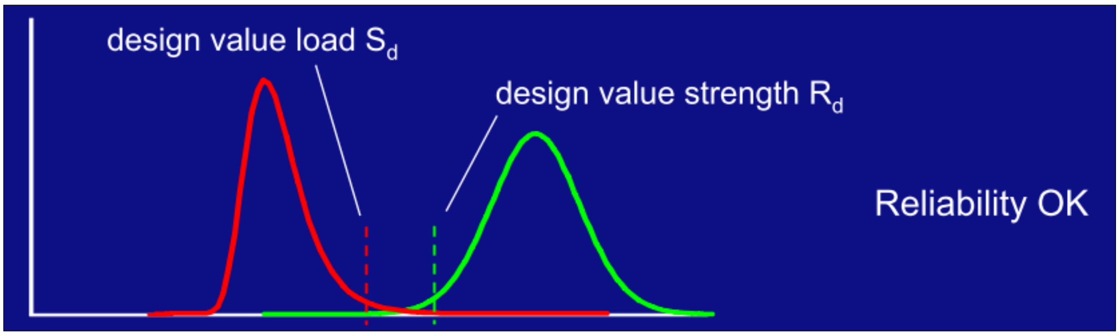


Figure 23 Principle of Reliability of the system

3.1 COMBINATIONS OF THE LOAD:

During the immersion different weight combinations will be working on the pontoons as a load at the same time. The floating capacity of the pontoons should be sufficient to resist/carry the different combinations of the loads (weights):

$$\frac{R_{rep}}{\gamma_{rep}} > \sum \gamma_{si} * S_{rep;i} \tag{51}$$

The pontoons should have enough floating capacity to be able to carry the weight of:

- The self-weight of the pontoons and equipment on the pontoons
- Weight of the tunnel element
- Weight of Ballast to overcome the salinity difference of the water
- Weight of Ballast to overcome the upward dynamic wave load. Extra TE weight duo to construction inaccuracies

In general during the immersion process not all mentioned loads will work on the pontoons at the same time. It will be too conservative to add up the representative values of all the above mentioned loads and multiply by the same safety factor. Because the maximum values of the loads do not act at the same time on the system. That's why the following design conditions are considered in determining the dimensions of the pontoons. Different load combination are analyzed in different design combinations according the Eurocode.

3.2 LOADS

There are several loads which are working on the system. Different types of loads are important during different stages. The loads considered in this study are the loads which are important in the transportation but mainly in the immersion phase. The loads which are determinative for designing the system are:

- Self-weight tunnel element and the equipment
  (including the weight of the immersion rigs)
- Buoyancy force
- Dynamic wave loads
- Current forces

The loads are classified dependent on their variation in the time as follows:

- Permanent load (the weight of the structure and the equipment)
- Variable loads (buoyancy force, wave forces, current forces)
- Exceptional forces (explosion or collision forces 'not considered here')

The loads will be represented by their characteristic values. The variation in the weight of the tunnel element can't be disregarded. That's why an upper limit value and lower limit value will be calculated. The upper limit is then the 5% fractal and the under limit is the 95 % fractal of the statistical distribution. The Gaussian distribution has been assumed as a statistical distribution.

Variable Loads

For the variable loads the characteristic value must match the upper limit value with a purposed chance of not been exceeded during a certain reference period of 50 years. In cases where the statistical distribution is unknown a nominal value will be used as the characteristic value.

The characteristic value of loads due to weather conditions is based on the probability 0.02 that its time-varying portion is exceeded in a reference period of one year. This means that an average return period of 50 years is been assumed for the variable loads due to weather conditions.

Dynamic Loads

For the dynamic loads such as wave load and current loads which cause accelerations of the structures (tunnel element and the immersion rigs) the structural system will be dynamically analyzed. The system will be modelled as mass spring system. The syetem will be in a 3D model analyzed. The dynamic wave loads will be presented in the frequency and time domain. Also the response of the structure will be presented in the time and frequency domain.

Environmental influences

The environmental influences which affect the durability of the structure are considered only for the selection of the construction concrete. In the selection of the structural material for the immersion rigs and different kind of cables they are disregarded.

Material properties

The properties of the materials or products are represented by the characteristic values. There where a low value of the material property has a negative influence on the system the characteristic value has been determined as the 5% fractile value. This is done to determine the characteristic value of the cable strength.

The reduction of the material strength in time due to the reparative load effects, such as fatigue is disregarded in the calculations. The structural stiffness parameters such as elastic moduli of steel is represented by its average value. Different values are used in order to bring the time effect of the load. The values presented in the Eurocode will be used for the different parameters. In order to get the design values a partial material factor will be used.

Geometric data

In order to count the effect of the geometric uncertainties for the weight calculations it's been assumed that due to the imperfections the weight of the tunnel element could be 0.30% more or less than calculated nominal value. The geometric values of the immersion rigs are presented by the nominal values.

Load factors

Generally for Hydraulic structures safety class 3 according to Eurocode is applicable. However for the temporary structures as the immersion rigs the safety class 1 can be applied. The method of partial factors will be used for all design situations to determine the design values for strength and loads.

Design values of Loads

The design values of the different are loads formulated as:

$$S_d = \gamma_s * S_{rep} \tag{52}$$

$$S_{rep} = \psi * S_k \tag{53}$$

- S_k Characteristic value of the load
- S_{rep} Representative value of the load
- γ_s Is a partial load factor which takes into account the possibility of unfavourable deviations of the load values relative to the representative values.
- ψ Combination factor (1.00 or ψ_0 in this model)

3.3 DESIGN VALUES OF THE STRENGTH

The design values of the strength can be represented as:

$$R_d = \frac{R_k}{\gamma_m} \tag{54}$$

The design value of the strength is calculated directly from the characteristic values of the material.

Partial safety factors:

Different partial safety factors are specified for the different materials and loads. The partial safety factors are used to be sure that the capacity of the different types of structural components is sufficient in regard to different modes of failure.

Furthermore for the cases when more than one variable load component is acting, load combination factors are used. This is done to take into account that it's unlikely that all the variable loads are acting with the extreme values at the same time.

The partial safety factors for the strength and the loads related to the ULS should be chosen such that the reliability levels for the components in the system should be as close as possible to the target reliability index according to the Eurocode. The partial safety factors in combination with the characteristic values are used to ensure that the target reliability value is achieved according the Eurocode.

$$\gamma_m = \frac{R_k}{R^*} = \frac{1 + k * V_R}{1 + \alpha_R * \beta_{target} * V_R} \qquad (55)$$

$$\gamma_s = \frac{S^*}{S_k} = \frac{1 + \alpha_s * \beta_{target} * V_s}{1 + kV_s} \qquad (56)$$

With:

R_k Characteristic strength value

R^* Design point value of the strength (level 2 method)

(This point can be defined as a point on the failure boundary closest located to the average or expected value of the strength in the space of the normalized strength and load)

γ_m Partial safety factor strength parameter

k $k = 1.64$ for the 5% fractile of the standardised normal distribution

V_X Coefficient of variation strength parameter ($V_X = \sigma_X/\mu_X$)

α_R FORM factor strength (level 2 method)

γ_s Partial safety factor load

S^* Design point value of the load (level 2 method)

S_k Characteristic load value

α_s FORM factor load (sensibility factor →level 2 method)

β_{target} Target reliability index value according to the Eurocode (RC1→$\beta = 3.3$) (The chosen value of the target reliability index consist with the reliability class RC1 in Eurocde)

V_s Coefficient of variation load

Remark: the values of k and α_R are negative in the above represented equations. For the calculations of the partial safety factors standardized α values are used:

α	Significance of the load
0.70	Dominant load parameter
0.28	Other load parameter
-0.80	Dominant strength parameter
-0.32	Other strength parameter

Table 27 alpha factors for different loads

Permanent load parameters

The following values are been calculated for the partial factors for the permanent loads. In this model the permanent loads are the loads related to the different kind of self-weights. It's been assumed that the weight is normal distributed. Furthermore different values for the variation coefficient have been used.

In this model it's been assumed that the tunnel elements which has to be immersed are the standard types. The dimensions and the configuration of the elements have been given. The uncertainties in the weight of the tunnel element will be for the important part due to in the uncertainties in the density of the reinforced concrete. The uncertainties in the in the geometric properties are separately has been taken into account. That's why a lower variation coefficient has been used for the weight of the tunnel element V_G = 0.05.

The weight of the immersion rigs and the equipment is more uncertain. That's why a greater variation coefficient V_G = 0.20 has been uses for the other weights. The partial factor is given in the formula form in the equation (57)

$$\gamma_G = \frac{\theta_d}{\theta_{nom}} * (1 + \beta_{target} * \alpha_S * V_G) \tag{57}$$

The partial factor for the concrete:

The partial factor when the load is considered to be unfavourable and it's a dominant load:

$$\gamma_G = 1.05 * (1 + 3.3 * 0.7 * 0.05) = 1.20 \tag{58}$$

The partial factor when the load is considered to be unfavourable and it's not a dominant load:

$$\gamma_G = 1.05 * (1 + 3.3 * 0.28 * 0.05) = 1.10 \tag{59}$$

The partial factor when the load is considered to be favourable and it's a dominant load:

$$\gamma_G = 1.00 * (1 - 3.3 * 0.8 * 0.10) = 0.86 \tag{60}$$

The partial factor when the load is considered to be favourable and it's not a dominant load:

$$\gamma_G = 1.00 * (1 - 3.3 * 0.32 * 0.10) = 0.95 \tag{61}$$

The partial factor for the other weights:

The partial factor when the load is considered to be unfavourable and it's a dominant load:

$$\gamma_G = 1.05 * (1 + 3.3 * 0.7 * 0.20) = 1.55 \tag{62}$$

The partial factor when the load is considered to be unfavourable and it's not a dominant load:

$$\gamma_G = 1.05 * (1 + 3.3 * 0.28 * 0.05) = 1.25 \tag{63}$$

The partial factor when the load is considered to be favourable and it's a dominant load:

$$\gamma_G = 1.00 * (1 - 3.3 * 0.8 * 0.10) = 0.5 \tag{64}$$

The partial factor when the load is considered to be favourable and it's not a dominant load:

$$\gamma_G = 1.00 * (1 - 3.3 * 0.32 * 0.10) = 0.8 \tag{65}$$

Variable load parameters

For determining the partial factors for the variable loads the same approach has been used as presented above. But because the Eurocode standards are not specifically aimed for the hydraulic structures as modelled here not for all variable loads is the same approach has been followed. For the determining the wave load a different approach has been followed.For the most variable loads a normal distribution has been assumed and for the wave load a Poisson distribution has been assumed. For the variable loads a coefficient of variation V_Q =0.20 has been assumed.

$$\gamma_Q = \frac{\theta_d}{\theta_{nom}} * (1 + \beta_{target} * \alpha_S * V_Q) \tag{66}$$

The partial factor when the load is considered to be unfavourable and it's a dominant load:

$$\gamma_Q = 1.05 * (1 + 3.3 * 0.7 * 0.20) = 1.55 \tag{67}$$

The partial factor when the load is considered to be unfavourable and it's not a dominant load:

$$\gamma_Q = 1.05 * (1 + 3.3 * 0.28 * 0.05) = 1.25 \tag{68}$$

The factor ψ that considers the reduction in the design values of the variable loads which accrue together with the dominant variable load is:

$$\psi_0 = \frac{1.25}{1.55} = 0.80 \tag{69}$$

Remark:

In this model used partial factors γ_G and γ_Q differs from the values presented in the Eurocode:

$$\gamma_G = 1.35, 1.15, 1.00 \text{ and } 1.00 \tag{70}$$

$$\gamma_Q = 1.50 \tag{71}$$

$$\psi_0 = 0.70 \tag{72}$$

This is because of the different value used for the target reliability index and different values used for the variation coefficients.

Partial value for the wave loads

The most important waves in the project area are the wind generated waves. The immersion system (the tunnel element and the immersion rigs) have to be designed such that they must be able to withstand a certain wave height. This the so called wave height has to be known. From measurements in the project area the omni-directional significant wave height and the wave peak periods are known.

The significant wave height occurs regularly and it's much lower that the design wave height which has to be considered. For the calculations only the significant wave height during the normal condition is considered. Because it's assumed that the tunnel element will be not immersed during the extreme weather conditions. Also the effect of the shallow water is been disregarded because the significant wave height during the normal condition is relatively small compared to the water depth. Given the earlier mentioned assumption the statistical distribution of the wave height has been assumed to be Rayleigh distribution. The probability of exceedance of a certain wave can be presented as:

$$\Pr(H > X) = exp\left\{-2 * \left(\frac{X}{H_s}\right)^2\right\} \quad (73)$$

The exceedandce probability of a wave height greater than the design wave height during the immersion operation with N waves can be given as:

$$\Pr(H > H_d) = 1 - exp\left\{-N * e^{-2\left(\frac{H_d}{H_s}\right)^2}\right\} \quad (74)$$

The duration of the immersion operation has been assumed to be 36 hours:

$$T_{immersion} = 3\ hours \quad (75)$$

The wave conditions in Fehmarnbelt are generally mild. The waves are short with over 90% of the mean wave periods below 4.0 s and peak wave periods generally less than 6.5 s. The presuming wave period during the immersion operation is assumed to be 5 s. The expected number of waves is then:

$$N = \frac{T_{immersion}}{T_{wave}} \approx 2700\ waves \quad (76)$$

According the earlier mentioned approach an acceptable probability for the exceedance is chosen $\phi(-\beta_{\text{target}})$ In formula form this can be given as:

$$\Pr(H > H_d) = \phi(-\beta_{\text{target}}) = 0.0005 \quad (77)$$

Then the design wave height is:

$$H_d = 2.1 * H_s \quad (78)$$

In all calculations a wave height H_d will be used instead of the significant wave height.

4 ANNEX (CONCRETE DENSITY)

4.1 IMPORTANT DESIGN MIX ASPECTS

The immersed tunnel Fehmarnbelt is build and buried in a marine environment, gaining access to the outside surface for inspection or repair is impractical and cost ineffective. It's necessary to ensure that the tunnel elements will be designed such that

structure is inherently durable and will require little or no maintenance during the life time. Any maintenance work that is required should be carried out from within the tunnel without disruption to its operation. The desirable durability of the tunnel elements should be achieved through the measures in the design and construction of the tunnel element.

The durability of the tunnel elements can be divided into two components:

- The durability of the materials
- Water tightness of the tunnel structure

For the concrete tunnels the durability is related to the quality of the used concrete mix. Watertight concrete is achieved by eliminating the cracks in the structure and good quality impermeable concrete.

Water tightness
Water tightness of the immersed tunnel as a whole structure is achieved by proper design of the joints (not considered here) and throughout the structure itself. By water tightness of the tunnel elements is mean:

- Free of all visible leakage, seepage, and damp patches

Also the concrete mix should be composed such that the saline sea water could not reach the reinforcement bars and cause erosion of the reinforcement steel. The tunnel will be built in aggressive marine environment with height salinity rates. The maximum salinity rates measured in the 60 years measurements is 27 PSU (1031 kg/m^3) on the surface water. At the bottom of the of the water column is the water salinity larger, approximately is the water salinity at the bottom 5 PSU more than at the surface. It's important that the chloride intrusion in the concrete matrix must be limited. A target chloride diffusivity coefficient will need to be achieved.

Concrete Strength
In general the concrete strength in an immersed tunnel element does not need to be exceptional. In fact quite low concrete strength can be used for the elements. This is because the wall and slab thicknesses are often slightly oversized to ensure there is sufficient weight in the structure to resist the uplift. But the elements of the Fuhrmanbelt link are placed in deep water. In the deepest point of the link the elements will be ballasted by more than 30 m water column. Also the tunnel elements has to be pre-stressed in the transportation phase. In order to ensure that the elements can carry the loads in the final and construction phase a minimum concrete strength of 45 C is chosen in for this project. Also the rate of strength gained in early days and weeks of curing needed to be taken into account in order to be able to be able to couple the production process of the tunnel elements.

The strength requirements are also influenced by the construction process of the tunnel elements. The young strength requirements are influenced by the fact the formwork shutters and the falsework supports are released. The decision on this will depend on the method of procurement.

Traditionally the immersed elements are built in a construction dock. There for strength requirement of 28-day strength to ensure the concrete matures will be sufficient. However for the tunnel elements of the Fehmarnbelt project additional requirements for strength gain are needed to facilitate the construction process. As mentioned before, the tunnel elements will be built in a fabric with a complex set of construction operations. After pouring a segment the internal formwork supporting the roof slab will be released and the segment will be pushed along the skidding beam.

For releasing the formwork supports and pushing the segments along the skidding beam the reference values of the Oresund tunnel project are used (reference project). It's assumed that the internal roof slab will be released after 3 days and that the segment will be pushed after 5 days. This operation will influence the required earlier age strength of the poured concrete segments. As a result of the earlier mentioned operation some rapid strength gain is needed.

Hydration of cement and crack control

The generated heat during the construction of the elements is of the major importance. By choosing the concrete mix measures has to be considered to limit the heat development. In addition the shrinkage properties of the concrete mix need to be well understood and control over the degree of shrinkage is needed.

To eliminate the through-section cracking in a tunnel element during the construction can be influenced by the proper choice of the concrete mix. Instead of cement, replacement materials such as GGBFS (ground granulated blast furnace slag) and PFA (Pulverized fly ash) can be used to reduce primarily the hydration heat and subsequently the contraction. The impact of using such replacement materials will be that the concrete has a slower rate of gain in strength. For a proper proportion of the replacement materials the rules given in the Eurocode could be used. On the other hand as explained in the subparagraph strength, after pouring the concrete some initial strength gain is needed for the construction operations. These two demands are contradictory. Some balance is needed between these two contradictory demands.

The watertightness of the tunnel elements will be seriously comprised if the cracking penetrates the full thickness of the concrete in the external perimeter of the tunnel elements. The risk of this kind of cracking should be avoided and eliminated. Cracking in concrete of the tunnel elements occur due to tensional forces in the structure. The following cracking types should be considered for the tunnel elements of the Fehmarnbelt tunnel:

- Restraint to movement of the completed tunnel
- Loading causing longitudinal bending moments in the tunnel
- Restraint to movement between concrete pours during the construction stage

The first two types of cracking are part of the structural analysis and design of the tunnel elements, they are not considered here. This section of the report deals with the cracking due to restrains to movement between concrete pours. This kind of cracking occurs during the earlier age while the poured concrete is curing.

Traditionally immersed tunnel elements are constructed as series of concrete pours. They are typically constructed in the following sequence:

1. Base slab (fist stage pour)
2. Walls and the roof (second stage pour)

This method of construction requires construction joints between the stages of concreting. This kind of joints gives rise to the most common threat of watertightness of the tunnel elements.

The reinforcement and the 'old' poured concrete prevent the 'new' poured concrete to contract. Due to difference in the hydration process of the 'old' and the 'new' poured concrete there occur tensional forces in the concrete and exceeds the available tensional strength of the young concrete which leads to cracking (see Figure 24)

The tunnel elements are placed in relatively deep water, in deepest point more than 30 m above the roof of the tunnel element. The elements in the will be exposed to high external water pressure (more than 40 bars in the deepest point of the bay). The crack control of the tunnel elements is of imminent importance. Cracking which will cause water leakage should be eliminated in the construction process. The tunnel elements of the Oresund tunnel project were produces in a different way than the traditional immersed tunnels. The concrete used in one segment was poured in one operation in the following sequence:

1. Base slab
2. Walls
3. Roof slab

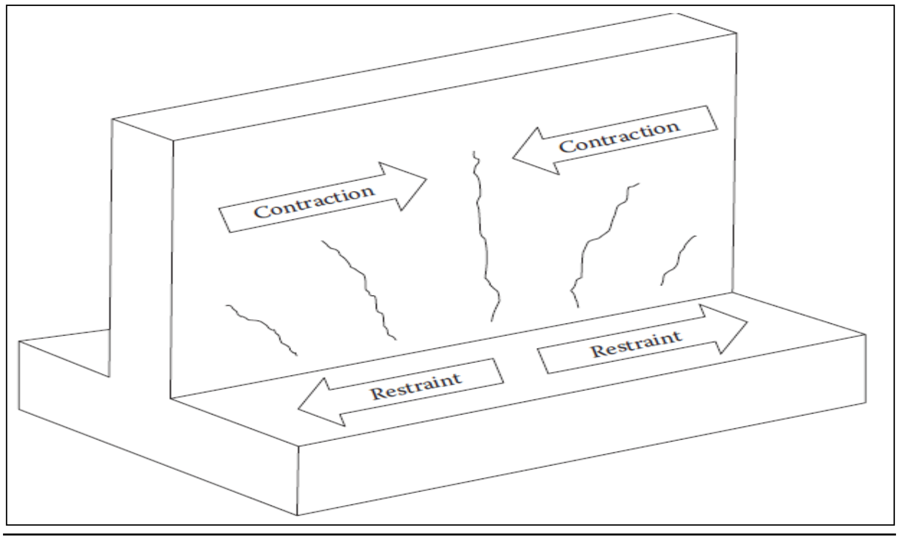


Figure 24 cracking caused by restraint between slab and wall

This technique has a great advantage above the traditionally used construction method that the construction joints in the elements were eliminated. Other advantage of this method is that the cast-in cooling pipes are not necessary to be used.

The Oresund tunnel was the first immersed tunnel project on great scale in which this technique has been used. The technique appears to be 100% success full in avoiding the early age cracking due to thermal effects. This technique is also been used for the Busan tunnel project (other reference project).

Because of the similarities of the Fehmarnbelt Fixed link project with the mentioned reference project it is plausible to assume that the tunnel elements of the Fehmarnbelt tunnel will be produced in the same manner. For long transportation tunnels such as the Fehmarnbelt tunnel this method of construction is to be preferred approach because of the many advantages.

Flow ability of the concrete

The tunnel elements are large concrete structures with dimensions (L x B x H → 217.9 x 41.2 x 8.9). Each element is built from 9 segments with a length of 24.21 m. The elements are produced in a fabric environment using large volume concrete pours. High-flow mixes are needed with associated admixtures to ensure that segregation does not occur. In order to be able to handle the large volumes of concrete addition of the retarders is needed. The risk of the poor compaction needed to be avoided. The composed mix must be able to flow into the geometric features like the immersion joints, around waterbars and around tie bars. In order to increase the flow ability the use of superplasticizer is needed.

Water - cement ratio

Cement and water form adhesive that bound the aggregate material (sand and gravel) together. The ratio mass of the added mass and cement is indicated as water cement factor (wcf). The wcf determines the strength and durability properties of the produced concrete. The increasing water cement factor has a negative influence on the quality of the cement stone. By height values of water cement factor the distance between the cement particles getting bigger in the cement adhesive. That leads to the fact, that capillary pore system in the cement stone is not completely closed. On the other hand a higher water cement factor leads to more flow able cement mix. Also a higher wcf leads to a higher hydration rate. The negative influences of higher water cement factor on the concrete properties are more than the advantages. That’s why a target water/cement ratio is specified as 0.40.

Cement content

A minimum cement content is specified at around 300 kg/m^3.

Chloride content

There is need to specify a maximum chloride content provided in the concrete mix. Because off chloride stimulate the corrosion of the reinforcement significantly. Given the great importance of the preventing the reinforcement corrosion, a frequent and repeated control of the chloride content is needed of the all used components in the mixture (cement, mixing water, additives). A maximum chloride content is specified as 0.10% of total powder content to minimize the initial chloride in the concrete.

Alkali silica reaction

To avoid alkali silica reaction a maximum alkali content is specified for an equivalent Na_2O of 3.0 kg/m^3 for a concrete with a mortar content (concrete less coarse aggregate) of 60%.

The above mentioned aspects and given values are used as starting points for the concrete mix design. This values are based on the previous immersed tunnel projects and some adjustments due to the local condition in the Fehmarnbelt. Later on the calculated mix will be verified using the requirements in the Eurocode.

4.2 CHOICES FOR THE DESIGN MIX

Required strength (C/B value)

As mentioned before the strength requirement for the tunnel elements is not determinative. Furthermore, the strength of the young concrete is important for the Fehmarnbelt project, because of the complexity of the construction process. To get an idea what kind of strength criteria for the young concrete should be taken into account some stress calculations are made during the construction process. It's assumed that the internal formwork of the roof slab will be released after 3 days and that the segment will be pushed after 5 days. This operation will influence the required earlier age strength of the poured concrete segments. As a result of the earlier mentioned operation some rapid strength gain is needed. In Table 28 the result of the stress calculation are presented.

The assumptions and boundary conditions for the calculations:

- The friction on the skidding beam is assumed to be 20% of the mass of one segment .
- When removing the internal formwork of the roof slab the weight has to be carried by the walls. The longest span of 11.05 is been taken into account and the determinative dimension are the dimension of the internal walls.
- When the segment is being pushed the force is equally distributed on the whole are of the concrete.

Compressive stress in the concrete area due to pushing activities			Compressive stress in the concrete area due to removing the formwork of the roof slab		
	Value	Unit		Value	Unit
volume concrete one segment	3335,44	m^3	Volume concrete longest span (per meter width)	16,2	m^3
Weight segment	83386	KN	Weight concrete	405	KN
Horizontal force needed for the replacement of the structure 20% friction on the skidding beam.	16677,2	KN	Force due to weight self-weight on each wall	202,5	KN
Stress in the concrete area (Compressive)	0,187	N/mm^2	Stress in the concrete wall due to self-weight	0,608	N/mm^2

Table 28 stress calculation for the young concrete

Environmental class

The tunnel elements in the Fehmarnbelt need be designed as reinforces concrete structure exposed to the sea water. According to the definition of the Eurocode the environmental exposure class can be defined as:

- XS2 → Permanently submerged.

In order to provide more safety and to be sure that all the measures are taken properly it's chosen to apply a stricter environmental class, namely:

- XS3 → Tidal, splash and spray zones.

Consistency class (Workability)

The extent to which the concrete mortar can be handled is described according the concrete regulations rules as consistency aspects. The Eurocode split the handling qualities into 4 consistency areas. As described earlier because of the geometric features of the tunnel elements high flow able mixture is required for the construction. That´s why the designed mixture must fulfil the requirements of the consistency class number 4. This consistency class can be obtained by adding the super superplasticizer.

Choice of materials

Concrete mixture proportioning refers to the process of determining the quantities of concrete ingredients. In general for the complete Fehmarnbelt tunnel project more than 3 million m^3 concrete is needed. Using local materials for concrete production is an interesting option from the economic en logistical point of view. Especially aggregate forms a large part of the concrete volume.

Using the local aggregate recourses reduces transportation costs and energy expended in moving heavy bulk materials to the project area. Optimal use of local aggregates also reduces truck traffic and the number of axle loadings on the highway system to the project area. In the project area there is no suitable aggregate material available on great scale. But relatively nearby in Norway there are many companies which can deliver aggregate of the required quality. Norway produces different high quality aggregate material for concrete. For the coarse aggregate two options are considered:

- Crushed stone
- Gravel

Both materials are available from good quality with different characteristics. The crushed stone aggregate for use in the concrete mix, is extracted from bedrock by blasting. The material is crushed and sorted to the most appropriate size for use in concrete mix. The gravel could be extracted from the natural gravel deposits. Figure 25 indicates the natural deposits of the aggregate material. Especially the deposits in south Norway nearby the German and Danish coast are interesting for the Fehmanrbelt project. In general the crushed-rock aggregate is expensive due to the cost of blasting and crushing compared to the natural gravel aggregate.

Depending on the requirements for the concrete mix the appropriate aggregate can be chosen. Mostly the crushed stone aggregate is used in combination with the height strength of the cement to obtain height strength concrete. Different sizing composition of the course aggregate can be delivered by the producers.

The roundness of the aggregate material is important too. The crushed stone aggregate which are more edgy than the gravel aggregate needs more water and cement compared to the rounded gravel aggregate to get equal consistency level. On the other hand the tensile strength of the concrete in which crushed stone aggregate has been used is larger than the tensile strength of the concrete mix with the gravel aggregate.

It has to be mentioned that some rock types are also not suitable to use as coarse aggregate in concrete. The rock types which cause Alkali reaction in the concrete are generally not suitable to use as aggregate in concrete. In southern of Norway there are different companies which can deliver nonreactive stones for the concrete aggregate. The appropriate stones and some important characteristics are given in Table 30.

Gravel and sand from marine location could be also used as aggregate for concrete. But this requires pre-treatment of the aggregate. The marine material should be treated with fresh water to wash out the south. Also the aggregate materials should be pre-treated to minimize the amount of shells in the material.

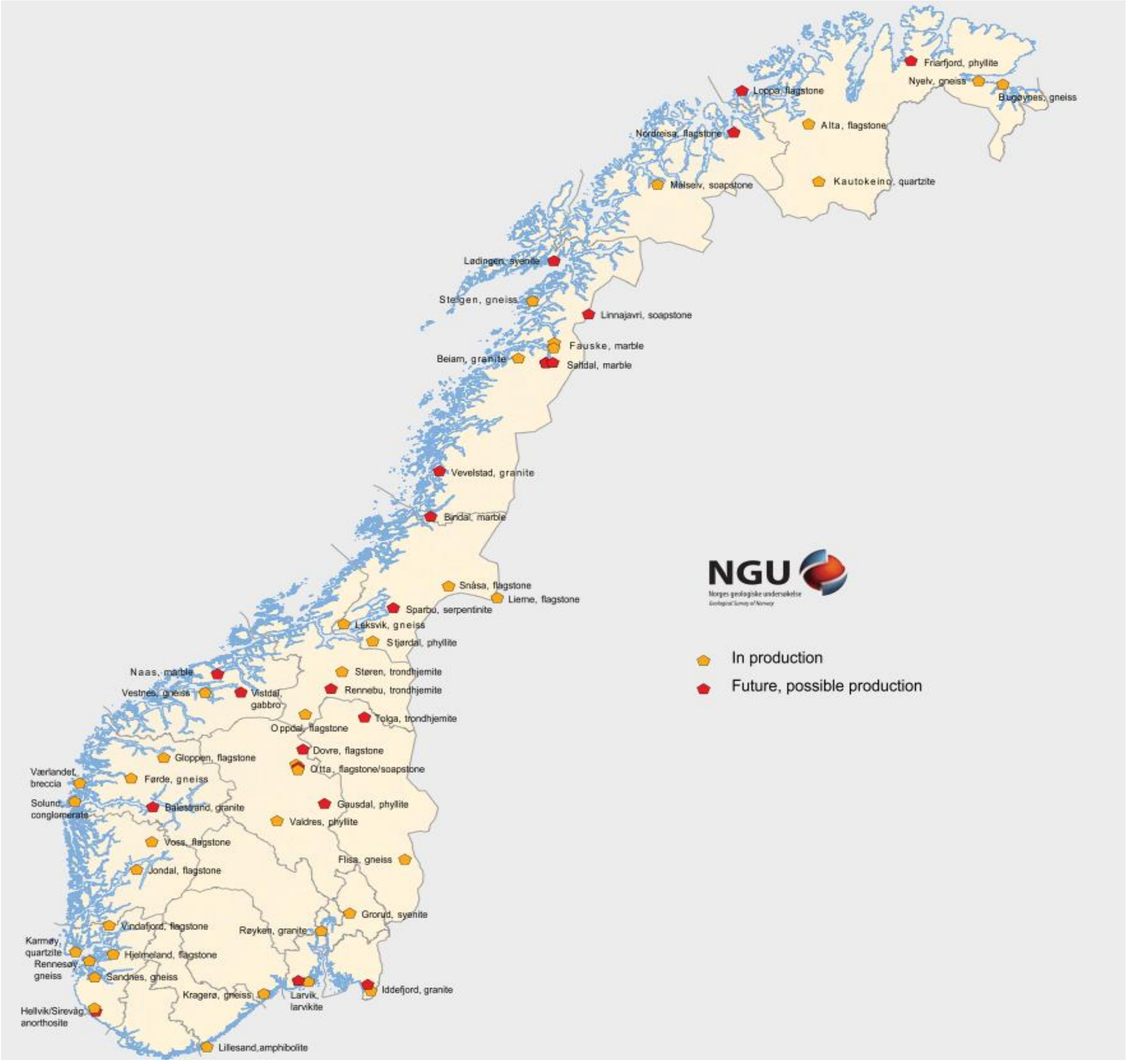


Figure 25 Sand, gravel and aggregate deposits

Determination of composition of cement paste

In order to choose the proper cement for the concrete mix design the following 3 cement types has been considered:

1. Blast-furnace cement (hc)
2. Portland cement (pc)
3. Portland fly-ash cement (pvlc)

Between the last two mentioned cement types there are no important differences. But between the hc and pc there exist some important differences. The differences are mainly related in the slag levels above the 60%. In order to choose the proper design mix the following differences between the pc and hc are taken into account:

1. As a consequence of difference in the chemical composition between the pc and hc the cement stone of the hc with equal degree of hydration has a much denser pore structure than the cement stone of the pc. This difference is very important for the durability aspects. This leads to much lower rate of penetration of the harm full materials in the concrete such as: chloride and sulphate which occur in the saline water.
2. At the normal ambient temperatures especially in the beginning after pouring the concrete the hydration reaction of the slag is much slower than the hydration reaction of the clinker. Therefore the young age strength of the hc concrete is much lower than the young age strength of the pc concrete for the same strength class.
3. The ambient temperature influence on the hydration reaction for the hc concrete is greater than the pc concrete. Especially during the low temperatures hc concrete needs more hardening time than pc concrete. Therefore a concrete mix with hc needs more time in order to be able removing the formwork. Contradictory to the previous attribute the hc concrete hardens much faster that pc concrete when the ambient temperature is high. Concluded form the previous the hc concrete needs more temperature regulations that than a pc concrete to get a proper hydration reaction.
4. As a result of the slower hardening process of the hc concrete it requires more after care treatment than the pc concrete when concrete has been poured. If the poured concrete is not treated with care then the risk of the poor surface quality for the hc concrete is greater than for the pc concrete. The durability advantages for hc concrete are only obtained when it has been treated with care after the pouring. Generally one can say that the hc concrete needs more aftercare treatment than the pc concrete.
5. The hydration heat of the hc is less than of pc. That´s why the hc with sufficient slag percentage usually meets the requirements of cement with low hydration heat. Such cement is usually used when there is a risk of cracking due to temperature stresses.
6. Concrete prepared with the hc shows a dark blue colour after removing the formwork. The dark blue colour disappears when the poured concrete is in touch with air. In the beginning this dark blue colour is good indication to check if the poured concrete has been treated with enough care.

In order to choose the proper cement type for the concrete mix the above mentioned differences has been taken into account. The cement type used greatly influences the processing and concrete properties. Due to the dense pore structure of the cement stone and the low hydration heat the blast-furnace cement has been chosen for the concrete mix. This means that attention should be paid to good temperature regulations during the pouring and a good after treatment in order to get a proper quality concrete mix.

Blast-furnace cement consist mainly from Portland cement clinker, aggregated blast furnace slag and natural puzzolanas. The slag has more a slower hydration reaction than the clinker. That’s why for cement type 3 with low hydration reaction has been chosen. According to the Eurocode the cement type III/B LH has a percentage of:

- Portland cement clinker (20-34)
- Aggregated blast-blast furnace slag and natural puzzolanas (66-80)

Previously has been explained that a durability class XS 3 has been chosen for the concrete. The minimum cement content according to the Eurocode EN 206:2013 is: 340 kg /m^3 to meet the required durability requirement. The strength indications for the chosen cement type are given in Table 29.

Type of cement	Code name	Norm strength N of cement [N/mm^2]			
		1 day	2 days	3 days	28 days
Portland Cement	CEM I 32.5 R CEM I 42.5 R CEM I 52.5 R	10 19 29	17 30 39	25 35 44	48 58 63
Portland flyash cement	CEM II/B V32.5 R	13	22	25	49
Blast Furnace cement	CEM III/A 32.5	7	14	19	46
	CEM III/A 42.5	8	17	22	59
	CEM III/B 32.5 LH	5	10	14	48
	CEM III/B 42.5	8	17	25	58

Table 29 strength values different cements

Air content

Since the tunnel elements are always deep under water suited no freezing requirements has been taken into account. Despite of the compaction method, there is always entrained-air content in the concrete. It's been assumed that the concrete will be sufficiently compacted during the pouring. An air content of 1% is been taken into account.

Aggregate

Two aspects of aggregates have an important influence on properties of the concrete mix which affects the workability of the fresh concrete.

1. Grading (particle size and distribution)
2. Nature of particles (shape, porosity, surface texture)

Aggregate grading is important for attaining an economical mixture. It affects primarily the amount of the required cement and subsequently the needed amount of water.

Coarse aggregates requires less cement, that's why is economically it's interesting to choose the large particle size as large as possible. On the other side all the aggregate should be able to pass the space between the reinforcement. That's why the largest size has been chosen according the Eurocode sieving size 31.5 mm. It's assumed that the there is enough space for the coarse aggregate of size 31.5 mm to pass the space between the reinforcement bars. Grading also influences the workability and placeability of the concrete. The aggregate with enough fine material is stable and more workable. Durability may also be affected by the grading size. Various options for choosing the grain size are available to obtaining the optimal grading. For the calculated concrete mix the grading size of the chosen aggregate is equal to the sieving size of 31.5 mm.

The amount of mixing water required to produce a unit volume of concrete of a given consistency class is dependent on the shape and the maximum size and amount of coarse aggregate. Larger sizes grains minimize the water requirement and thus allow the cement content to be reduced. Also the cement content is dependent on the shape of aggregate. Rounded aggregate requires less mixing water than a crushed stone aggregate of equal slump. Crushed stone aggregate requires approximately 10 litters more water per m^3 for equal slump size.

Coarse aggregate can be distinguished by its origin and its density. For the calculation in it is been assumed that the aggregate consist from river sand and river gravel. Since relatively nearby the project area the natural deposits for gravel and sand are available (see Figure 25). The use of the gravel instead the crushed stone aggregate is preferred.

Generally it's cheaper compared to the crushed stone aggregate. Also the workability and the required amount of cement and water for the gravel aggregate are lesser than for the crushed stone aggregate. For very high strength concrete it is recommended to use the crushed stone material rather than gravel. But as explained earlier, the strength requirements for the tunnel elements are not normative. The durability requirements are normative for the tunnel elements. All these aggregate must fulfil the requirements given in the Eurocode. Namely the requirements in the NEN 5905 and NEN-EN 12620+A1.

Many factors which determines the aggregate choice has to be investigated more in detail. It's still possible that the crushed stone aggregate can be economically and technically a proper choice than the gravel aggregate. That's why also the concrete density has been calculated for the different stone types which occur nearby the project area.

In Table 30 are some properties of the possible aggregate materials for the normal concrete are presented and the range of the properties.

Coarse aggregate	Bulk density	Water absorption after 30 min. in water (%m/m)	Grain Strength (N/mm^{22})
'River' gravel	2500 - 2700	0,8 - 1,2	150 - 200
Porphyry	2550 - 2800	0,4 - 0,6	180 - 300
Gabbro/diorite	2900 - 3000	1,0 - 1,1	170 - 230
Limestone	2650 - 2850	0,9 - 1,2	80 - 230
Sandstone/quartzite	2600 - 2650	0,2 - 1,2	150 - 300
Basalt	2850 - 3000	Very low	250 - 400
Granite	2600 - 2800	0,5 - 0,7	160 - 260

Table 30 Properties of some common coarse aggregates for normal concrete

The granular structure of the coarse aggregate for the concrete is of great importance. The better the grain sizes, the less cement and water are required. This is not only of importance for the price of concrete, but also for various other features such as the degree of shrinkage, creep and the effects on durability. For the mixture calculations a continuous distribution of the aggregate material has been assumed. Also a discontinue material distribution could be applied to the concrete mixture. In the latter case the grains are placed closer to each other and less water and cement will be needed. The disadvantage is of discontinues mixture is that the concrete it's difficult to compact and the concrete the mixture is more sensible for the segregation than a mixture with a continuous grain distribution. Also in the nature the aggregate material occurs in a continuous distribution. So the waste of the material is also less.

In the previous Dutch Regulations VBG the limits of the size distributions of the aggregate material were given for concrete technology. The limits of the size distributions of the aggregate material for the nominal grain size 31,5 mm are given in Figure 26. In any case the mixture must fit between the under and upper line.

In the current applications of the Eurocode NEN-EN 206-1 and NEN 8005 there are no requirements for the limits of the aggregate mixture. It's recommended to choose an aggregate mixture which fits between the boundaries of the curves. The aggregate curves in area I are preferred above area II.

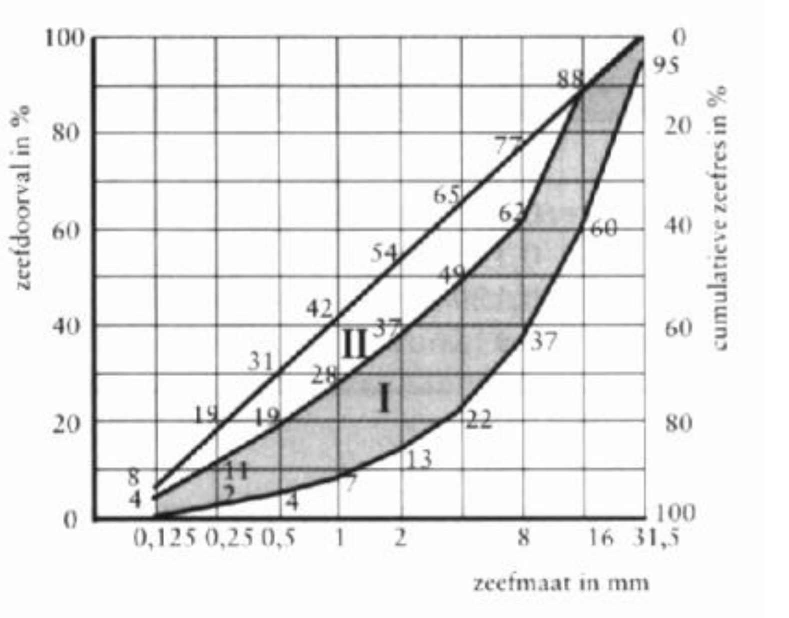
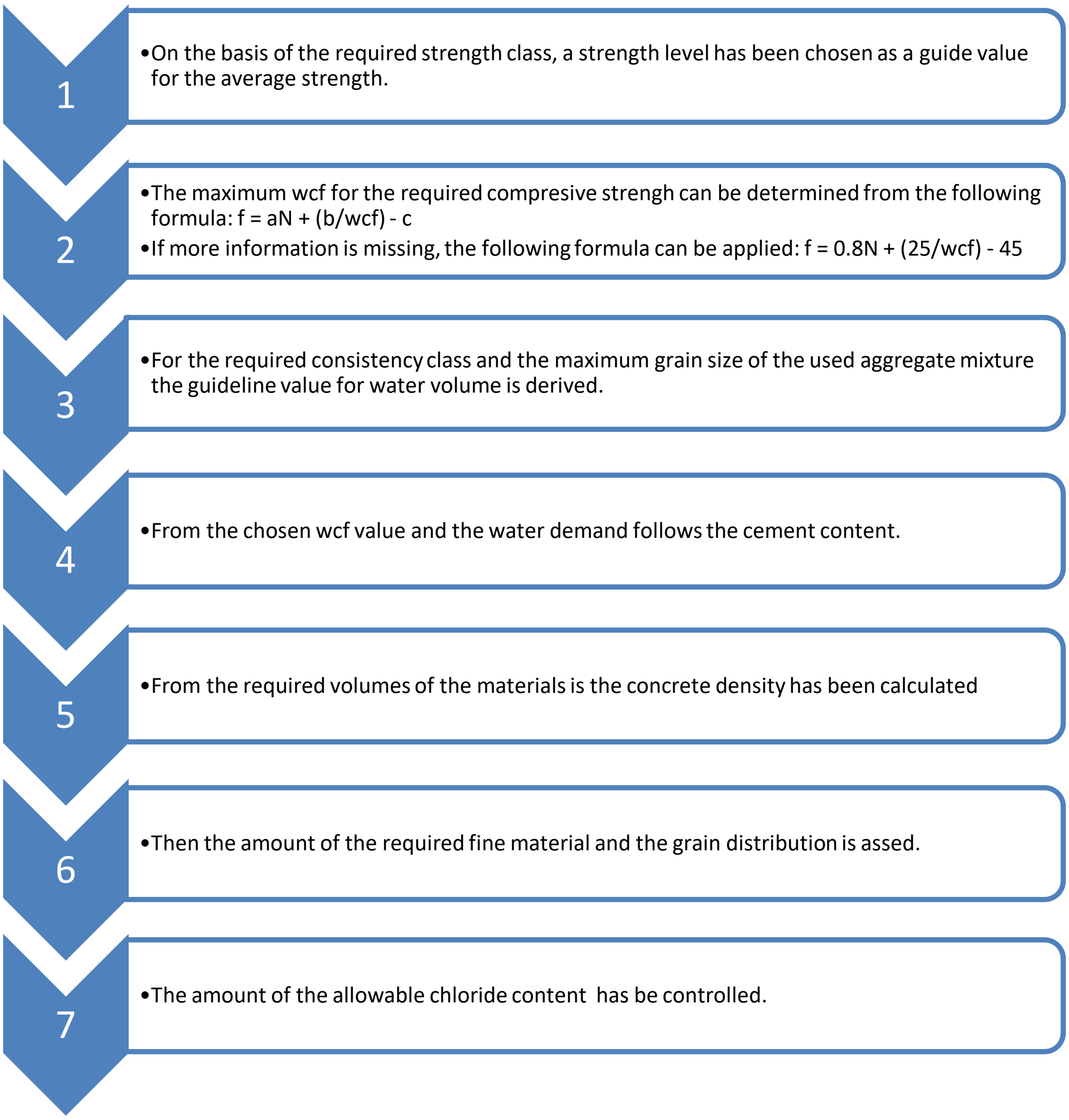


Figure 26 Design Area for grain group 0/32 mm

Mix design calculation

In order to calculate the density of the design mix is the following scheme has been followed.



1

- On the basis of the required strength class, a strength level has been chosen as a guide value for the average strength.

2

- The maximum wcf for the required compresive strengh can be determined from the following formula: f = aN + (b/wcf) - c
- If more information is missing, the following formula can be applied: f = 0.8N + (25/wcf) - 45

3

- For the required consistency class and the maximum grain size of the used aggregate mixture the guideline value for water volume is derived.

4

- From the chosen wcf value and the water demand follows the cement content.

5

- From the required volumes of the materials is the concrete density has been calculated

6

- Then the amount of the required fine material and the grain distribution is assed.

7

- The amount of the allowable chloride content has be controlled.

Figure 27 Mixed design scheme

The calculations take 1 m^3 concrete into account. The density of the concrete will be calculated based on the formula:

$$\rho_{concrete} = \sum \rho_i * V_i \qquad (79)$$

In which:

ρ_i Mass density of ith component in the concrete mix

V_i Volume of ith component in the concrete mix

The mass will be expressed in kg and the volume will be expressed in m^3. For the values of the mass density of the base materials are the following values has been used:

Raw material	Mass density	Units
Blast furnace cement	2950-3000	[Kg/m^3]
Sand	2600-2650	[Kg/m^3]
Gravel	2500-2700	[Kg/m^3]
Water	1000	[Kg/m^3]
Porphyry	2550-2800	[Kg/m^3]
Gabbro/diorite	2900-3000	[Kg/m^3]
Sandstone/quartzite	2600-2650	[Kg/m^3]
Basalt	2850-3000	[Kg/m^3]
Granite	2600-2800	[Kg/m^3]

Table 31 Mass densities bass materials

On the basis of the required strength class C45/55, a strength level has been chosen as a guide value for the average strength. There for a concrete strength of 56 N/mm2 is needed to meet the requirements for the strength class C45/55. The maximum value of the wcf for the compressive strength 58 N/mm2 can be calculated for the formula:

$$f_c' = aN + \frac{b}{wcf} - c \qquad (80)$$

In which:

f_c' Norm strength of cement after n days

a 0.80 or 0.85 dependent on the chosen cement

N Norm strength of cement after n days

b 20 or 25 dependent on the chosen cement

c 40 or 50 depending on the nature of the used cement, the influence of the temperature of the mixture on the final strength and the kind of the used aggregate.

From the calculation it follows that the maximum allowable wcf is 0.44 to meet the strength requirements. As explained earlier the target value for the maximum wcf is chosen to be 0.4. So the value of 0.4 is normative and will be further used in the calculations.

The indicative water need can be estimated from the Table 32. The values for the water need for the consistency class 2 are used. As explained before the extra fluid ability of the concrete mix will be obtained by means of adding the superplasticizer. The chosen value for the water need is: 170 lit/m^3.

	Guideline value for water need of 1 m^3 concrete mortar (litters/m^3)							
Maximum grain size D_{max}	8		16		31.5		63	
Grading area	A-B	A-C	A-B	A-C	A-B	A-C	A-B	A-C
Consistency class 1	175	195	160	180	150	170	140	155
Consistency class 2	192	213	180	200	165	185	155	170
Consistency class 3	205	225	195	218	180	200	168	190

Table 32 Guideline for water needs (from the book Betontechnologie C. Souwberben 1991)

The cement content must be:

$$\frac{Massa\ water}{massa\ cemetn} = water\ cement\ ratio = \frac{W}{C} = wcf \tag{81}$$

With a target value of wcf of 0.4 the required amount of cement must be C=425 kg/m^3. This calculated value satisfies the minimum required cement content according the Eurocode. By using the formula of Rangers-Antonisse the amount of the needed amount of sand can be determined:

$$P_s = 10F_s + 28 + 0{,}05Z - 0{,}08C \tag{82}$$

In which:

F_s	Finess modulus of the sand (2.94)
Z	Slump value
C	cement content

From the calculations follows a sand percentage of 33%. The percentage of gravel can be determined from the formula:

$$P_g = 100 - P_s \tag{83}$$

The percentage of the needed gravel is: 67%. Volumes of material used for 1 m^3 concrete are given in Table 33.

	Gravel aggregate		Crushed stone aggregate	
	Value	**Unit**	**Value**	**Unit**
Cement	0,144	m^3	0,153	m^3
Water	0,170	m^3	0,180	m^3
Air	0,010	m^3	0,010	m^3
Sand Aggregate	0,221	m^3	0,202	m^3
Coarse Aggregate	0,455	m^3	0,456	m^3
Total Volume	1	m^3	1	m^3

Table 33 Volumes base material in 1 m^3 concrete.

Based on the volumes mentioned in Table 33is the mass density of concrete has been calculated for different coarse aggregate. Different values for the water content and cement content are used for different kind of coarse aggregate. For the crushed stone is 10 litter more water per 1 m^3 is applied in the calculations. The different water content leads also to different cement content and the amount of aggregate. No replacement materials are used for sand. The fine aggregate sand is difficult to replace by other material. The unreinforced density of concrete for different coarse aggregate types is given in Table 34.

Unreinforced Concrete density		
	min	max
Gravel	2306,90	2416,19
Porphyry	2291,59	2473,69
Gabbro/diorite	2451,17	2564,88
Sandstone/quartzite	2314,40	2405,30
Basalt	2428,38	2564,88
Granite	2314,40	2473,70

Table 34 Unreinforced concrete density

In order to check the stability of the concrete mix, the amount of the fine material is checked. To be able to check the amount of the fine material a grading curve for the aggregate material has to be assumed in this stage of the project. If the grading of the different aggregate material is known these calculation has to be redone. For the grading of sand and other coarse aggregate sieve according to NEN 2560 is been assumed.

Sieve according NEN 2560	Cumulative sieve residue [%]	
	Fine aggregate (sand)	Coarse aggregate
C 31.5	--	--
C 16	--	28
C 8	--	70
C 4	3	94
2 mm	9	100
1 mm	28	100
0.5 mm	60	100
0.25 mm	94	100
0.125 mm	100	100
Fineness modulus F	2.94	100

Table 35 Sieve according NEN 2560

Given the sand percentage and the gravel percentage the sieve according NEN 2560 should be:

Sieve according NEN 2560	Cumulative sieve residue [%]				
	Fine aggregate (sand)	Coarse aggregate	Sand 32.65%	Gravel 67.35%	Mixture curve
C 31.5	--	--	--	0	0
C 16	--	28	--	18,86	18,86
C 8	--	70	--	47,15	47,15
C 4	3	94	0,98	63,31	64,29
2 mm	9	100	2,94	67,35	70,29
1 mm	28	100	9,14	67,35	76,49
0.5 mm	60	100	19,59	67,35	86,94
0.25 mm	94	100	30,69	67,35	98,04
0.125 mm	100	100	32,65	67,35	100
Fineness modulus F	2.94	100			

Table 36 Sieve according NEN-EN 2560

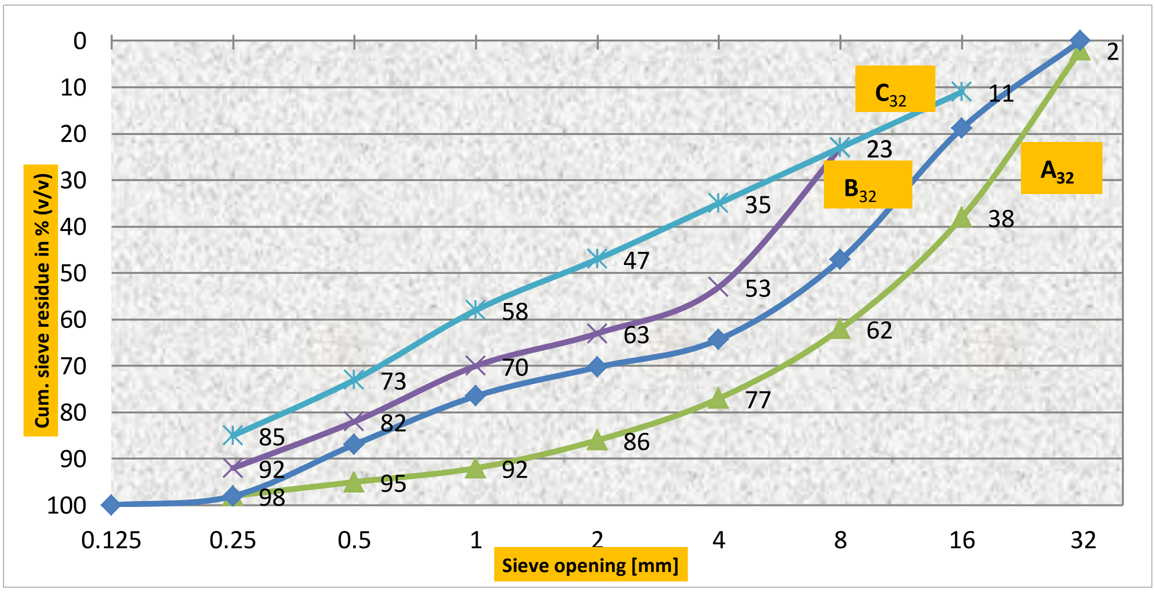


Figure 28 Sieve curve of the mixture with sand aggregate

In Figure 28 the mixture curve is indicated with the blue line. Form the figure it is obvious that the mixture curve fits between the predefined limit lines A-C and also the curve fits in area I A-B. In figure Figure 29 is the sieving curve for the crushed stone aggregate depicted. It must be noted that the in the calculations it is assumed that both the crushed stone aggregate and the gravel aggregate have the same grading sizes but in different quantities.

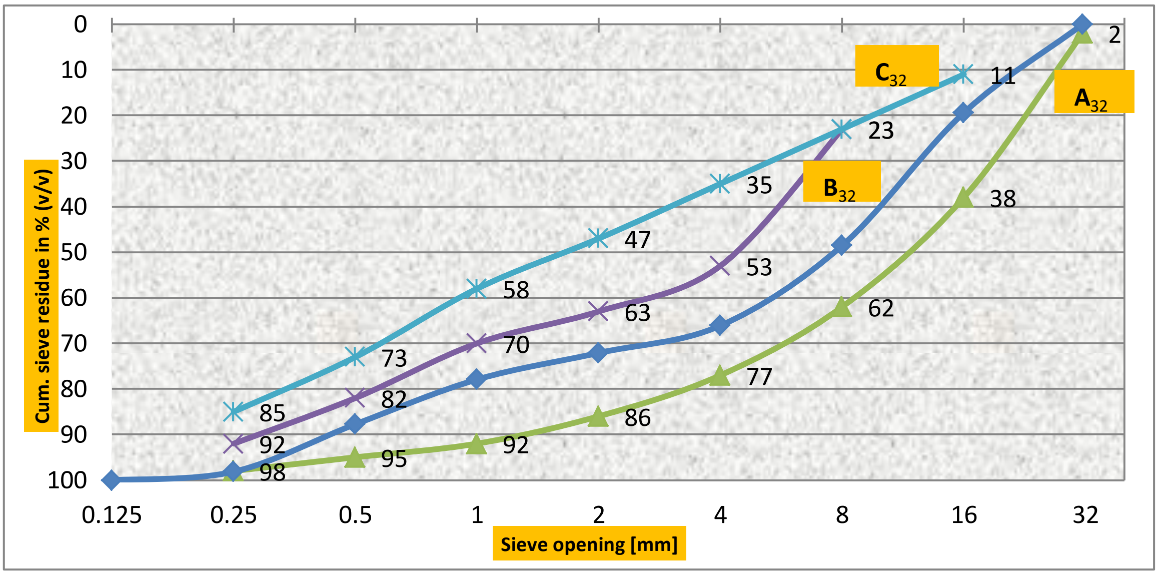


Figure 29 Sieve curve of the mixture with crushed stone aggregate

Amount of fine material

The volume cement in the mixture is:

$$V_{cement} = \frac{M_{cement}}{\rho_{cement}} \tag{84}$$

The volumes of cement in a mixture with crushed stones and gravel are:

$$V_{cedment} = 144\ lit.\ \text{(gravel)}$$
$$V_{cemetn} = 152\ lit.\ \ (crushed\ stone)$$

The amount of fine sand < 250 µm is:

$$Volume\ fine\ sand\ <\ 250\ \mu m = \frac{\left(\frac{6}{100}\right) * M_{sand}}{\rho_{sand}} \tag{85}$$

The volumes of fine sand in a mixture with crushed stones and gravel are:

$$V_{fine\ sand<250\ \mu m} = 13.2\ \ lit.\ (gravel)$$

$$V_{fine\ sand<250\ \mu m} = 12.7\ \ lit.\ (crushed\ stone)$$

Total volume fine material for in the concrete mixtures:

$$V_{fine\ <250\ \mu m} = 157.3\ \ lit.\ (gravel)$$

$$V_{fine\ <250\ \mu m} = 165.2\ \ lit.\ (crushed\ stone)$$

In Eurocode is a minimum amount of fine material < 250 µm is required to be minimal 115 litters for a nominal diameter D_{max} of 31,5 mm. The calculated mixture satisfies this amount. In order to access whether the concrete mixture meets the requirements a suitability test has to be done in the laboratory. There by different aspects as consistency, compressive strength and chemical composition of the materials should be tested. If all tests give, sufficient result then the mixture is of sufficient quality. Otherwise the mixture has to be adjusted such that it can meet the requirements.

5 ANNEX THEORETICAL BACKGROUND

5.1 STATIC STQABILITY

This part of the report discuss only the static properties of the system. For the calculations in this part it's been assumed that any disturbance to the equilibrium state will be brought so slowly that all dynamic effects can be ignored. In different conditions the floating structures (pontoons + tunnel element) will experience many external and internal loads (from equipment, ballast water, wind and waves) trying to turn the floating structure. Each floating structure must be able to resist these turning over via its static stability.

Static stability is a measure or the tendency of each floating structure in the system to return to its upright configuration when the external/internal load which cause to incline the structure from its upright position will be removed. To little stability is undesirable because each small turning-over moment will cause that the floating structure will be capsized. Too much stability is also undesirable because static stability affects the natural roll and pitch frequency of the floating structure in the system. Beside of that too much stability can be costly too.

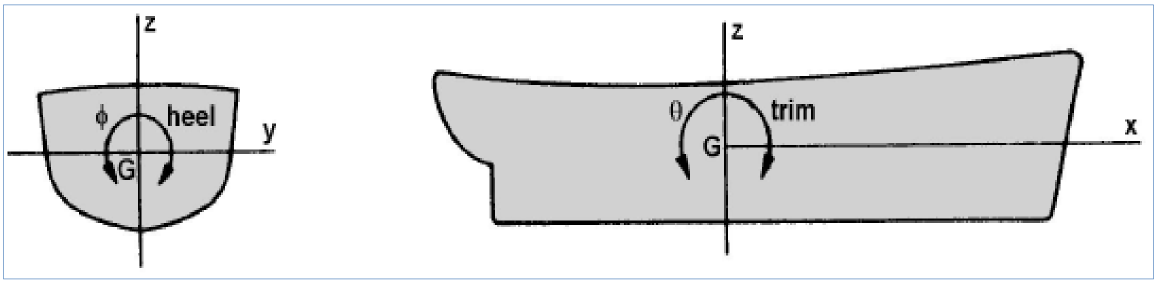


Figure 30 Notation used for the calculations

Theory Static Stability

The total weight of the floating body passes through its centre of gravity (G or CoG). The buoyancy force F_B acting on the floating body passes through the centre of buoyancy B, which corresponds to the centre of the displaced fluid. When the floating body is subjected to a heeling moment M_H it will heel with an angle ϕ. As result of heeling of the structure, the underwater shape will be changed. The centre of buoyancy will shift from B to B_φ, while the centre of gravity of the floating body remains unchanged at G. An equilibrium will be achieved when the righting moment M_S equal the external heeling moment M_H (see Figure 31).

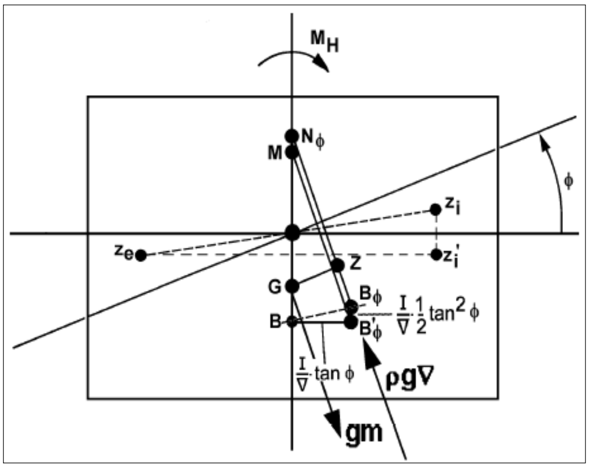


Figure 31 Rotational Equilibrium at heel angle

In formula form the equilibrium can be given as:

$$\mathrm{M_S} = \mathrm{M_H} \quad (86)$$

$$\mathrm{M_S} = \rho g \nabla * \overline{\overline{\mathrm{GZ}}} \quad (87)$$

$$\overline{\overline{\mathrm{GZ}}} = \overline{\mathrm{GN}} * \sin(\varphi) \quad (88)$$

$$\mathrm{M_S} = \rho g \nabla * \overline{\mathrm{GN}} * \mathrm{si\ n}(\varphi) \quad (89)$$

A vertical line drawn upward from B_φ intersects the line of symmetry at N_φ, known as the metacentre. GN_φ is known as the metacentric height. The position of the metacentre depends on the new position of the centre of buoyancy B_φ. Thus also on the shape of the structure and the water plane area. It can be said that the heel angel ϕ and the shape of the immersed and the immerged shapes controls the position of the metacentric point.

The floating structures which are studied in this report (pontoons and the tunnel element) are wall-sided structures. This means that the wall sides which are covered or uncovered by the changing water plane area are vertical. Regardless from the underwater geometry for all wall-sided structures the distance BN_φ can be represented by the Scribanti Formula:

$$\overline{\mathrm{BN}_\varphi} = \frac{\mathrm{I_T}}{\nabla} * (1 + \frac{1}{2}\tan^2(\varphi)) \tag{90}$$

$\mathrm{I_T}$ is the transverse moment of inertia of the not heeled water plane about the axis of the inclination for the both half of the water planes. ∇ is the displaced volume of water. This formula is valid till the water plane area doesn't change very rapidly. When the bilge comes out from the water or the deck enters the water the above given formula is not valid for the calculations.

Both the tunnel element and the pontoons are assumed to be symmetric structures for the calculations. This means that the meta centre is suited in the middle line plane for both disturbance heel and trim. (ϕ and θ = 0). The stability lever arm $\overline{\overline{GZ}}$ is determined by the hydrostatic properties of the submerged structure and the mass distribution of the structure. The distance $\overline{GN}$ can be also expressed as:

$$\overline{\mathrm{GN}_\varphi} = \overline{\mathrm{KB}} + \overline{\mathrm{BN}_\varphi} - \overline{\mathrm{KG}} \tag{91}$$

$$\overline{\mathrm{GN}_\varphi} = \overline{\mathrm{KB}} + \overline{\frac{\mathrm{I_T}}{\nabla} * (1 + \frac{1}{2}\tan^2(\varphi))} - \overline{\mathrm{KG}} \tag{92}$$

In this equation the symbol K represent the keel point of the floating structure. The expression KB can be determined from the underwater form of the submerged structure. The expression KG can be determined from the mass distribution of the floating structure.

Submerged Structure

The tunnel element during the installation will have no water plane area. The definition $\overline{BN_\varphi}$ is zero for the fully submerged structure. This means that the meta centre of the fully submerged tunnel element will coincide the centre of buoyancy. For this case the expression for the stability lever arm $\overline{GN_\varphi}$ reduces to:

$$\overline{\mathrm{GN}_\varphi} = \overline{\mathrm{GM}} = \overline{\mathrm{KB}} - \overline{\mathrm{KG}} \tag{93}$$

For the static stability equilibrium the following three situations can be distinguished:

(a) If N_φ is above G (GN_φ > 0), a restoring couple acts on the floating body in its displaced position tending to restore it to its original position. Hence, the body is in stable equilibrium.

(b) If N_φ is below G (GN_φ < 0), an overturning couple acts on the body. Hence, the body is in unstable equilibrium.

(c) If $N_\varphi$coincides with G (GN_φ = 0), the resultant couple is zero, and the body has no tendency to return to, nor move further away from its original position. Hence, the body is in neutral equilibrium.

If a floating body floats stable, then the natural frequency for the roll and pitch motions can be given as:

$$T = 2\pi * \left(\frac{k_{xx}(\text{or})\ k_{yy}}{\sqrt{g * GN_{\varphi}}}\right) \tag{94}$$

Where $k_{xx}(and)\ k_{yy}$ are the radii of gyration of the floating body about its longitudinal and transversal axis. From the formula it can be concluded that larger values of GN_{φ} decrease the natural periods of pitch and roll. Hence larger values of GN_{φ} give rise to more rapid oscillations. However, the larger values of GN_{φ} give more static stability to the floating structure. The above two requirements are conflicting for the choice of GN_{φ}. A good design should thus entail adequate but not excessive values of GN_{φ}. For this case study it's been assumed that the value of GN_{φ} should at least must be 1 m.

Arbitrary Loading

When a load is placed or added on an arbitrary point of the floating structure, the static response of the floating structure can be split into two components:

- The structure will sink deeper parallel to the original water plane
- The floating structure will rotate (heel and/or trim)

The parallel sinkage is caused by the increased total mass, the rotation is caused by the generated moment. The moment is caused by the shifted centre of gravity and centre of buoyancy. The vertical positions of the centre of buoyancy and centre of gravity will be changed. The new position of the structure after placing a load can be determined in tow steps.

1. The additional mass P must be placed above or under the centre of the water plane in a horizontal plane. The increased draft can be calculated as:

$$\Delta T_0 = \frac{P}{\rho A_{WL}} \tag{95}$$

The result of this step is that the floating structure will sink deeper parallel to the water plane area. The centre of buoyancy and the centre of gravity will be shifted because of the added submerged volume and added mass P. The related shifts of the centre of buoyancy and the centre of gravity can be calculated from the first moments of volumes and masses. For the tunnel element and the pontoons is the structure hull symmetric with respect to a plane through the points $G_{0,}$ B_0 and M_0, so the centre of the water plane lies in the symmetry plane too. The horizontal shift of the centre of buoyancy and the centre of gravity will be zero. The initial and the later metacentre is are given by equation(96)and (97)

$$\overline{B_0M_0} = \frac{I_T}{\nabla_0} \tag{96}$$

$$\overline{BM} = \frac{I_T}{\nabla_0 + \Delta\nabla_0} \tag{97}$$

2. The second step is this calculation is: to shift the mass P in a horizontal direction over a distance c to its actual position and adding a heeling or a trimming moment M_H to the calculation. The moment M_H depends on the angle of heel or trim. The moment M_H can be given as:

$$M_H = P * g * c * \cos(\varphi) \tag{98}$$

The heeling moment M_H must be equal to the righting stability moment M_S

$$M_H = M_S \tag{99}$$

$$g * P * c * \cos(\varphi) = g * \rho\nabla * \overline{GN_{\varphi}} * \sin(\varphi) \tag{100}$$

The heel angle ϕ or trim angle θ follows from the moment equilibrium presented in equation (101):

$$\varphi = arccos\left(\frac{\rho\nabla * \overline{GN_{\varphi}} * sin(\varphi)}{P * c}\right) \tag{101}$$

For the wall sided structures as pontoon and the tunnel element the rotation angle ϕ or θ can be presented as:

$$\varphi = \arccos\left(\frac{\rho\nabla * \left\{\overline{GM} + \frac{1}{2}\overline{BM} * \tan^2(\phi)\right\} * sin(\varphi)}{P * c}\right) \tag{102}$$

By using tan(ϕ) we can write equation (102)) as:

$$\frac{1}{2}\overline{BM} * tan^3(\varphi) + \overline{GM} + \tan(\varphi) = \frac{P * c}{\rho\nabla} \tag{103}$$

Equation()) can be solved iteratively by using the Regula- Falsi Method.

Free Surface correction

Free surface fluid inside the floating structure will have influence on the static stability of the floating structure. The free surface fluid in the floating structure reduces the righting stability moment M_s or stability lever arm GZ. The underwater geometry o the entire structure and the boundaries of the wedges at the water plane as well in the ballast tank play a role. They determine together the angle of heel/trim and the shifts of the centres of gravity and buoyancy. The principle of the

The tunnel element will be immersed by using the ballast tanks. For the calculations it's assumed that the tanks are wall sided. The shifts if the centres of buoyancy and gravity can be calculated with the aid of the first moment of volumes with the triangular cross sections. The righting moment reduction can be determined by calculating the reduction of the lever arm GZ. The reduction of the stability lever arm is given by the equation(105)

$$\overline{GG''} * \sin\varphi \tag{104}$$

$$\overline{GG''} = \frac{\rho' i}{\rho\nabla} * (1 + \frac{1}{2} * tan^2(\varphi)) \tag{105}$$

The magnitude $\overline{GG''}$ is called the free surface reduction or the reduction of the matacentric height. For small angles of heel or trim the expression in equation 20 for the reduction of the metacentric height can be simplified by neglecting the $tan^2(\varphi)$ term. The simplified expression for the free surface correction is given by:

$$\overline{GG''} = \overline{GG'} = \frac{\rho' i}{\rho\nabla} \tag{106}$$

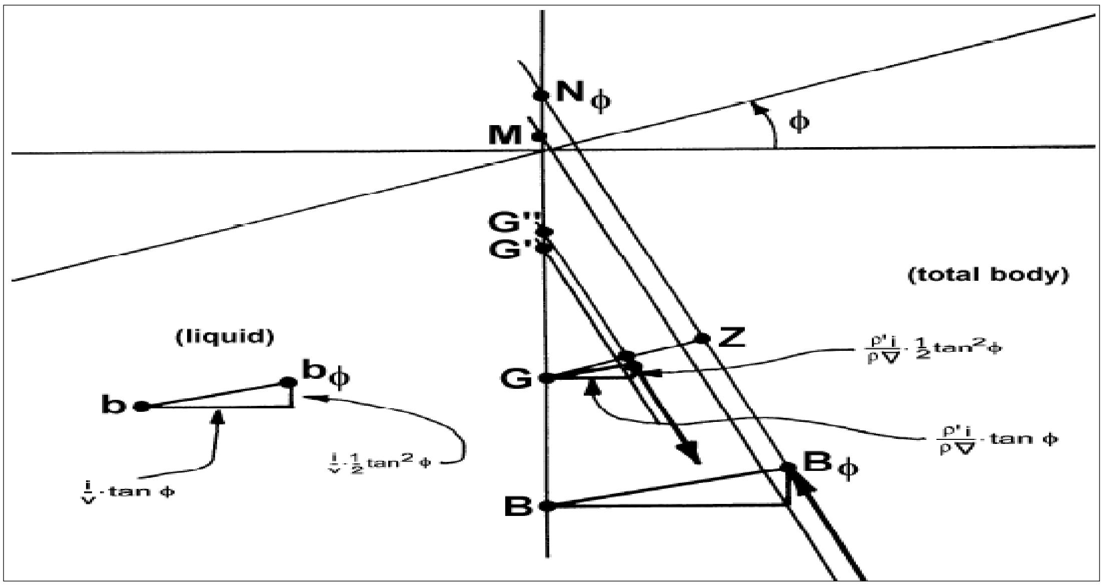

Nϕ
M
ϕ
G''
G'
(total body)
(liquid)
bϕ
b
Z
G
$\frac{\rho' i}{\rho \nabla} \cdot \frac{1}{2} \tan^2 \phi$
$\frac{\rho' i}{\rho \nabla} \cdot \tan \phi$
$\frac{i}{\nabla} \cdot \frac{1}{2} \tan^2 \phi$
$\frac{i}{\nabla} \cdot \tan \phi$
Bϕ
B

Figure 32 principle of the metacentric height reduction

The principle of the metacentric height is presented in (107)During the immersion there will be more ballast tanks in the tunnel element. The expression for more than one ballast tank is given by:

$$\overline{GG''} = \frac{\sum \rho' i}{\rho \nabla} * (1 + \frac{1}{2} tan^2 \varphi) \qquad (107)$$

It must be noted that the position of the tank doesn't influence the reduction of the metacentric height. Only the transverse moment of inertia of the surface in the tank counts.

5.2 WAVE KINIMATICS

To be able to model the problem, first a bunch of theoretical knowledge is gained which can be used for modeling. This part of the report is not meant to describe the complete theory. This part provide an inside in the choices which are made in the model. For the complete theoretical knowledge see the references.

Waves

The waves in the model will be described by the aid of linear wave theory.Linear wave theory is based on set of two equations:

- mass balance equation
- and a momentum balance equation.

These equations describe the kinematic and the dynamic aspects of waves ocean. Linear theory is applicable when the amplitude of the waves is small compared to water depth and wave length. In this case the nonlinear wave effects can be neglected. It's also assumed that water is ideal fluid. Which means that it can be assumed as incompressible, constant density, no viscosity and that the water particles can't rotate around their own axes.From the mass balance equation thecontinuity equation can be derived.

To solve the continuity equation use is made of the velocity potential function $\Phi = \Phi(x, y, z, t)$. The potential function has a property that spatial derivatives in a arbitrary point in the flow field are equal to the velocity of the water particles in that point.

The profile of a wave which has a small steepness will be similar to a sine or a cosine and the motion of a water particles in a wave depends on the distance below the still water level. For this reason a wave potential can be written as:

$$\Phi_w(x,z,t) = P(z) * \sin(kx - \omega t) \tag{108}$$

The velocity potential of the harmonic waves has to fulfill 4 very important requirements:

1. Continuity condition or Laplace equation
2. Sea bed boundary condition
3. Free surface dynamic boundary condition
4. Free surface kinematic boundary condition.

The first 3 requirements lead to more complete expression of velocity potential. Which can be expressed as:

$$\Phi_w = \frac{\zeta_a g}{\omega} * \frac{\cosh \mathrm{k(h+z)}}{\cosh(kh)} * \sin(kx - wt) \tag{109}$$

To determine the relationship between the wave period T and the wave length L, the free surface kinematic boundary condition gives this relation. The so called dispersion relation describes the relation between T and L or equivalently k and $\boldsymbol{\omega}$.

$$\omega^2 = kg * \tanh(kh) \tag{110}$$

With the given dispersion relation the wave celerity c can be expressed as:

$$c = \sqrt{\frac{g}{k} * \tanh(kh)} \tag{111}$$

In deep water, the phase velocity is found by substituting $\tanh(kh) = 1$
In the equation the wave celerity can be expressed as:

$$c = \sqrt{\frac{g}{k}} = \frac{g}{\omega} \tag{112}$$

For shallow water the wave celerity can be expressed as:

$$c = \sqrt{gh} \quad \text{(shallow water)} \tag{113}$$

Water particle kinematics

The kinematics of a water particle can be found from the velocity potential and the dispersion relation. The velocities of the water particle in the x and z direction can be expressed as:

$$u = \zeta_a \omega * \frac{\cosh \mathrm{k}(h+z)}{\sinh(kh)} * \cos(kx - \omega t)\ \omega = \zeta_a \omega * \frac{\sinh \mathrm{k}(h+z)}{\sin(kh)} * \sin(kx - \omega t) \tag{114}$$

The water particle acceleration follows directly from the differentiating the velocity components of the water particle. The expression for the water particle acceleration are given as:

$$\dot{u} = \zeta_a \omega^2 * \frac{\sinh \mathrm{k}(h+z)}{\sinh(kh)} * \cos(kx - \omega t)\ \dot{\omega} = -\zeta_a \omega^2 * \frac{\sinh \mathrm{k}(h+z)}{\sinh(kh)} * \cos(kx - \omega t) \tag{115}$$

Pressure
The pressure in the linear wave theory follows from the linearized Bernoulli's equation. It means that the nonlinear terms in the equation are neglected.

$$\frac{\partial \Phi_w}{\partial t}+\frac{P}{\rho}+gz=0 \qquad or \qquad P=-\rho gz-\rho\frac{\partial \Phi_w}{\partial t} \tag{116}$$

Substituting the expression for the wave potential, the linear pressure can be expressed as:

$$P=\ -\rho gz+\rho g\zeta_a*\frac{\cosh \mathrm{k}(h+z)}{\cosh(kh)}*\cos(kx-\omega t) \tag{117}$$

Irregular Waves:
In the reality the wave profile at the sea doesn't look like nice sinusoidal wave. The realistic image of the wave profile in the reality looks very unregularly. Beside that the wave profile image changes with the time without repeating itself. It is possible to represent the irregular sea surface using a linear superposition of wave components of the regular waves.

When a time history of the wave profile is available, then a simple analysis can be carried out to obtain statistical data from the given record. The time history has to have a minimum length of 100 times the longest wave period.

Statistical information about the waves can be obtained from the probability density function f(x). The exceedance probability is given by the following formula:

$$P\{H_w>a\}=\int_a^\infty f(x)dx \tag{118}$$

Usually the wave high is given as the significant wave height. The significant wave high $H_{\frac{1}{3}}$ is the average of the highest 1/3 waves highs. This is an important parameter in practical applications of the wave statistics. There is a good correlation between the significant wave height and visually estimated wave heights.

The standard deviation σ of the water level ζ(t) is dependent on the number of the measurements N. It is also related to the significant wave amplitude and significant wave height. The relation between the different parameters are given the following formulations.

Standard deviation of the wave spectrum:

$$\sigma=\sqrt{\frac{1}{N-1}*\sum_{n=1}^{N}\zeta_n^2} \tag{119}$$

Significant wave amplitude:

$$\zeta_{a1/3}=2*\sigma \tag{120}$$

Significant wave amplitude:

$$H_{1/3}=4*\sigma \tag{121}$$

In statistical terms the water level elevation can be expressed by the so called Gaussian/Normal distribution. The probability density function of the normal distributed value is given by:

$$f(x)=\frac{1}{\sigma*\sqrt{2\pi}}*exp\left\{(\frac{x}{\sigma*\sqrt{2}})^2\right\} \tag{122}$$

Integrating the probability density function gives the occurrence chance of a value that will be exceeded.

$$P\{\zeta > a\} \quad = \int_a^{\infty} f(x)dx \tag{123}$$

$$= \frac{1}{\sigma * \sqrt{2\pi}} * \int_a^{\infty} exp\left\{(\frac{x}{\sigma * \sqrt{2}})^2\right\}dx$$

The wave amplitude is Rayleigh distributed. However it must satisfy the condition that the frequency range must be not too large. The spectrum must be narrow banded. The waves in the project area fulfill this condition. And that's why for the amplitude distribution is a relight distribution is chosen. The Rayleigh distribution is expressed by the equation

$$f(x) = \frac{x}{\sigma^2} * exp\left\{(\frac{x}{\sigma * \sqrt{2}})^2\right\} \tag{124}$$

To calculate the exceedanse probability of a given wave height the formula given below can be used. Here is H_w an individual wave height given in a wave field characterized by the significant wave height.

$$P\{H_w > H\} = exp\left\{-2(\frac{H}{H_{\frac{1}{3}}})^2\right\} \tag{125}$$

To describe the maximum wave height in a storm an engineering judgment is applied. For the practice it is enough to describe the maximum wave height in a storm as a wave height that will be exceeded once in 1000 storm waves. This is an arbitrary value, but in the practice it work well. And the reason is also that it takes at least 3 hours for passing of 1000 waves in a storm. Due to this a the worst peak of the storm will be passed. The maximum expected wave height in a 3 hour storm is: $H_{max} = 1.86H_{1/3}$

Wave Spectra

Wave period plays also a crucial role in loading of the structure. The wave period isthe reciprocal of the wave frequency. Irregular waves can be studied as the superposition of the regular waves. That's why the frequency characteristics are studied of an irregular wave record by using the Fourier series analysis. For this kind of analysis assumption is made that the signal being studied repeats itself after a sufficient long time. Off course this is not the case in the practice.

For the analysis the wave record of a long crested irregular sea in time domain is translated in the frequency domain. This is done to write the irregular wave signal as a large number of wave components in the frequency domain.

$$\zeta(t) = \sum_{n=1}^{N} \zeta_{an} * \cos(k_n x - \omega_n t + \varepsilon_n) \tag{126}$$

The Fourier series will contain a set of values $\zeta_{an}$and ε_n wich are assosiated with the ω_n. As mentioned befor in this cahapter the ω_n , k_n are related by the dispertion relation. If enough Fourier series terms are included, the entire time record can be reproduced in the frequency domail. Howerver the exact one is not interested in the exact wave record at some time moment. The statistical properties of the signal are of importance in the terms of wave frequency and amplitude. In this operation the information about the phase angel is lost in the analysis.

The amplitudes ζ_{an} can be obtained by a Fourier analysis of the wave record. The wave amplitude ζ_{an} of each frequency ω_n can be expressed in a wave spectrum.
The value $\frac{1}{2}\zeta_{an}^2(\omega)/\Delta\omega$ is plotted on the vertical access of the wave spectrum. If the wave spectrum is multiplied by ρg the wave energy spectrum will be obtained. It's easy to describe the wave spectrum in a formula, but the phase angle information is lost. As described earlier this is not of importance if one only needs

the statistical information. The wave spectrum can be expressed as given in the formula below.

$$S(\omega_n) * \ \Delta\omega = \sum_{\omega_n}^{\omega_n+\Delta\omega} \frac{1}{2}\zeta_{an}^2(\omega) \tag{127}$$

If $\Delta\omega$→ 0 the definition of the spectrum will be:

$$S(\omega) * d\omega = \frac{1}{2}\zeta_{an}^2 \tag{128}$$

The variance of the surface elevation is equal to the surface area under the spectrum and can be expressed as :

$$\sigma_\zeta^2 = \int_0^\infty S_\zeta(\omega) * d\omega \tag{129}$$

Several relationship can be found from the moments of the spectrum. The moments of the spectrum can be computed by multiplying the area under the spectrum with respect to the vertical axis at $\omega = 0$. The n^{th}order moment is given by:

$$m_n = \int_0^\infty \omega^n * S(\omega) * dw \tag{130}$$

The significant wave height and significant wave amplitude can be expressed in the terms of the zero order moment of the wave spectrum.

Significant wave amplitude:

$$\zeta_{a1/3} = 2 * \sqrt{m_{0\zeta}} \tag{131}$$

Significant wave height:

$$H_{1/3} = 4 * \sqrt{m_{0\zeta}} \tag{132}$$

Characteristic wave period can also defined from the moments of the wave spectrum. The T_2 or also called T_z and T_1 are defined as:

$$T_1 = 2\pi * \frac{m_{0\zeta}}{m_{1\zeta}} \tag{133}$$

$$T_2 = 2\pi * \sqrt{\frac{m_{0\zeta}}{m_{21\zeta}}} \tag{134}$$

<u>Standard wave spectra</u>

In order to be able to describe the wave statistics, standard spectra are avilable for use. The mathematical description of the standard uni directional spectra is given. If the input parameters for each type of spectrum are available then the wave spectrum can be reproduced for several aims of calculations. Tow very important spectra wich has to be considered are:

- Pierson-Moskowitz Wave Spectrum.
- JONSWAP Wave Spectra

The above mentioned spectra are limited apllicable for different kind of seas. The Pierson-Moskowitz Wave Spectrum is more appliable for the fully developed seas. The JONSWAP spectrum is appropriate for the fetch-limited (or costal) wind ginerated seas. The project are of the Fehmarn Belt is a typical example of a fetch limited sea. So that's why for the futher calculation the JONSWAP spectrum will be used. The mathematical expresion for the JONSWAP spectrum is:

$$S_{\zeta}(\omega) = \frac{320 * H_{1/3}^{2}}{T_p^4} * \omega^{-5} * exp\left\{\frac{-1950}{T_p^4} * \omega^{-4}\right\} * \gamma^{A} \tag{135}$$

Peakednees factor: $\gamma = 3.3$

$$A = exp\left\{-\left(\frac{\frac{\omega}{\omega_p} - 1}{\sigma * \sqrt{2}}\right)^2\right\} \tag{136}$$

Circular frequency at spectral peak:

$$\omega_p = \frac{2\pi}{T_p} \tag{137}$$

$$\sigma = a\ step\ function\ of\ \omega: \ = \left\{\begin{cases} if\ \omega < \omega_p & \sigma = 0.07 \\ if\ \omega > \omega_p & \sigma = 0.09 \end{cases}\right\} \tag{138}$$

$$T_1 = 1.073 * T_2 = 0.843 * T_p \tag{139}$$

$$0.932 * T_1 = T_2 = 0.777 * T_p \qquad \text{or} \quad T_p = 1.287 * T_2 = 1.199 * T_1 \tag{140}$$

Directional Spectra:
If the dominant wave direction has to be taken into account, then use of the cosine-squared rule can be made. The cosine rule is mostly introduces in th ecalculation if the directional spreading of the wave energy has to be taken into account. When the cosine rule is applied the the uni directional wave energy is scaled. The directional information is used in the analysis as a scalar. That's why the form of the spectrum in each direction is the same. The intensity of the spectrum will change as a function of the direction. The directional wave spectrum is:

$$S_{\zeta}(\omega, \mu) = \left\{\frac{2}{\pi} * \cos^2(\mu - \bar{\mu})\right\} * S_{\zeta}(\omega) \tag{141}$$

$$-\frac{\pi}{2} \leq (\mu - \bar{\mu}) \leq +\frac{\pi}{2} \tag{142}$$

When the nonlinear terms of the wave loads has to be included in the calculations the analysis has to be applied in the time domain. In this situation the wave spectrum in the frequency domain has to be translated in the time domain. This is reversal case if he case explaned earlier for the frequency domain. However the exact wave record can't be reproduced, because the the phase angle information is lost when translating the wave record to the time domain. But this will not lead to probplems in the analysis becau for the calculations is a wave record neded wich is statisticaly indistinguisable from the original one.

The wave amplitude can be obtained from:

$$\xi_{an} = 2\sqrt{S_{\xi}(\omega) * \Delta\omega} \tag{143}$$

The coresponding wave number can cumputed from the dispersion relationship. The wave number and frequency are related by the dispersion relationship. In order to obtain irrigular wave history, the phase angle ε_n has to included again. The new ε_n can be a random number in the range $0 \leq \varepsilon_n \leq 2\pi$. The ε_n is neded to generate the time record.

5.3 DYNAMIC BEHAVIOUR OF THE SYSTEM

The dynamioc behavior of the system is governed by different factors. The most important factors are the combination of the different external forces, moments and the inertial properties of the system. The forces and moments should be considered acting on the body as distributed forces. In general three different force mechanism are important when one is considering the dynamical behavior of the system in fluid: inertia, gravity and viscous.

To include all force mechanisms in a mathematical model is difficult. Chooses has to be made which force mechanisms are important for the for the case in the consideration. To determine the dominant force mechanisms it is useful to estimate the magnitude of the 3 mentioned force mechanisms. Mostly the viscous effects can be neglected, also for the present study the viscous effects are ignored in the first place. In the following the theory of the dynamical behavior is described for a floating system.

The translational and rotational behavior of the system can be described in 6 degrees of freedom, namely: surge, sway, heave, roll, pitch and yaw. All the mentioned translations and rotations are related to the center of gravity G of the floating system. Mostly the motion components have small amplitudes. Also for the considered case the immersion system for the Fehmarn Belt Tunnel is this true. The most used right handed orthogonal coordinate system to describe the motions of the floating body are:

- An earth-bound coordinate system $S(x_0, y_0, z_0)$
- A body–bound coordinate system $G(x_b, y_b, z_b)$
- A steadily translating coordinate system $O(x, y, z)$

For the modeling the earth-bound coordinate system $S(x_0, y_0, z_0)$ will be used. And the harmonic wave elevation in the earth bounded system is defined as:

$$\xi = \xi_a * \cos(\omega t - kx_0) \qquad (144)$$

The wave speed is defined as

$$c = \frac{\omega}{k} = \frac{\lambda}{T} \qquad (145)$$

The six motion of the center of gravity(COG) of the system can be expressed by the following equations:

1.*Surge* $x(t) = x_a \cos(\omega t + \varepsilon_{x\xi})$

2.*Sway* $y(t) = y_a \cos(\omega t + \varepsilon_{y\xi})$

3.*Heave* $z(t) = z_a \cos(\omega t + \varepsilon_{z\xi})$

4.*Roll* $f(t) = f_a \cos(\omega t + \varepsilon_{f\xi})$

5.*Pitch* $\theta(t) = \theta_a \cos(\omega t + \varepsilon_{\theta\xi})$

6.*Yaw* $\psi(t) = \psi_a \cos(\omega t + \varepsilon_{\psi\xi})$

The phase angles ε are related to undisturbed wave at origin of steadily translating system at CoG. The origin is located at the average position of the CoG of the floating system.An positive phase angle indicates that the motion is ahead then the wave elevation. This makes only sense when the steady state solution is studied. The wave elevation of at the CoG can be expressed as: $\zeta = \zeta_a * cos(\omega t)$

Motions at other points of the system

When the motion at the CoG are known, the motions at other points can be calculated by the means of the superposition. For the calculations the angels of rotations (roll, pitch and yaw) are assumed small (in order of 0.1 rad). The assumption for the small angels is necessary, because the linearization. The angels must be expressed in radians for the calculation. The linearized local motions of the system in steadily translating axes system can be calculated from:

$$\begin{pmatrix} x_p(t) \\ y_p(t) \\ z_p(t) \end{pmatrix} = \begin{pmatrix} x(t) \\ y(t) \\ z(t) \end{pmatrix} + \begin{pmatrix} 0 & \psi(t) & \theta(t) \\ \psi(t) & 0 & -f(t) \\ -\theta(t) & f(t) & 0 \end{pmatrix} * \begin{pmatrix} x_{bp} \\ y_{bp} \\ z_{bp} \end{pmatrix}$$

Using the given transformation matrix, the absolute harmonic motions of a certain point at the system can be expressed as:

$$\begin{aligned} x_p(t) &= x(t) - y_{bp} * \psi(t) + z_{bp} * \theta(t) \\ y_p(t) &= y(t) + x_{bp} * \psi(t) - z_{bp} * f(t) \\ z_p(t) &= z(t) - x_{bp} * \theta(t) + y_{bp} * f(t) \end{aligned}$$

Modeling Motions of the floating Structure

It's obvious that due to the waves the floating system (tunnel element + 2 Pontoon barges) will oscillate. It can be stated that the wave forces affect the motions of the floating system. In its turn the motion of the system has also an effect on the wave force. An irregular sea motion causes an irregular response of the floating structure/system. Mostly the first harmonic motions of the structure are of interest. For the first order motions of the floating system the superposition principle can be used. This means that components at range of frequencies can be added. This gives a realistic results for the systems motions. This approach is called frequency domain calculation.

If the motion of the floating structure are considered to be linear, it means that at each frequency the ratio's between the motions amplitude and the wave amplitude and the corresponding phase shifts are constant. It means that when one of the variables changes then the other variables also changes in a constant ratio, but the phase angle remains constant. The advantage of the linear theory is that the motions in irregular seas can be obtained from superposing the result of the responses to the regular waves with different amplitudes and directions. When the wave spectrum is known the response spectrum can be obtained in combination with the frequency characteristics.

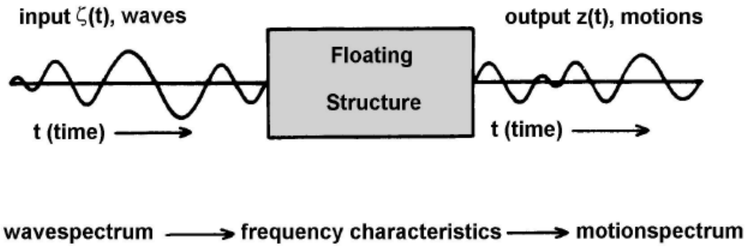


Figure 33 Motion principle of a floating structure

For predicting the floating bodies motion the simple model Mass Spring system can be used. The dynamic behavior of the system can be described by Newton's Second Law. The translation of the floating body can be calculated from the force balance of the system and the rotations are obtained from the moment balance. In this model the assumption is made that the mass of the floating body doesn't change during the time interval which should be longer then the oscillation period. In formula form the equations for 3 translations and 3 rotations of the centre of gravity are given by:

$$\vec{F} = \frac{d}{dt}(m\vec{U}) \text{ and } \vec{M} = \frac{d}{dt}(\vec{H}) \tag{146}$$

The system is assumed to be linear, the response of the system due to different wave loads can be superposed to obtain the total response of the system. That's why the wave loads can be split in different parts.

The response to each wave load type will be calculated separately and then each part will be added to get the total response. In order to model the wave loads, two important assumptions are made.
The hydro dynamical loads and moments are calculated by assuming that the body is oscillating in still standing water. The second assumption is that the waves are acting on a fixed body. The figure below shows the principle.

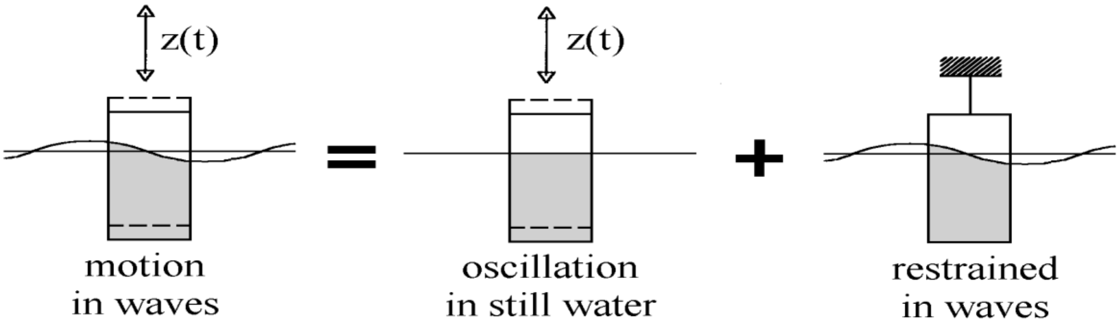


Figure 34 Summation of the two motional components

Oscillation in still water
The force due to the motions in still water can be split in three parts .

- Force in phase with the acceleration of the floating body (added mass force)
- Force in phase with the velocity of the floating body (added damping)
- Force in phase with displacement of the floating body (restoring force, spring force)

The first two elements of the force are called the hydro dynamical part of the force. The third part is the hydrostatic force.

The free oscillation of the body in initially standing still water will cause extra forces on the floating body. These forces has terms which are proportional to the mass terms and damping terms. The term which are proportional to the velocity terms are called added damping and the terms which are proportional to the acceleration terms are called added mass. The terms of added damping and added mass are indicated with the letters **a** and **b** in the equation of motion. The coefficient a has a dimension of mass and the coefficient b has a dimension of mass per unit of time. In general these coefficients are not constant and depend on the frequency of the motion.

The free motions of the floating body will generate weaves, which radically propagate from it. The generated waves will transport energy which is withdraw from the oscillations of the body. The withdrawal of the energy will cause that the oscillation of the body will decrease in time and finally it will die out. The wave damping is linear proportional to the velocity of the oscillation in the linear systems. The actual damping will be greater than the linear damping because of the viscosity of the fluid. For the linear systems the effects of the viscosity can be neglected because this effect is small. The added damping or the so called radiation damping decreases from the water surface. Because the wave generation will be less if a body lying deep under the surface.

The hydro dynamical reaction force which is proportional to the acceleration term as mentioned before is called the added mass. This force is the result of the accelerations
which are given to the water particle near the floating structure. The difference with the previous hydro dynamical load type is that added mass energy doesn´t dissipate energy it behaves like a standing wave.

When the amplitudes of the waves are small the accelerations and velocity behave quite linear. It can be stated that the terms are linear for the wave heights in the project area. The hydro dynamical forces are the total reaction forces which are performed in the still standing water can be expressed as:

$$m\ddot{x} = F_h \qquad with \qquad F_h = -a\ddot{x} - b\dot{x} - cz \tag{147}$$

In the equation of motion for the translational motion the force balances has to be implemented. For the rotations the moment balances has to be taken into the considerations.

The radiated force caused by the motion of the floating bodies in still standing water can be calculated by integrating the pressure over the body surface.

$$F_k = -\iint\limits_{S_0} P n_k dS = -\rho\omega^2 \overline{\eta_J} \iint\limits_{S_0} \left(\phi_j\right) n_k dS \tag{148}$$

<u>Restoring spring terms of a floating body</u>

For free floating bodies only the restoring spring terms in the following degrees of freedom are present: heave, roll and pitch. The other degrees doesn't have a restoring moment. The restoring spring term for the heave follows from the water plane area. For the angular motions the terms follow from the static stability phenomena. The terms for the free floating structure can be expressed as:

Heave $\quad c_{zz} = \rho g A_{WL}$ (149)

Roll $\quad c_{\phi\phi} = \rho g \nabla * \overline{GM}$ (150)

Pitch $\quad c_{\theta\theta} = \rho g \nabla * \overline{GM\_L}$ (151)

In which $\overline{GM}$ *and* $\overline{GM_L}$ are the transverse and longitudinal metacentric heights of the floating structure.

<u>Wave Loads</u>

As described before to calculate the wave loads on a floating structure, it's been assumed that the structure is fixed in its location (cylinder tests). First the theory of the regular waves will be described, then the extension can be easily made by applying the Fouirrier Analysis to get the results for the irregular waves.

To determine the wave forces on a floating body in water which is interacting with waves the forces can be determined by integrating the pressure over the wetted body of the structure. The pressure P on the surface of a floating structure is given by the Bernoulli equation. Since we consider in the first place a linear system also a the linear Bernoulli equation will be considered:

$$P = -\rho \frac{d\Phi}{dt} \tag{152}$$

In this equation the hydrostatic terms has been neglected, because the hydrostatic pressure doesn't contribute to the oscillation of the floating structure. In the equation above ρ is the water density and Φ is the wave potential. The exciting wave force can be calculated by integrating the pressure over wetted surface S_o.

$$F_k = -\iint\limits_{S_0} P n_k dS = \iint\limits_{S_0} i\rho\omega(\phi_0 + \phi_7) n_k dS \, e^{i\omega t} \tag{153}$$

$$(f_{0k} + f_{7k}) * e^{i\omega t} = f_k * e^{i\omega t} \qquad with \qquad k = (1,2, \dots .6) \tag{154}$$

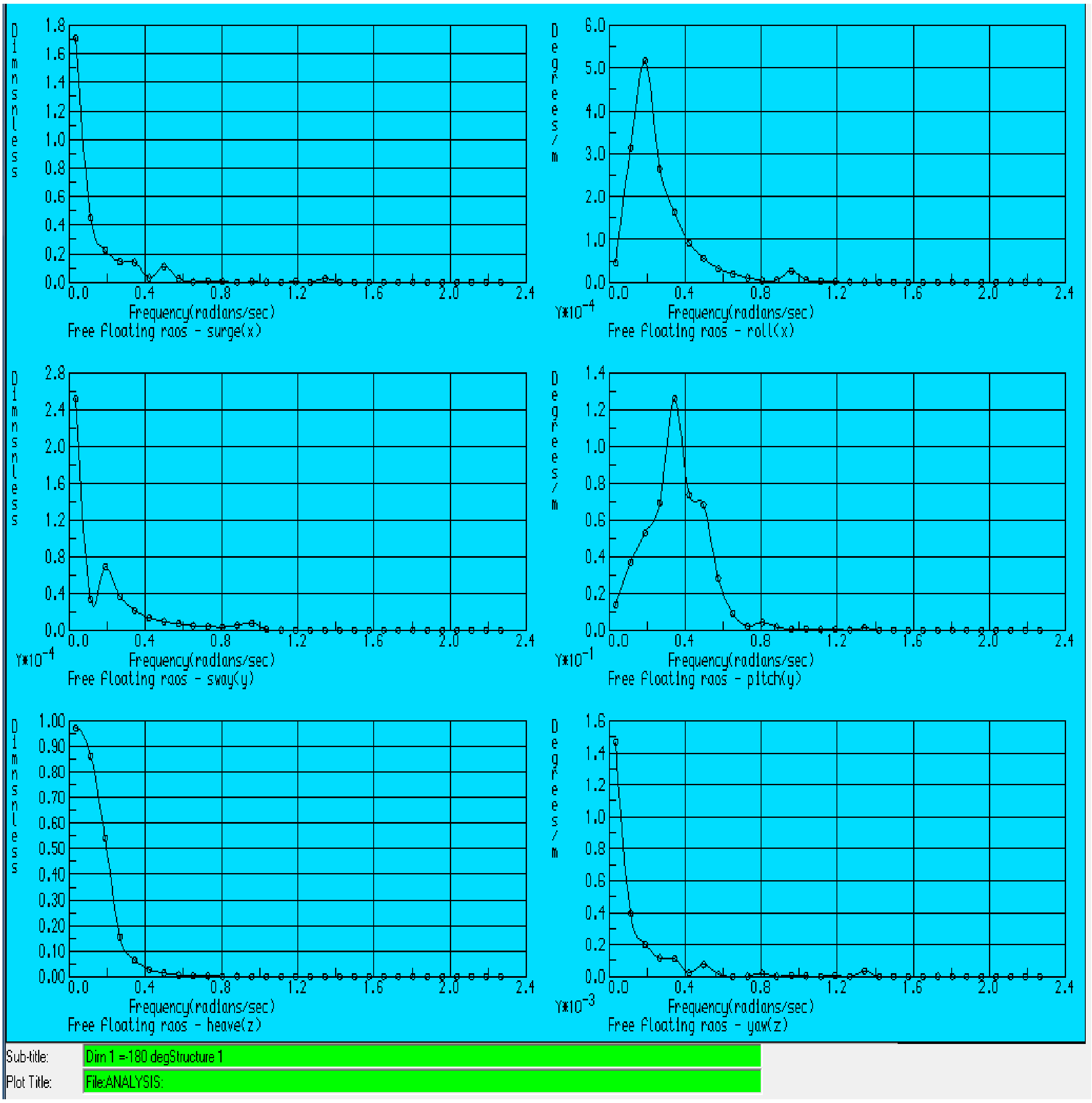
In this equation f_{0k} represent the undisturbed wave force or the so Called Froud Krilov Foce. The under script k indicates in which degree of freedom is the force considered. The f_{7k} is the diffracted force. This forces have harmonic character. The diffracted force is included in the equation because of the diffracting of the waves due to the presence of the structure. The total external wave force is on the floating structure is the sum of Froude-Krilov Force and diffracting force.

For the low frequencies (long waves) the diffracted part of the force is smaller compared to the undisturbed wave force part (Froude-Krilov Force). At the higher frequencies the diffracted part of the force getting more important.

6 ANNES (RAO'S FLOATING TUNNEL ELMENT)

6.1 INTRODUCTION

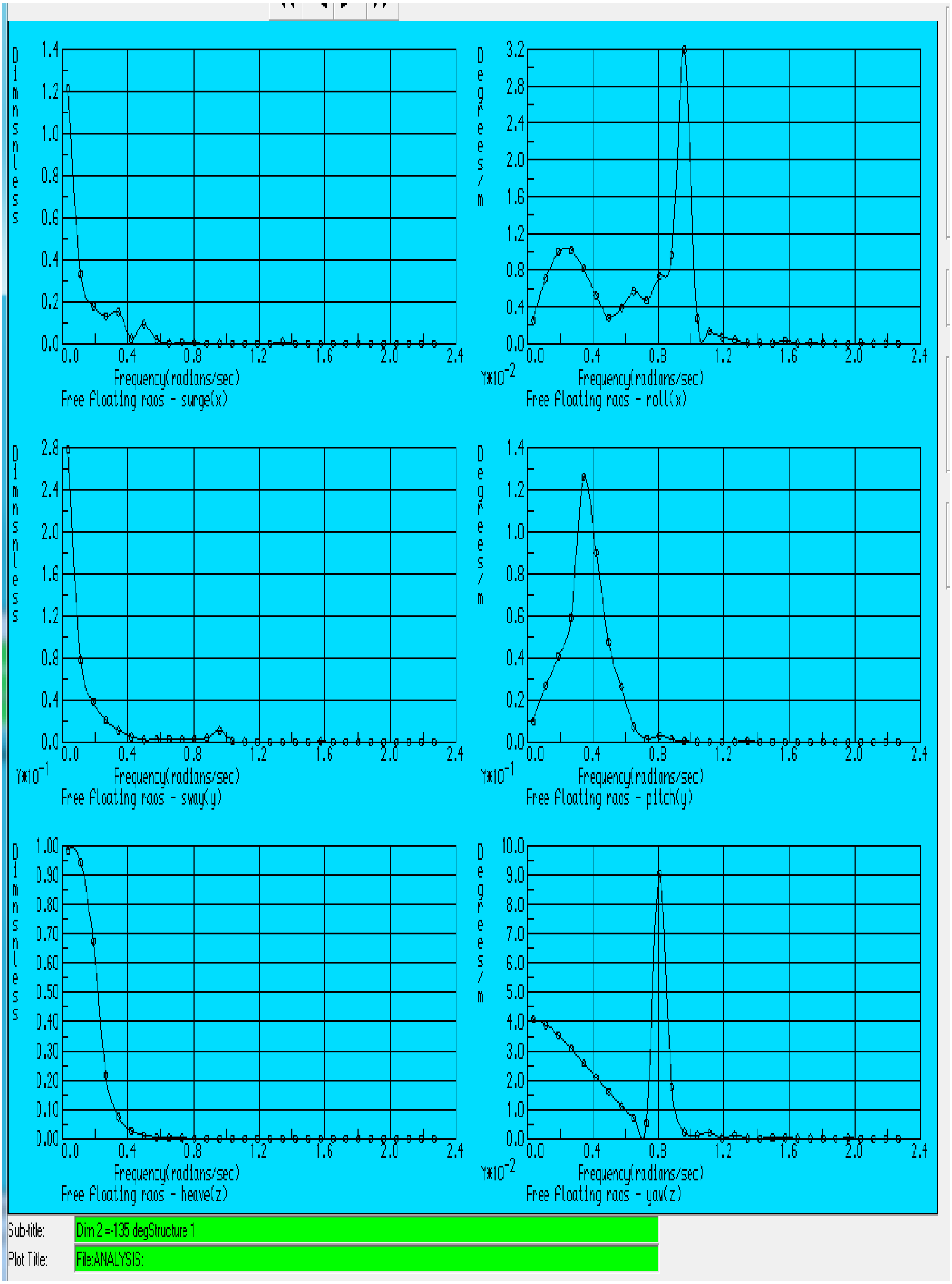
In this part of the report the RAO's for the floating tunnel element are given as function of wave frequency. The RAO's are calculated for a water depth of 30 [m] and a water density of 1031 [kg/m^3]. The calculations are performed with the aid of the 3D diffraction programme Ansys Aqwa. For the calculations a small Mesh was generated, namely : 2 by 2 in order to get enough accuracy in the results. The effect of the mooring lines is ignored in the calculations. In order to investigate the sensibility of the system to the wave direction the calculations are performed for the 360°. The assumed position of the element is during the immersion. The wave incident angle, indicate the angle to the longitudinal direction of the element.



Sub-title: Dim 1 =-180 degStructure 1

Plot Title: File:ANALYSIS:

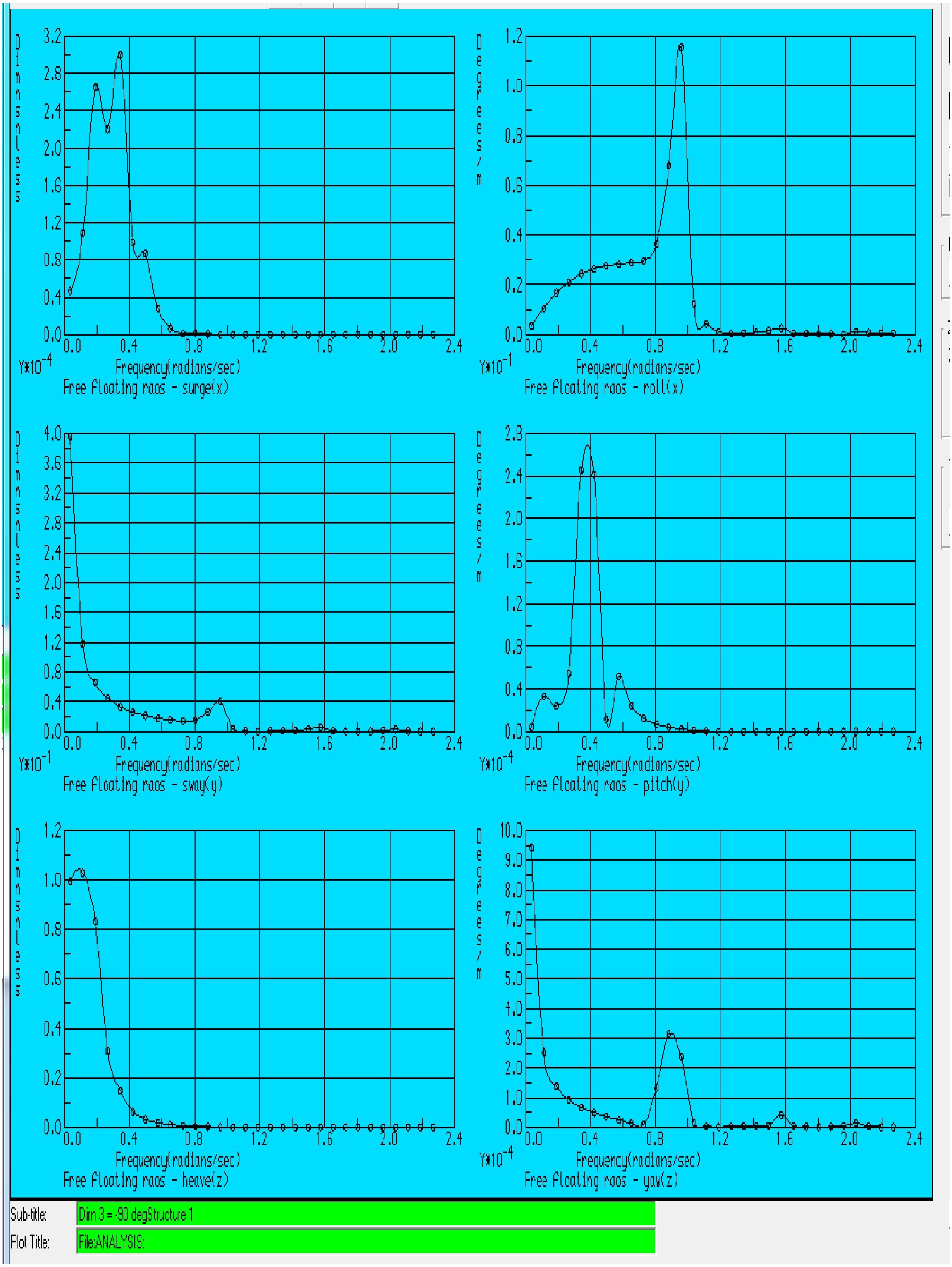
Figure 35 RAO's in 6 degrees of freedom for wave angel of (-180°)



Sub-title: Dim 2 =-135 degStructure 1

Plot Title: File:ANALYSIS:

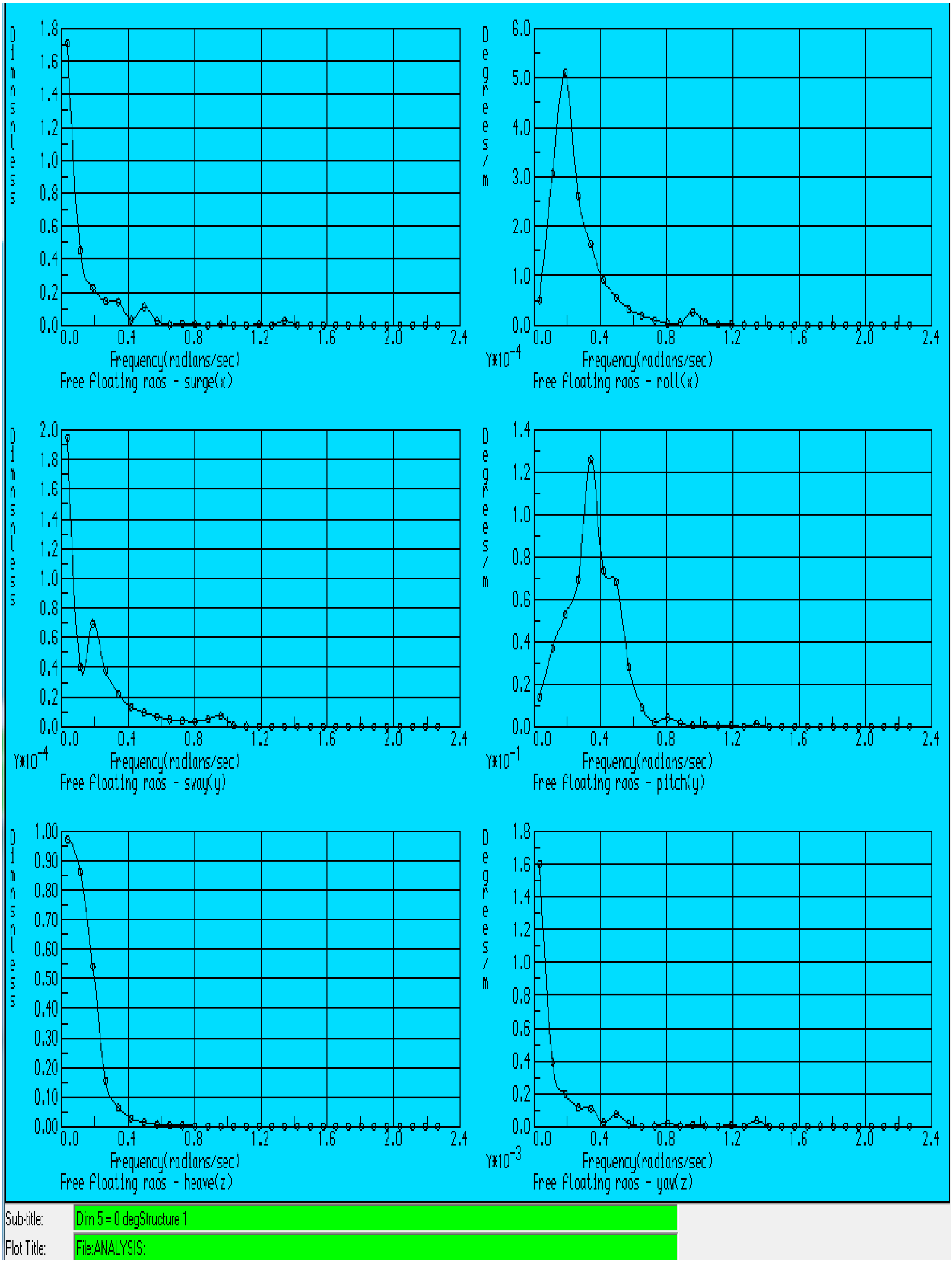
Figure 36 RAO’s in 6 degrees of freedom for wave angel of (-135°)



Sub-title: Dim 3 = -90 degStructure 1

Plot Title: File:ANALYSIS:

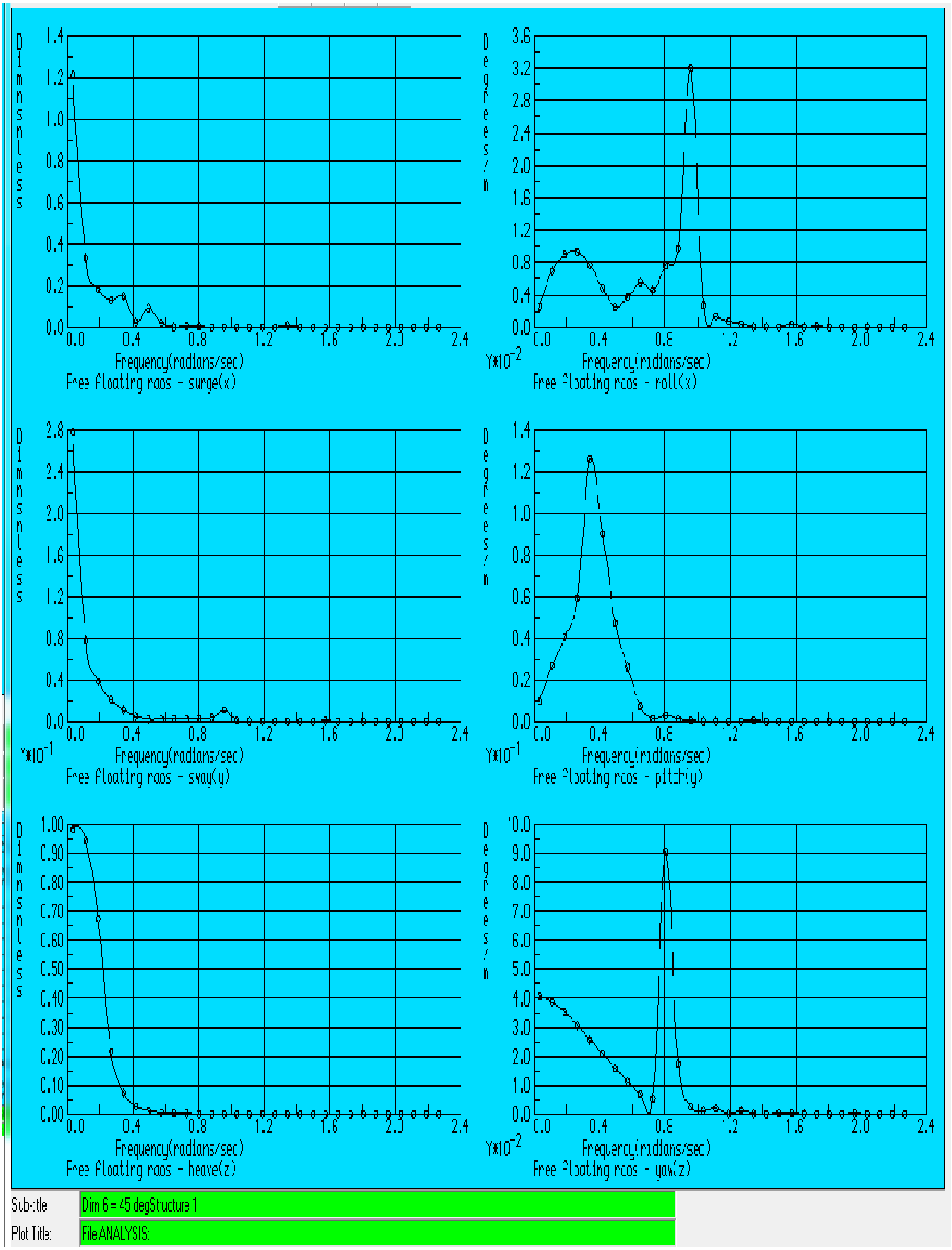
Figure 37 RAO's in 6 degrees of freedom for wave angel of (90°)



Sub-title: Dirn 5 = 0 degStructure 1

Plot Title: File:ANALYSIS:

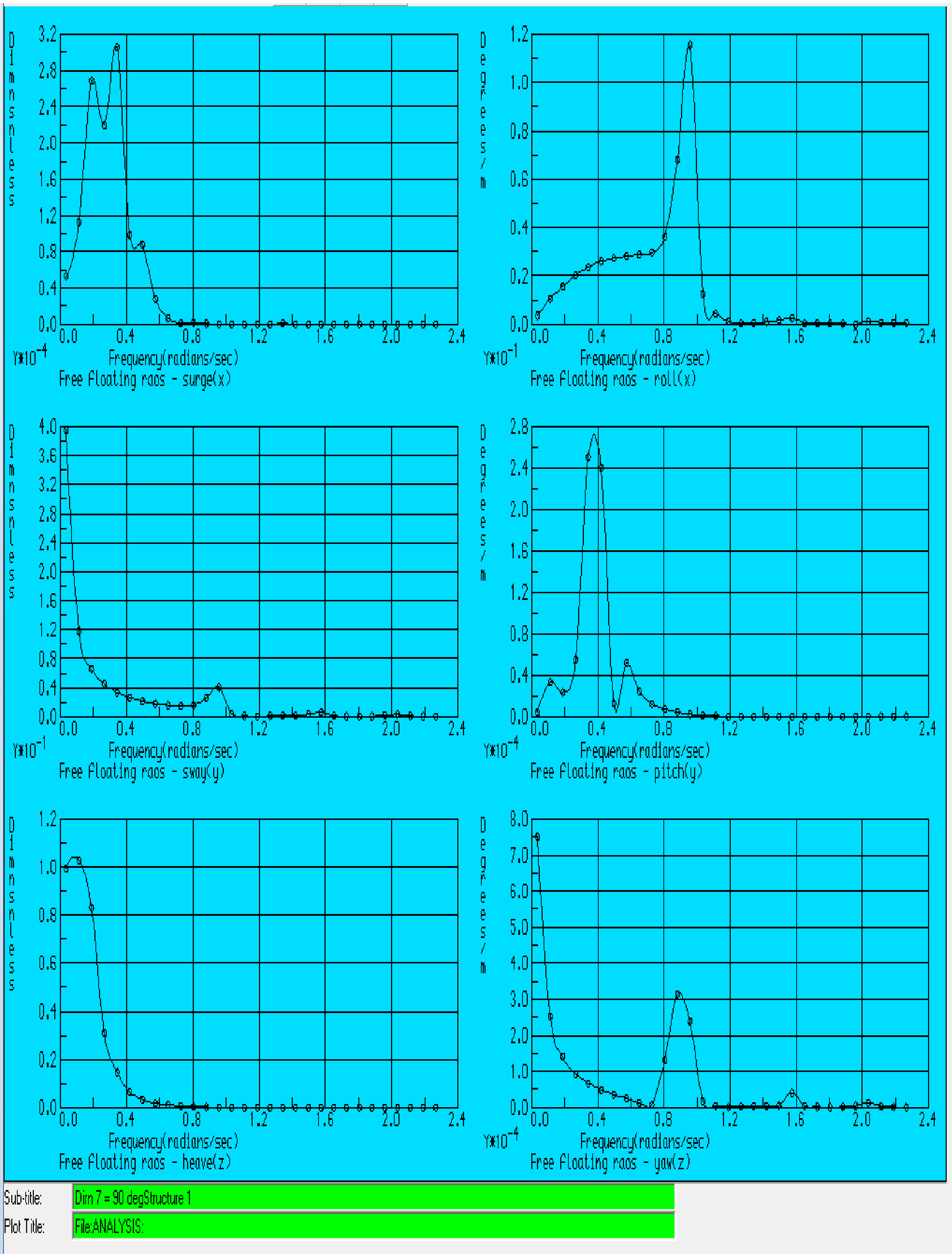
Figure 38 RAO’s in 6 degrees of freedom for wave angel of (0°)



Sub-title: Dim 6 = 45 degStructure 1

Plot Title: File:ANALYSIS:

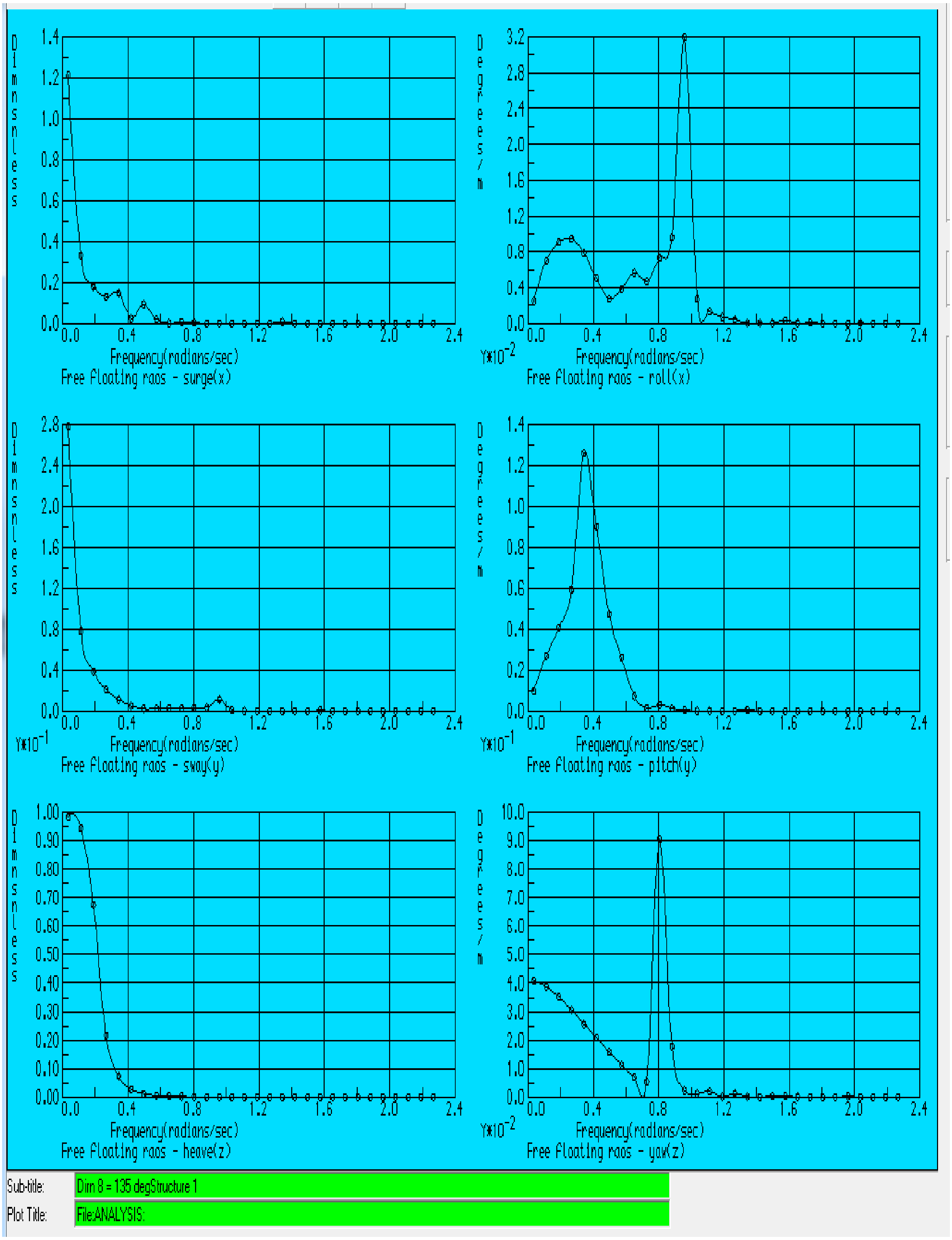
Figure 39 RAO's in 6 degrees of freedom for wave angel of (45°)



Sub-title: Dirn 7 = 90 degStructure 1

Plot Title: File:ANALYSIS:

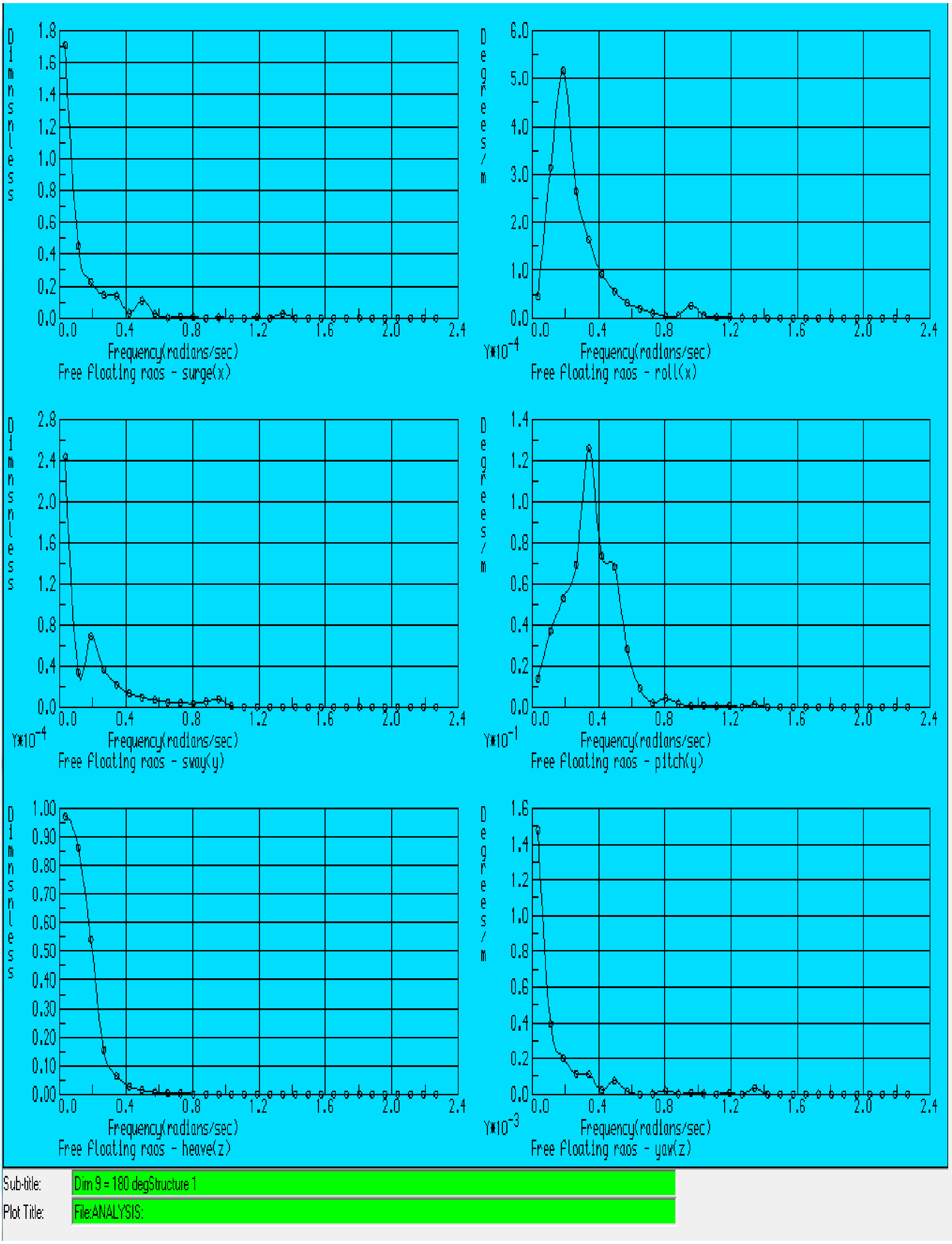
Figure 40 RAO's in 6 degrees of freedom for wave angel of (90°)



Sub-title: Dirn 8 = 135 degStructure 1

Plot Title: File:ANALYSIS:

Figure 41 RAO’s in 6 degrees of freedom for wave angel of (135°)



Sub-title: Dim 9 = 180 degStructure 1

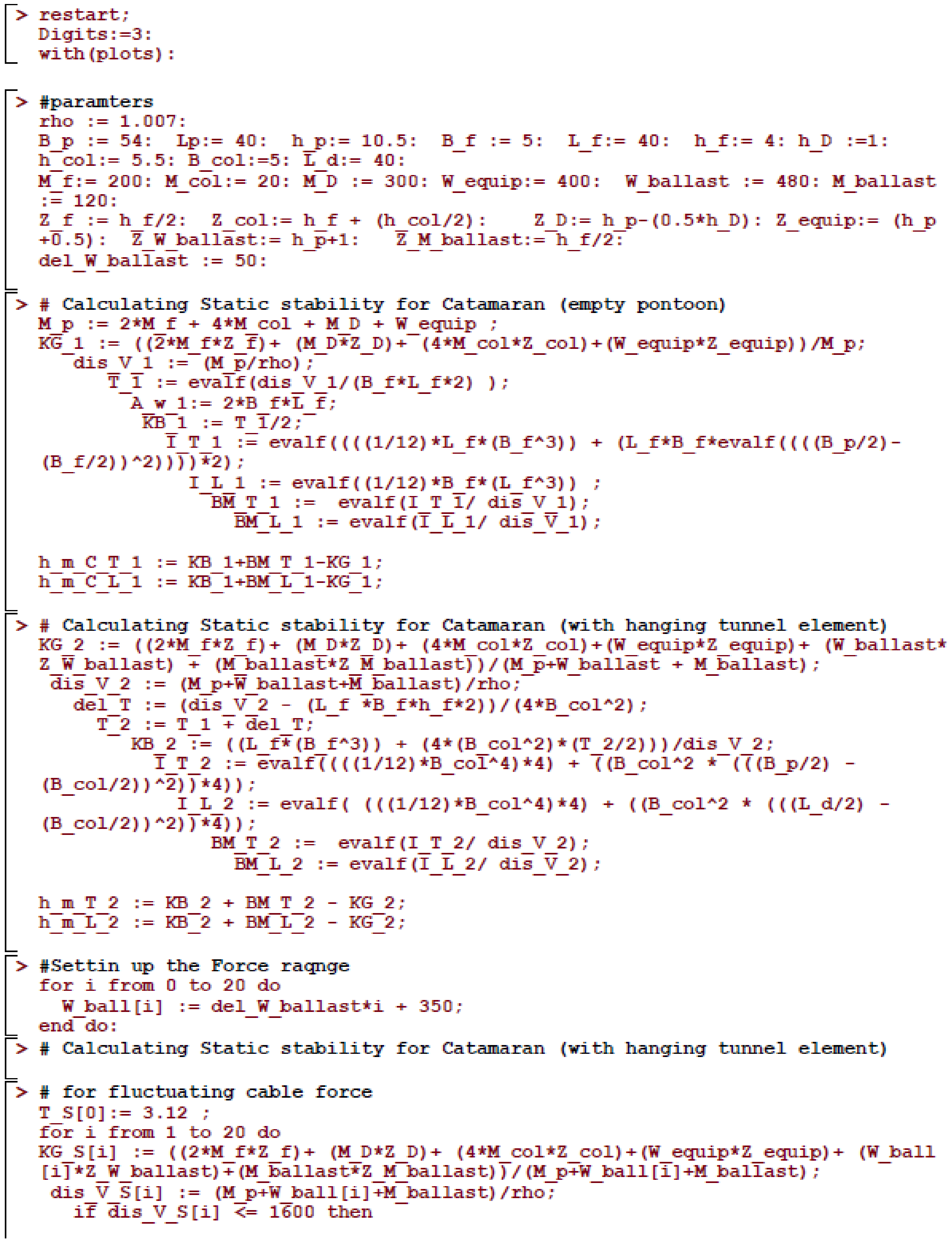
Plot Title: File:ANALYSIS:

Figure 42 RAO's in 6 degrees of freedom for wave angel of (180°)

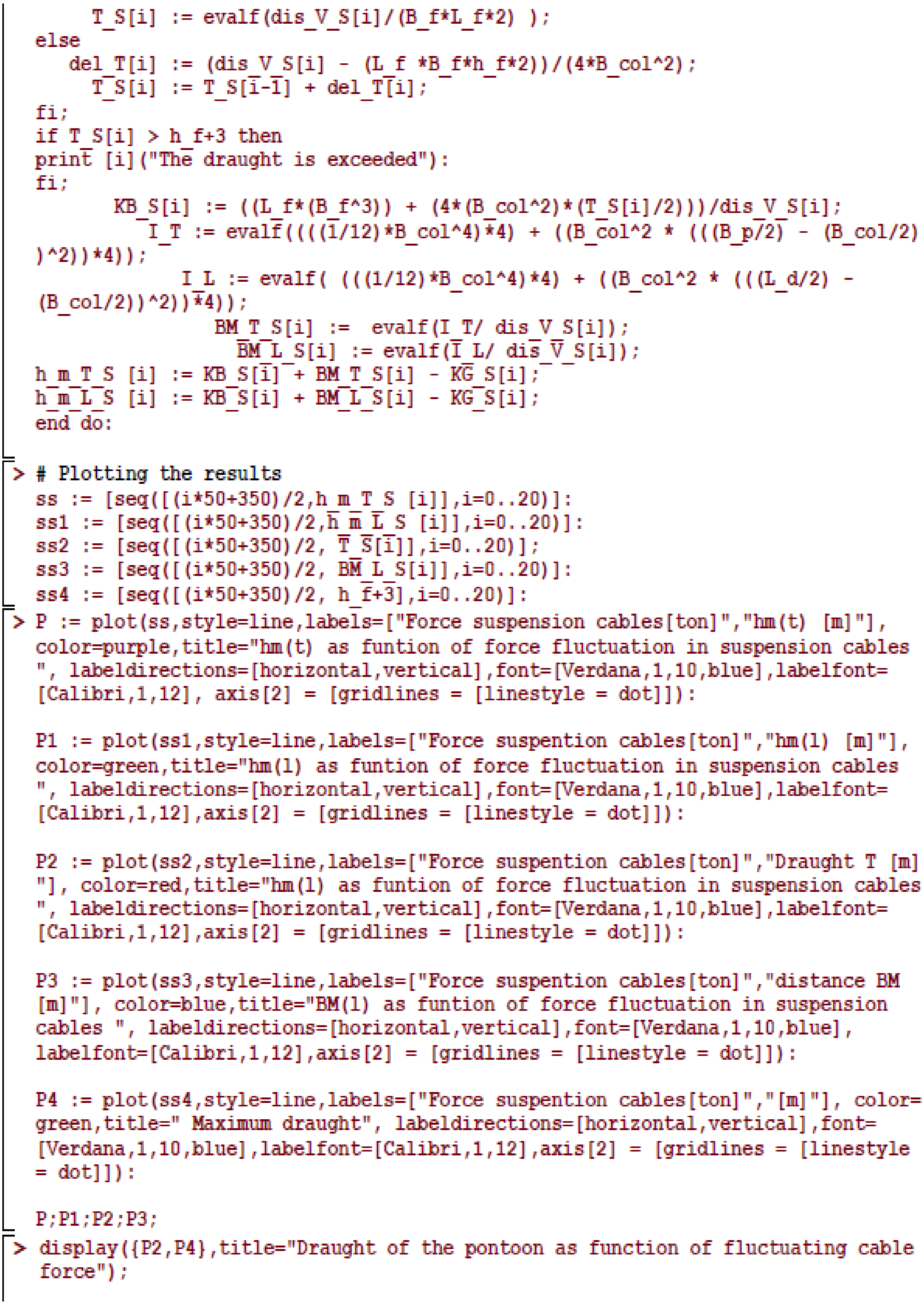
7 ANNEX (MAPLE FILES)

7.1 CALCULATIONS FILES STATIC STABILITY

Calculation files static stability Semi-Submersible Pontoon

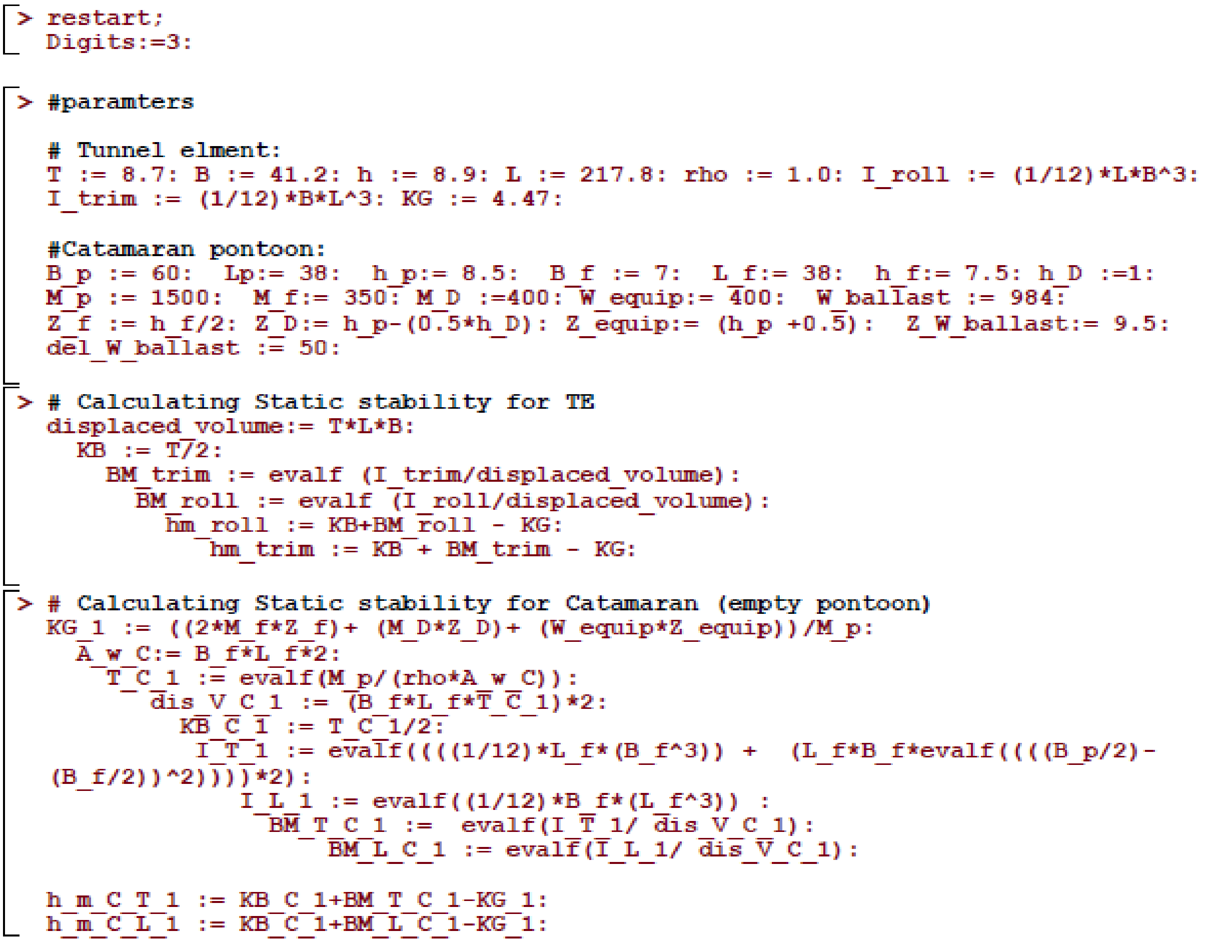


```
> restart;
  Digits:=3:
  with(plots):

> #paramters
  rho := 1.007:
  B_p := 54:  Lp:= 40:  h_p:= 10.5:  B_f := 5:  L_f:= 40:  h_f:= 4: h_D :=1:
  h_col:= 5.5: B_col:=5: L_d:= 40:
  M_f:= 200: M_col:= 20: M_D := 300: W_equip:= 400:  W_ballast := 480: M_ballast
  := 120:
  Z_f := h_f/2:  Z_col:= h_f + (h_col/2):    Z_D:= h_p-(0.5*h_D): Z_equip:= (h_p
  +0.5):  Z_W_ballast:= h_p+1:   Z_M_ballast:= h_f/2:
  del_W_ballast := 50:

> # Calculating Static stability for Catamaran (empty pontoon)
  M_p := 2*M_f + 4*M_col + M_D + W_equip ;
  KG_1 := ((2*M_f*Z_f)+ (M_D*Z_D)+ (4*M_col*Z_col)+(W_equip*Z_equip))/M_p;
     dis_V_1 := (M_p/rho);
        T_1 := evalf(dis_V_1/(B_f*L_f*2) );
          A_w_1:= 2*B_f*L_f;
           KB_1  := T_1/2;
             I_T_1 := evalf((((1/12)*L_f*(B_f^3)) + (L_f*B_f*evalf((((B_p/2)-
  (B_f/2))^2))))*2);
               I_L_1 := evalf((1/12)*B_f*(L_f^3)) ;
                 BM_T_1 :=  evalf(I_T_1/ dis_V_1);
                   BM_L_1 := evalf(I_L_1/ dis_V_1);

  h_m_C_T_1 := KB_1+BM_T_1-KG_1;
  h_m_C_L_1 := KB_1+BM_L_1-KG_1;

> # Calculating Static stability for Catamaran (with hanging tunnel element)
  KG_2 := ((2*M_f*Z_f)+ (M_D*Z_D)+ (4*M_col*Z_col)+(W_equip*Z_equip)+ (W_ballast*
  Z_W_ballast) + (M_ballast*Z_M_ballast))/(M_p+W_ballast + M_ballast);
   dis_V_2 := (M_p+W_ballast+M_ballast)/rho;
     del_T := (dis_V_2 - (L_f *B_f*h_f*2))/(4*B_col^2);
       T_2 := T_1 + del_T;
          KB_2 := ((L_f*(B_f^3)) + (4*(B_col^2)*(T_2/2)))/dis_V_2;
            I_T_2 := evalf((((1/12)*B_col^4)*4) + ((B_col^2 * (((B_p/2) -
  (B_col/2))^2))*4));
              I_L_2 := evalf( (((1/12)*B_col^4)*4) + ((B_col^2 * (((L_d/2) -
  (B_col/2))^2))*4));
                 BM_T_2 :=  evalf(I_T_2/ dis_V_2);
                   BM_L_2 := evalf(I_L_2/ dis_V_2);

  h_m_T_2 := KB_2 + BM_T_2 - KG_2;
  h_m_L_2 := KB_2 + BM_L_2 - KG_2;

> #Settin up the Force raqnge
  for i from 0 to 20 do
    W_ball[i] := del_W_ballast*i + 350;
  end do:
> # Calculating Static stability for Catamaran (with hanging tunnel element)

> # for fluctuating cable force
  T_S[0]:= 3.12 ;
  for i from 1 to 20 do
  KG_S[i] := ((2*M_f*Z_f)+ (M_D*Z_D)+ (4*M_col*Z_col)+(W_equip*Z_equip)+ (W_ball
  [i]*Z_W_ballast)+(M_ballast*Z_M_ballast))/(M_p+W_ball[i]+M_ballast);
   dis_V_S[i] := (M_p+W_ball[i]+M_ballast)/rho;
     if dis_V_S[i] <= 1600 then
```

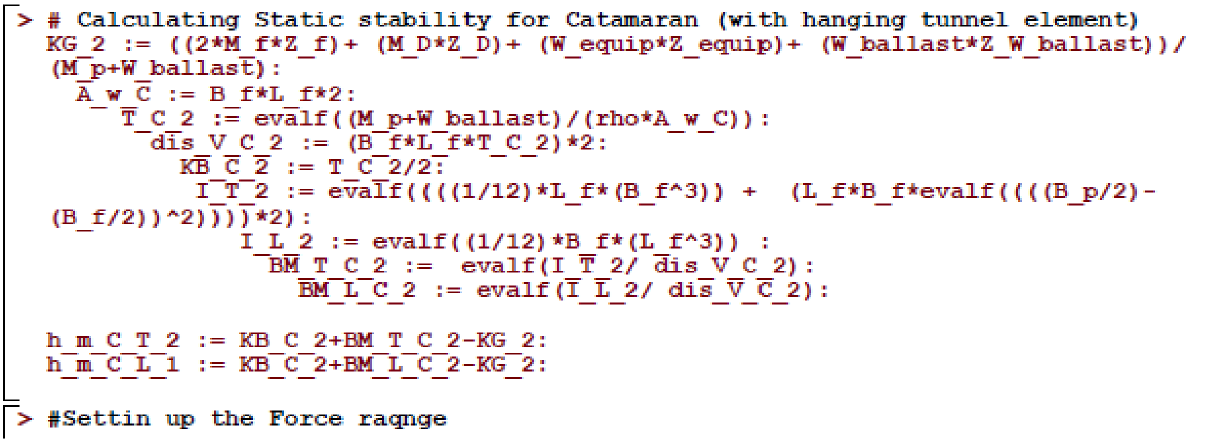


```
      T_S[i] := evalf(dis_V_S[i]/(B_f*L_f*2) );
  else
     del_T[i] := (dis_V_S[i] - (L_f *B_f*h_f*2))/(4*B_col^2);
      T_S[i] := T_S[i-1] + del_T[i];
  fi;
  if T_S[i] > h_f+3 then
  print [i]("The draught is exceeded"):
  fi;
       KB_S[i] := ((L_f*(B_f^3)) + (4*(B_col^2)*(T_S[i]/2)))/dis_V_S[i];
          I_T := evalf((((1/12)*B_col^4)*4) + ((B_col^2 * (((B_p/2) - (B_col/2)
  )^2))*4));
             I_L := evalf( (((1/12)*B_col^4)*4) + ((B_col^2 * (((L_d/2) -
  (B_col/2))^2))*4));
                BM_T_S[i] :=  evalf(I_T/ dis_V_S[i]);
                 BM_L_S[i] := evalf(I_L/ dis_V_S[i]);
  h_m_T_S [i] := KB_S[i] + BM_T_S[i] - KG_S[i];
  h_m_L_S [i] := KB_S[i] + BM_L_S[i] - KG_S[i];
  end do:

> # Plotting the results
  ss := [seq([(i*50+350)/2,h_m_T_S [i]],i=0..20)]:
  ss1 := [seq([(i*50+350)/2,h_m_L_S [i]],i=0..20)]:
  ss2 := [seq([(i*50+350)/2, T_S[i]],i=0..20)];
  ss3 := [seq([(i*50+350)/2, BM_L_S[i]],i=0..20)]:
  ss4 := [seq([(i*50+350)/2, h_f+3],i=0..20)]:
> P := plot(ss,style=line,labels=["Force suspension cables[ton]","hm(t) [m]"],
  color=purple,title="hm(t) as funtion of force fluctuation in suspension cables
  ", labeldirections=[horizontal,vertical],font=[Verdana,1,10,blue],labelfont=
  [Calibri,1,12], axis[2] = [gridlines = [linestyle = dot]]):

  P1 := plot(ss1,style=line,labels=["Force suspention cables[ton]","hm(l) [m]"],
  color=green,title="hm(l) as funtion of force fluctuation in suspension cables
  ", labeldirections=[horizontal,vertical],font=[Verdana,1,10,blue],labelfont=
  [Calibri,1,12],axis[2] = [gridlines = [linestyle = dot]]):

  P2 := plot(ss2,style=line,labels=["Force suspention cables[ton]","Draught T [m]
  "], color=red,title="hm(l) as funtion of force fluctuation in suspension cables
  ", labeldirections=[horizontal,vertical],font=[Verdana,1,10,blue],labelfont=
  [Calibri,1,12],axis[2] = [gridlines = [linestyle = dot]]):

  P3 := plot(ss3,style=line,labels=["Force suspention cables[ton]","distance BM
  [m]"], color=blue,title="BM(l) as funtion of force fluctuation in suspension
  cables ", labeldirections=[horizontal,vertical],font=[Verdana,1,10,blue],
  labelfont=[Calibri,1,12],axis[2] = [gridlines = [linestyle = dot]]):

  P4 := plot(ss4,style=line,labels=["Force suspention cables[ton]","[m]"], color=
  green,title=" Maximum draught", labeldirections=[horizontal,vertical],font=
  [Verdana,1,10,blue],labelfont=[Calibri,1,12],axis[2] = [gridlines = [linestyle
  = dot]]):

  P;P1;P2;P3;
> display({P2,P4},title="Draught of the pontoon as function of fluctuating cable
  force");
```

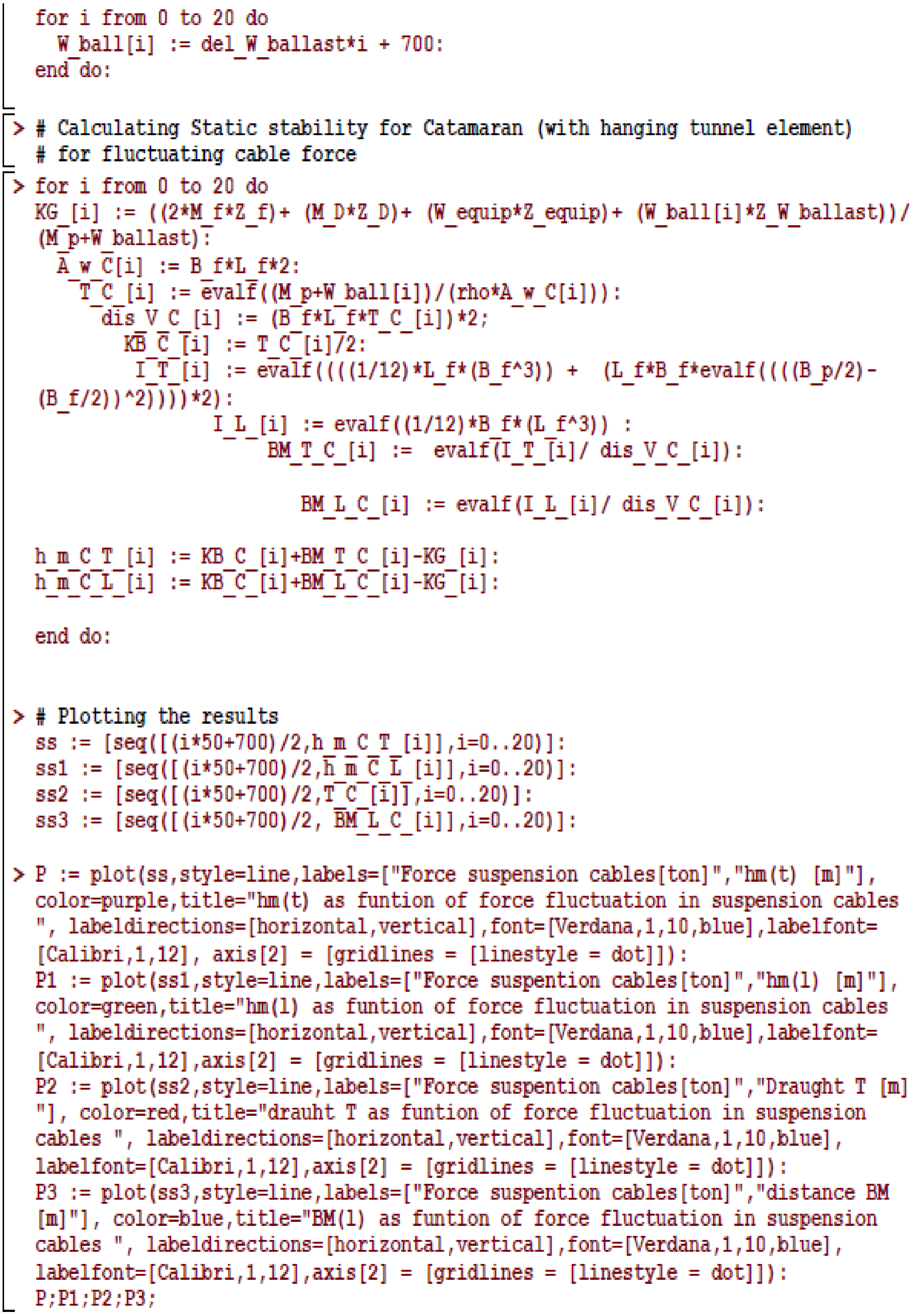
Calculation files static stability Catamaran Pontoon



```
> restart;
  Digits:=3:

> #paramters

  # Tunnel elment:
  T := 8.7: B := 41.2: h := 8.9: L := 217.8: rho := 1.0: I_roll := (1/12)*L*B^3:
  I_trim := (1/12)*B*L^3: KG := 4.47:

  #Catamaran pontoon:
  B_p := 60:  Lp:= 38:  h_p:= 8.5:  B_f := 7:  L_f:= 38:  h_f:= 7.5: h_D :=1:
  M_p := 1500:  M_f:= 350: M_D :=400: W_equip:= 400:  W_ballast := 984:
  Z_f := h_f/2: Z_D:= h_p-(0.5*h_D): Z_equip:= (h_p +0.5):  Z_W_ballast:= 9.5:
  del_W_ballast := 50:

> # Calculating Static stability for TE
  displaced_volume:= T*L*B:
    KB := T/2:
      BM_trim := evalf (I_trim/displaced_volume):
        BM_roll := evalf (I_roll/displaced_volume):
          hm_roll := KB+BM_roll - KG:
             hm_trim := KB + BM_trim - KG:

> # Calculating Static stability for Catamaran (empty pontoon)
  KG_1 := ((2*M_f*Z_f)+ (M_D*Z_D)+ (W_equip*Z_equip))/M_p:
    A_w_C:= B_f*L_f*2:
      T_C_1 := evalf(M_p/(rho*A_w_C)):
         dis_V_C_1 := (B_f*L_f*T_C_1)*2:
           KB_C_1 := T_C_1/2:
            I_T_1 := evalf((((1/12)*L_f*(B_f^3)) +  (L_f*B_f*evalf((((B_p/2)-
  (B_f/2))^2))))*2):
               I_L_1 := evalf((1/12)*B_f*(L_f^3)) :
                 BM_T_C_1 :=  evalf(I_T_1/ dis_V_C_1):
                     BM_L_C_1 := evalf(I_L_1/ dis_V_C_1):

  h_m_C_T_1 := KB_C_1+BM_T_C_1-KG_1:
  h_m_C_L_1 := KB_C_1+BM_L_C_1-KG_1:
```



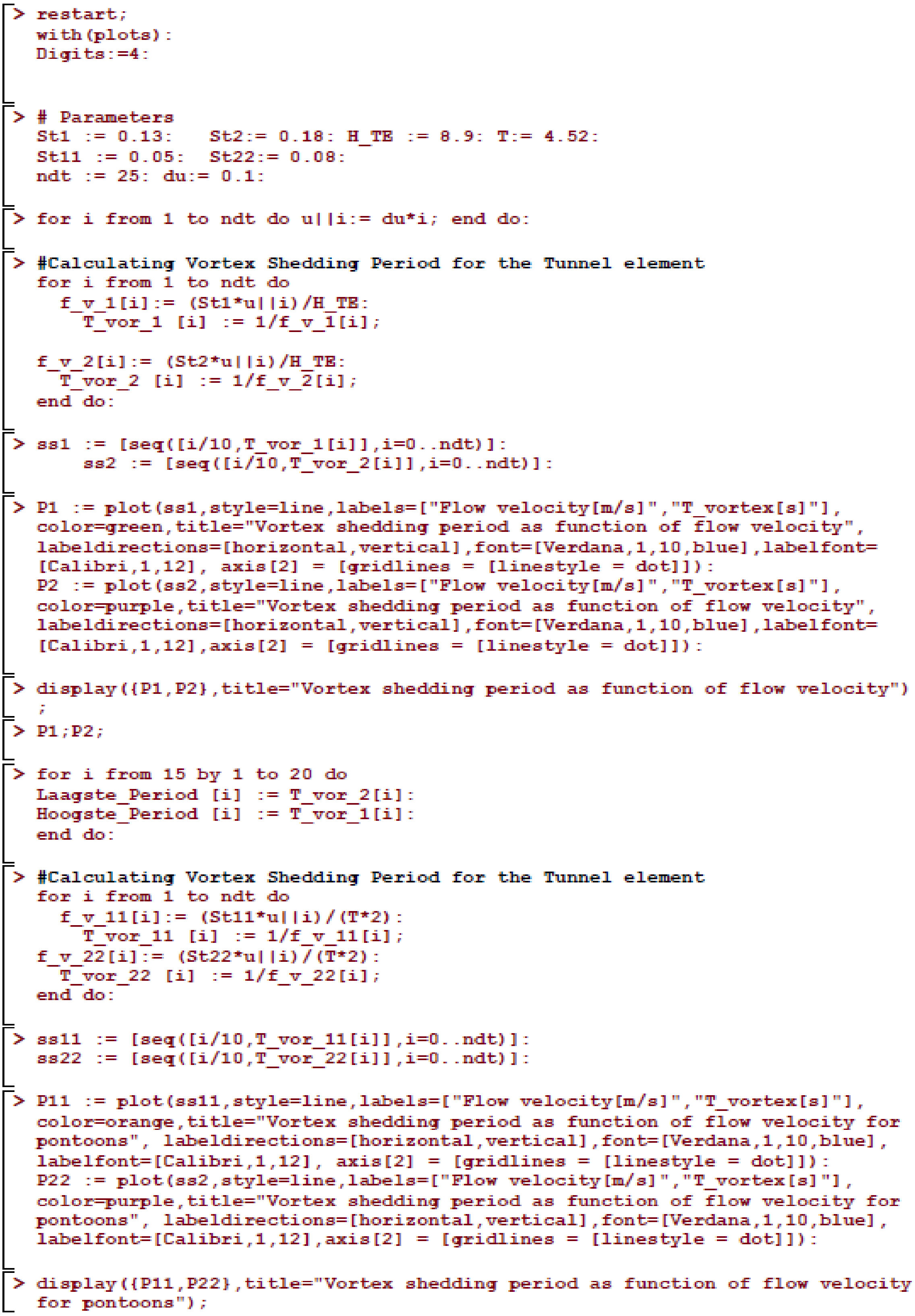
```
> # Calculating Static stability for Catamaran (with hanging tunnel element)
  KG_2 := ((2*M_f*Z_f)+ (M_D*Z_D)+ (W_equip*Z_equip)+ (W_ballast*Z_W_ballast))/
  (M_p+W_ballast):
    A_w_C := B_f*L_f*2:
       T_C_2 := evalf((M_p+W_ballast)/(rho*A_w_C)):
         dis_V_C_2 := (B_f*L_f*T_C_2)*2:
           KB_C_2 := T_C_2/2:
            I_T_2 := evalf((((1/12)*L_f*(B_f^3)) +  (L_f*B_f*evalf((((B_p/2)-
  (B_f/2))^2))))*2):
               I_L_2 := evalf((1/12)*B_f*(L_f^3)) :
                 BM_T_C_2 :=  evalf(I_T_2/ dis_V_C_2):
                   BM_L_C_2 := evalf(I_L_2/ dis_V_C_2):

  h_m_C_T_2 := KB_C_2+BM_T_C_2-KG_2:
  h_m_C_L_1 := KB_C_2+BM_L_C_2-KG_2:

> #Settin up the Force raqnge
```



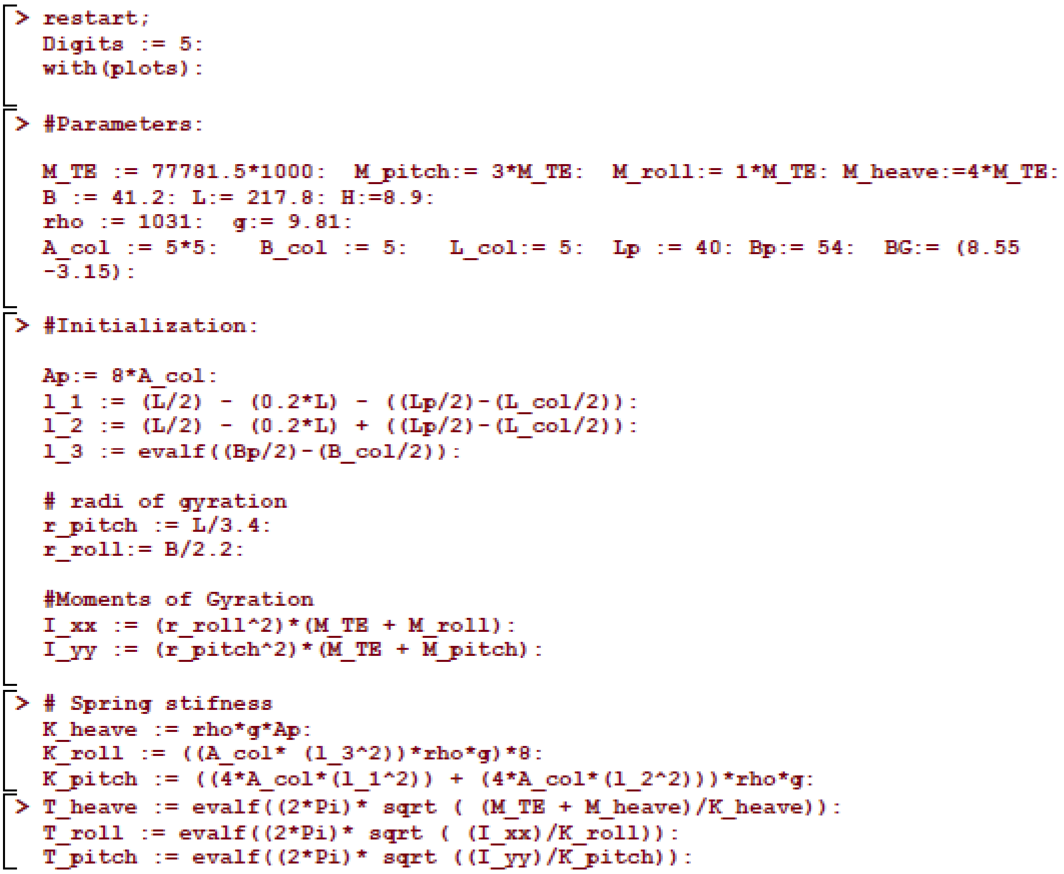
```
  for i from 0 to 20 do
    W_ball[i] := del_W_ballast*i + 700:
  end do:

> # Calculating Static stability for Catamaran (with hanging tunnel element)
  # for fluctuating cable force
> for i from 0 to 20 do
  KG_[i] := ((2*M_f*Z_f)+ (M_D*Z_D)+ (W_equip*Z_equip)+ (W_ball[i]*Z_W_ballast))/
  (M_p+W_ballast):
    A_w_C[i] := B_f*L_f*2:
      T_C_[i] := evalf((M_p+W_ball[i])/(rho*A_w_C[i])):
        dis_V_C_[i] := (B_f*L_f*T_C_[i])*2;
          KB_C_[i] := T_C_[i]/2:
           I_T_[i] := evalf((((1/12)*L_f*(B_f^3)) +  (L_f*B_f*evalf((((B_p/2)-
  (B_f/2))^2))))*2):
                    I_L_[i] := evalf((1/12)*B_f*(L_f^3)) :
                           BM_T_C_[i] :=  evalf(I_T_[i]/ dis_V_C_[i]):

                              BM_L_C_[i] := evalf(I_L_[i]/ dis_V_C_[i]):

  h_m_C_T_[i] := KB_C_[i]+BM_T_C_[i]-KG_[i]:
  h_m_C_L_[i] := KB_C_[i]+BM_L_C_[i]-KG_[i]:

  end do:


> # Plotting the results
  ss := [seq([(i*50+700)/2,h_m_C_T_[i]],i=0..20)]:
  ss1 := [seq([(i*50+700)/2,h_m_C_L_[i]],i=0..20)]:
  ss2 := [seq([(i*50+700)/2,T_C_[i]],i=0..20)]:
  ss3 := [seq([(i*50+700)/2, BM_L_C_[i]],i=0..20)]:

> P := plot(ss,style=line,labels=["Force suspension cables[ton]","hm(t) [m]"],
  color=purple,title="hm(t) as funtion of force fluctuation in suspension cables
  ", labeldirections=[horizontal,vertical],font=[Verdana,1,10,blue],labelfont=
  [Calibri,1,12], axis[2] = [gridlines = [linestyle = dot]]):
  P1 := plot(ss1,style=line,labels=["Force suspention cables[ton]","hm(l) [m]"],
  color=green,title="hm(l) as funtion of force fluctuation in suspension cables
  ", labeldirections=[horizontal,vertical],font=[Verdana,1,10,blue],labelfont=
  [Calibri,1,12],axis[2] = [gridlines = [linestyle = dot]]):
  P2 := plot(ss2,style=line,labels=["Force suspention cables[ton]","Draught T [m]
  "], color=red,title="drauht T as funtion of force fluctuation in suspension
  cables ", labeldirections=[horizontal,vertical],font=[Verdana,1,10,blue],
  labelfont=[Calibri,1,12],axis[2] = [gridlines = [linestyle = dot]]):
  P3 := plot(ss3,style=line,labels=["Force suspention cables[ton]","distance BM
  [m]"], color=blue,title="BM(l) as funtion of force fluctuation in suspension
  cables ", labeldirections=[horizontal,vertical],font=[Verdana,1,10,blue],
  labelfont=[Calibri,1,12],axis[2] = [gridlines = [linestyle = dot]]):
  P;P1;P2;P3;
```

7.2 CALCULATION FILES FLOW FORCES

Calculating Vortex Shedding Period TE

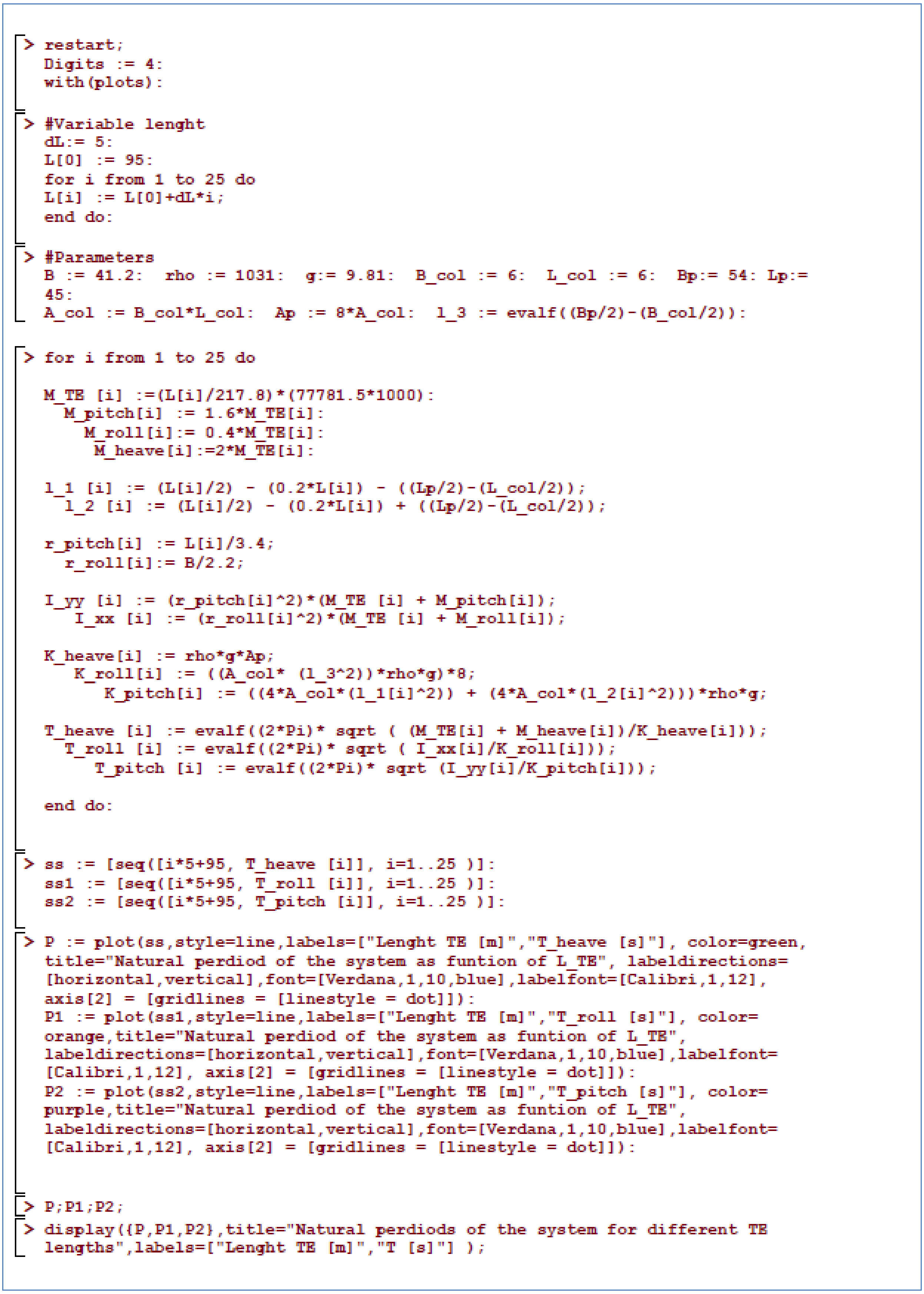


```
> restart;
  with(plots):
  Digits:=4:

> # Parameters
  St1 := 0.13:    St2:= 0.18: H_TE := 8.9: T:= 4.52:
  St11 := 0.05:   St22:= 0.08:
  ndt := 25: du:= 0.1:

> for i from 1 to ndt do u||i:= du*i; end do:

> #Calculating Vortex Shedding Period for the Tunnel element
  for i from 1 to ndt do
    f_v_1[i]:= (St1*u||i)/H_TE:
      T_vor_1 [i] := 1/f_v_1[i];

  f_v_2[i]:= (St2*u||i)/H_TE:
    T_vor_2 [i] := 1/f_v_2[i];
  end do:

> ss1 := [seq([i/10,T_vor_1[i]],i=0..ndt)]:
      ss2 := [seq([i/10,T_vor_2[i]],i=0..ndt)]:

> P1 := plot(ss1,style=line,labels=["Flow velocity[m/s]","T_vortex[s]"],
  color=green,title="Vortex shedding period as function of flow velocity",
  labeldirections=[horizontal,vertical],font=[Verdana,1,10,blue],labelfont=
  [Calibri,1,12], axis[2] = [gridlines = [linestyle = dot]]):
  P2 := plot(ss2,style=line,labels=["Flow velocity[m/s]","T_vortex[s]"],
  color=purple,title="Vortex shedding period as function of flow velocity",
  labeldirections=[horizontal,vertical],font=[Verdana,1,10,blue],labelfont=
  [Calibri,1,12],axis[2] = [gridlines = [linestyle = dot]]):

> display({P1,P2},title="Vortex shedding period as function of flow velocity")
  ;
> P1;P2;

> for i from 15 by 1 to 20 do
  Laagste_Period [i] := T_vor_2[i]:
  Hoogste_Period [i] := T_vor_1[i]:
  end do:

> #Calculating Vortex Shedding Period for the Tunnel element
  for i from 1 to ndt do
    f_v_11[i]:= (St11*u||i)/(T*2):
      T_vor_11 [i] := 1/f_v_11[i];
  f_v_22[i]:= (St22*u||i)/(T*2):
    T_vor_22 [i] := 1/f_v_22[i];
  end do:

> ss11 := [seq([i/10,T_vor_11[i]],i=0..ndt)]:
  ss22 := [seq([i/10,T_vor_22[i]],i=0..ndt)]:

> P11 := plot(ss11,style=line,labels=["Flow velocity[m/s]","T_vortex[s]"],
  color=orange,title="Vortex shedding period as function of flow velocity for
  pontoons", labeldirections=[horizontal,vertical],font=[Verdana,1,10,blue],
  labelfont=[Calibri,1,12], axis[2] = [gridlines = [linestyle = dot]]):
  P22 := plot(ss2,style=line,labels=["Flow velocity[m/s]","T_vortex[s]"],
  color=purple,title="Vortex shedding period as function of flow velocity for
  pontoons", labeldirections=[horizontal,vertical],font=[Verdana,1,10,blue],
  labelfont=[Calibri,1,12],axis[2] = [gridlines = [linestyle = dot]]):

> display({P11,P22},title="Vortex shedding period as function of flow velocity
  for pontoons");
```

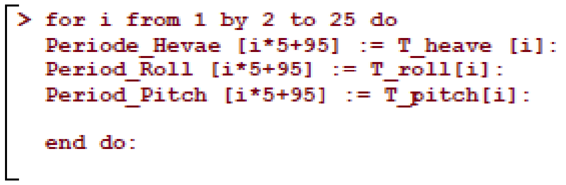


```
> restart;
  Digits := 5:
  with(plots):

> #Parameters:

  M_TE := 77781.5*1000:  M_pitch:= 3*M_TE:  M_roll:= 1*M_TE: M_heave:=4*M_TE:
  B := 41.2: L:= 217.8: H:=8.9:
  rho := 1031:  g:= 9.81:
  A_col := 5*5:   B_col := 5:   L_col:= 5:  Lp := 40: Bp:= 54:  BG:= (8.55
  -3.15):

> #Initialization:

  Ap:= 8*A_col:
  l_1 := (L/2) - (0.2*L) - ((Lp/2)-(L_col/2)):
  l_2 := (L/2) - (0.2*L) + ((Lp/2)-(L_col/2)):
  l_3 := evalf((Bp/2)-(B_col/2)):

  # radi of gyration
  r_pitch := L/3.4:
  r_roll:= B/2.2:

  #Moments of Gyration
  I_xx := (r_roll^2)*(M_TE + M_roll):
  I_yy := (r_pitch^2)*(M_TE + M_pitch):

> # Spring stifness
  K_heave := rho*g*Ap:
  K_roll := ((A_col* (l_3^2))*rho*g)*8:
  K_pitch := ((4*A_col*(l_1^2)) + (4*A_col*(l_2^2)))*rho*g:
> T_heave := evalf((2*Pi)* sqrt ( (M_TE + M_heave)/K_heave)):
  T_roll := evalf((2*Pi)* sqrt ( (I_xx)/K_roll)):
  T_pitch := evalf((2*Pi)* sqrt ((I_yy)/K_pitch)):
```

Natural periods Semi-Submersible as function of TE

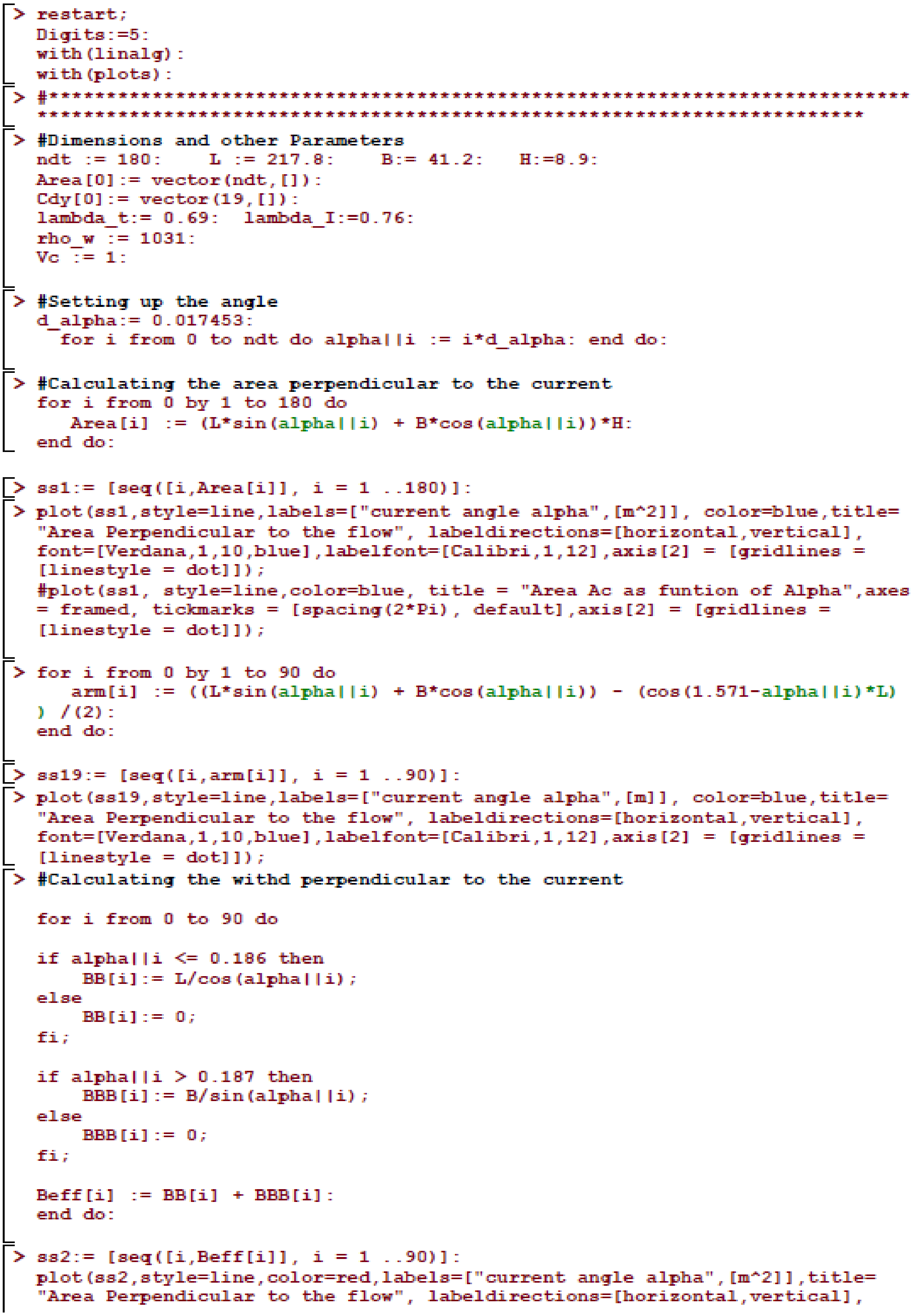


```
> restart;
  Digits := 4:
  with(plots):

> #Variable lenght
  dL:= 5:
  L[0] := 95:
  for i from 1 to 25 do
  L[i] := L[0]+dL*i;
  end do:

> #Parameters
  B := 41.2:  rho := 1031:  g:= 9.81:  B_col := 6:  L_col := 6:  Bp:= 54: Lp:=
  45:
  A_col := B_col*L_col:  Ap := 8*A_col:  l_3 := evalf((Bp/2)-(B_col/2)):

> for i from 1 to 25 do

  M_TE [i] :=(L[i]/217.8)*(77781.5*1000):
    M_pitch[i] := 1.6*M_TE[i]:
      M_roll[i]:= 0.4*M_TE[i]:
       M_heave[i]:=2*M_TE[i]:

  l_1 [i] := (L[i]/2) - (0.2*L[i]) - ((Lp/2)-(L_col/2));
    l_2 [i] := (L[i]/2) - (0.2*L[i]) + ((Lp/2)-(L_col/2));

  r_pitch[i] := L[i]/3.4;
    r_roll[i]:= B/2.2;

  I_yy [i] := (r_pitch[i]^2)*(M_TE [i] + M_pitch[i]);
     I_xx [i] := (r_roll[i]^2)*(M_TE [i] + M_roll[i]);

  K_heave[i] := rho*g*Ap;
     K_roll[i] := ((A_col* (l_3^2))*rho*g)*8;
        K_pitch[i] := ((4*A_col*(l_1[i]^2)) + (4*A_col*(l_2[i]^2)))*rho*g;

  T_heave [i] := evalf((2*Pi)* sqrt ( (M_TE[i] + M_heave[i])/K_heave[i]));
    T_roll [i] := evalf((2*Pi)* sqrt ( I_xx[i]/K_roll[i]));
       T_pitch [i] := evalf((2*Pi)* sqrt (I_yy[i]/K_pitch[i]));

  end do:


> ss := [seq([i*5+95, T_heave [i]], i=1..25 )]:
  ss1 := [seq([i*5+95, T_roll [i]], i=1..25 )]:
  ss2 := [seq([i*5+95, T_pitch [i]], i=1..25 )]:

> P := plot(ss,style=line,labels=["Lenght TE [m]","T_heave [s]"], color=green,
  title="Natural perdiod of the system as funtion of L_TE", labeldirections=
  [horizontal,vertical],font=[Verdana,1,10,blue],labelfont=[Calibri,1,12],
  axis[2] = [gridlines = [linestyle = dot]]):
  P1 := plot(ss1,style=line,labels=["Lenght TE [m]","T_roll [s]"], color=
  orange,title="Natural perdiod of the system as funtion of L_TE",
  labeldirections=[horizontal,vertical],font=[Verdana,1,10,blue],labelfont=
  [Calibri,1,12], axis[2] = [gridlines = [linestyle = dot]]):
  P2 := plot(ss2,style=line,labels=["Lenght TE [m]","T_pitch [s]"], color=
  purple,title="Natural perdiod of the system as funtion of L_TE",
  labeldirections=[horizontal,vertical],font=[Verdana,1,10,blue],labelfont=
  [Calibri,1,12], axis[2] = [gridlines = [linestyle = dot]]):


> P;P1;P2;
> display({P,P1,P2},title="Natural perdiods of the system for different TE
  lengths",labels=["Lenght TE [m]","T [s]"] );
```

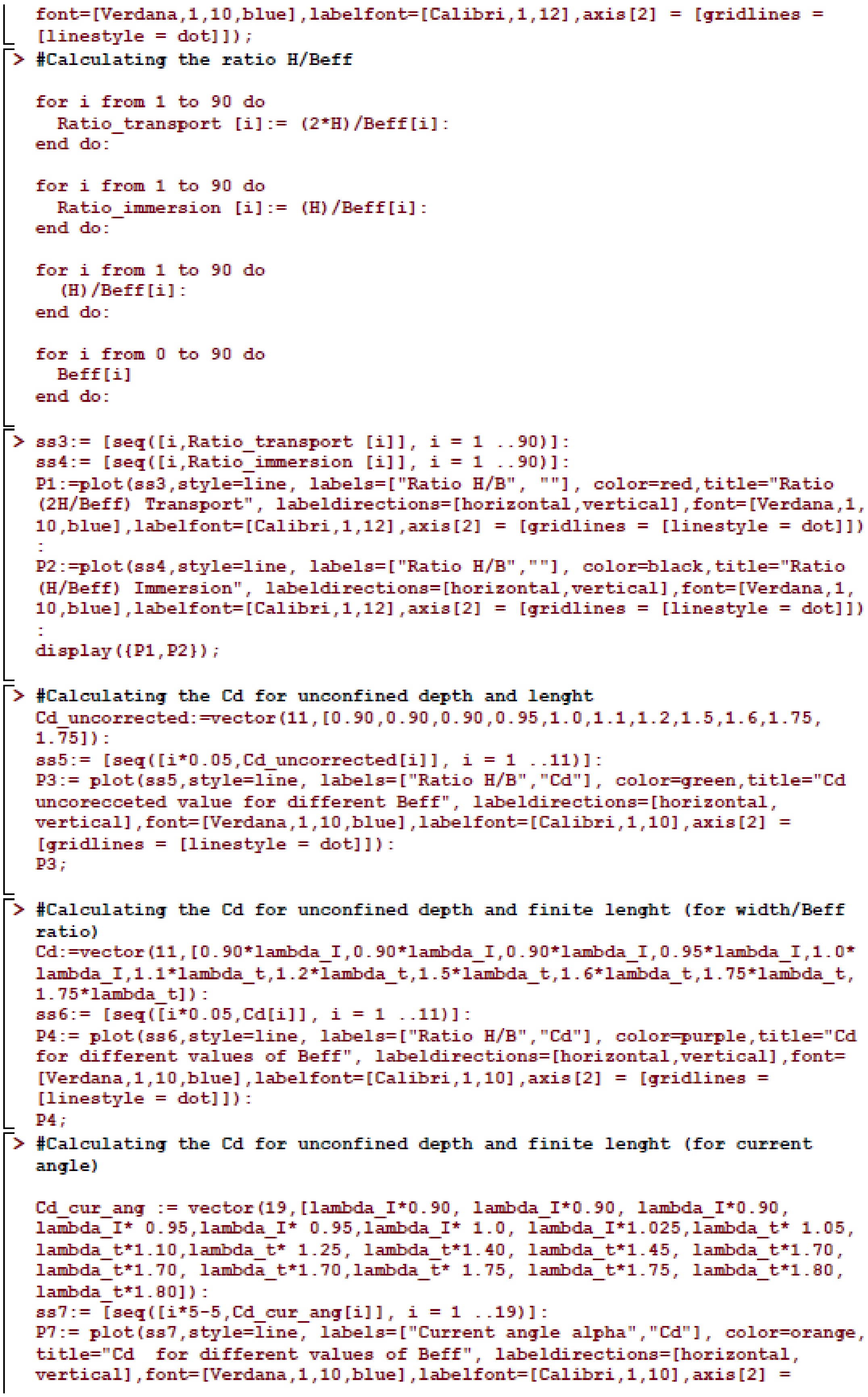


```
> for i from 1 by 2 to 25 do
  Periode_Hevae [i*5+95] := T_heave [i]:
  Period_Roll [i*5+95] := T_roll[i]:
  Period_Pitch [i*5+95] := T_pitch[i]:

  end do:
```

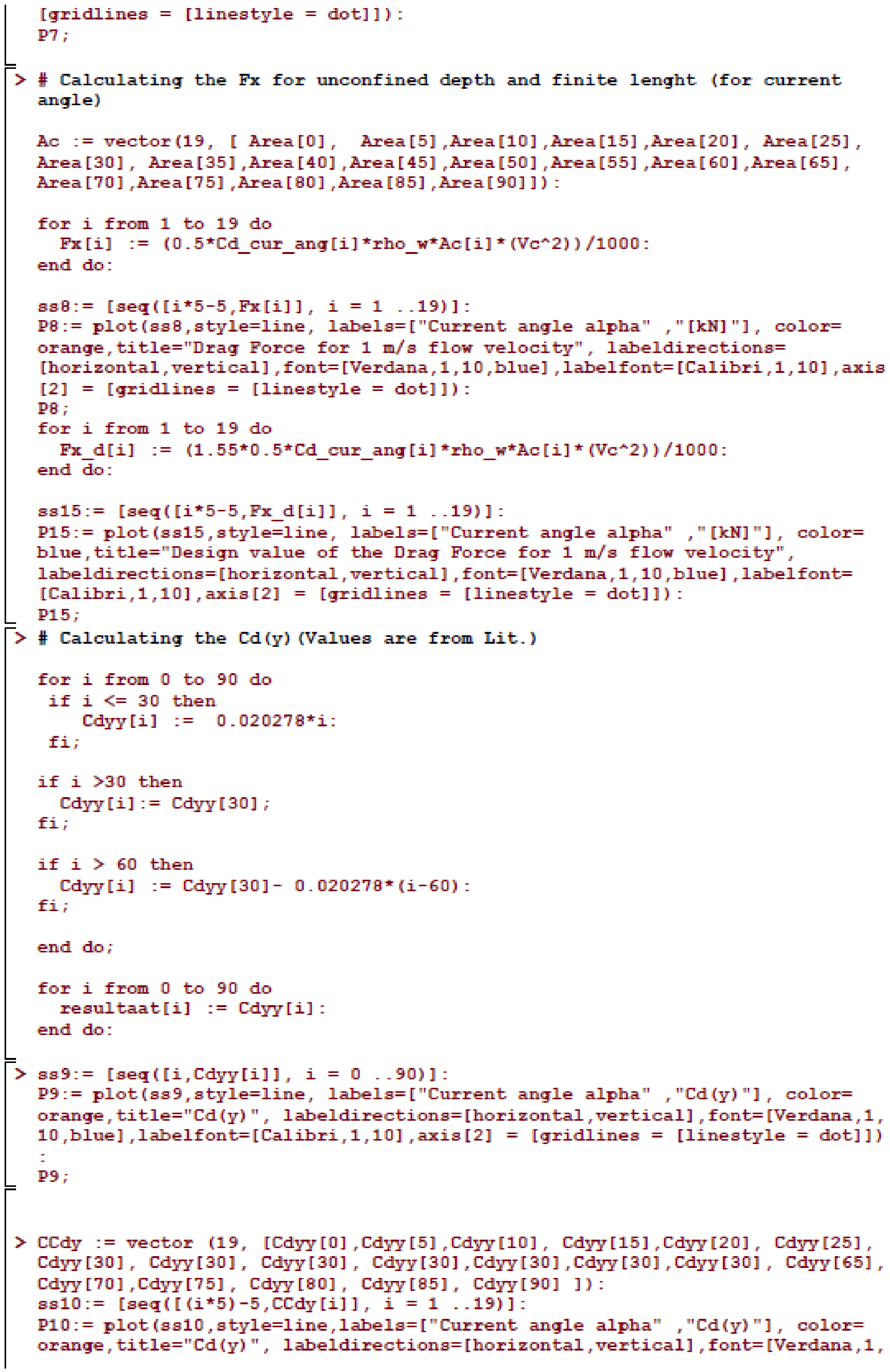
Forces on TE during different positions



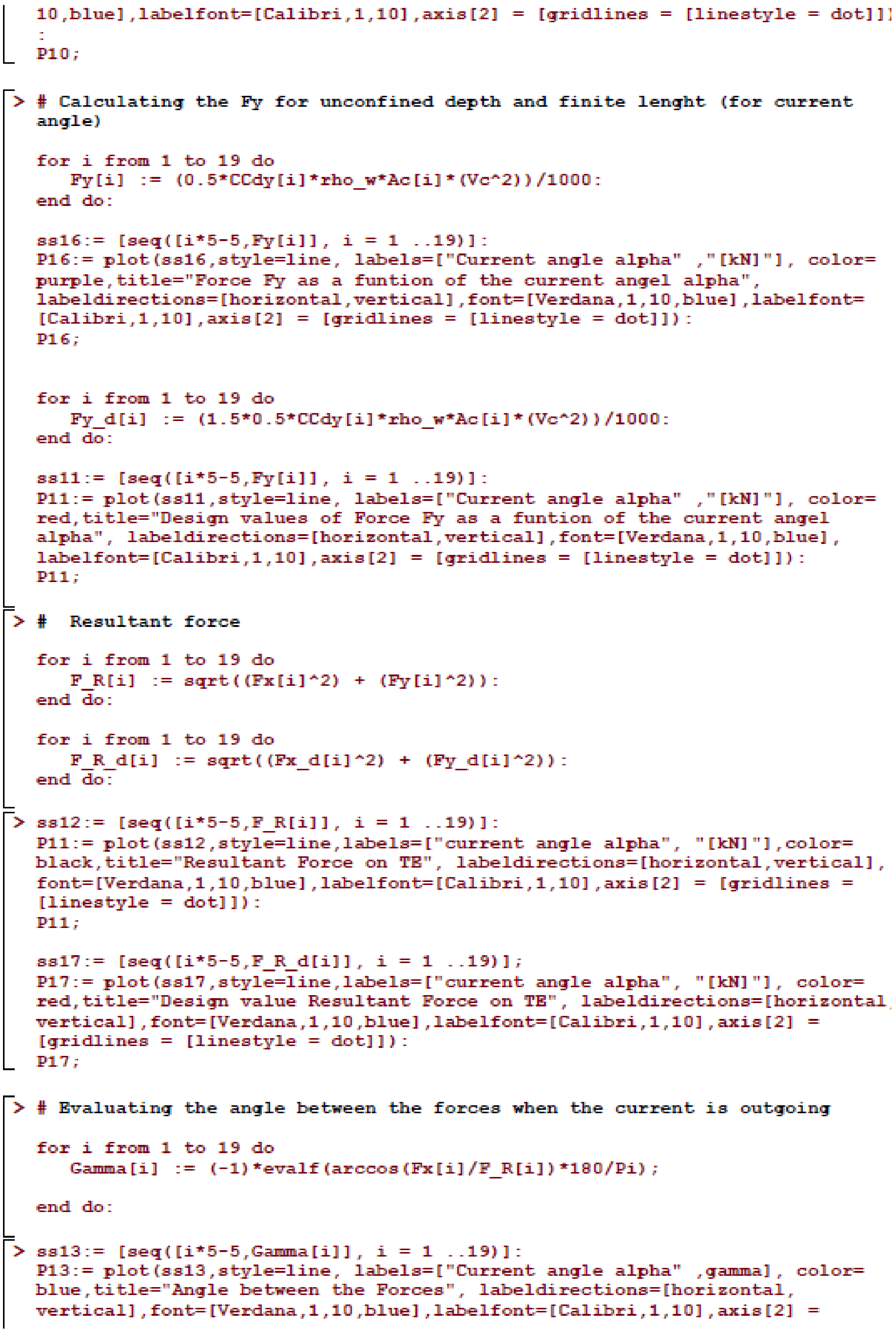
```
> restart;
  Digits:=5:
  with(linalg):
  with(plots):
> #*************************************************************************
  ***********************************************************************
> #Dimensions and other Parameters
  ndt := 180:    L := 217.8:    B:= 41.2:   H:=8.9:
  Area[0]:= vector(ndt,[]):
  Cdy[0]:= vector(19,[]):
  lambda_t:= 0.69:  lambda_I:=0.76:
  rho_w := 1031:
  Vc  := 1:

> #Setting up the angle
  d_alpha:= 0.017453:
    for i from 0 to ndt do alpha||i := i*d_alpha: end do:

> #Calculating the area perpendicular to the current
  for i from 0 by 1 to 180 do
     Area[i]  := (L*sin(alpha||i) + B*cos(alpha||i))*H:
  end do:

> ss1:= [seq([i,Area[i]], i = 1 ..180)]:
> plot(ss1,style=line,labels=["current angle alpha",[m^2]], color=blue,title=
  "Area Perpendicular to the flow", labeldirections=[horizontal,vertical],
  font=[Verdana,1,10,blue],labelfont=[Calibri,1,12],axis[2] = [gridlines =
  [linestyle = dot]]);
  #plot(ss1, style=line,color=blue, title = "Area Ac as funtion of Alpha",axes
  = framed, tickmarks = [spacing(2*Pi), default],axis[2] = [gridlines =
  [linestyle = dot]]);

> for i from 0 by 1 to 90 do
     arm[i]  := ((L*sin(alpha||i) + B*cos(alpha||i)) - (cos(1.571-alpha||i)*L)
  ) /(2):
  end do:

> ss19:= [seq([i,arm[i]], i = 1 ..90)]:
> plot(ss19,style=line,labels=["current angle alpha",[m]], color=blue,title=
  "Area Perpendicular to the flow", labeldirections=[horizontal,vertical],
  font=[Verdana,1,10,blue],labelfont=[Calibri,1,12],axis[2] = [gridlines =
  [linestyle = dot]]);
> #Calculating the withd perpendicular to the current

  for i from 0 to 90 do

  if alpha||i <= 0.186 then
      BB[i]:= L/cos(alpha||i);
  else
      BB[i]:= 0;
  fi;

  if alpha||i > 0.187 then
      BBB[i]:= B/sin(alpha||i);
  else
      BBB[i]:= 0;
  fi;

  Beff[i]  := BB[i] + BBB[i]:
  end do:

> ss2:= [seq([i,Beff[i]], i = 1 ..90)]:
  plot(ss2,style=line,color=red,labels=["current angle alpha",[m^2]],title=
  "Area Perpendicular to the flow", labeldirections=[horizontal,vertical],
```



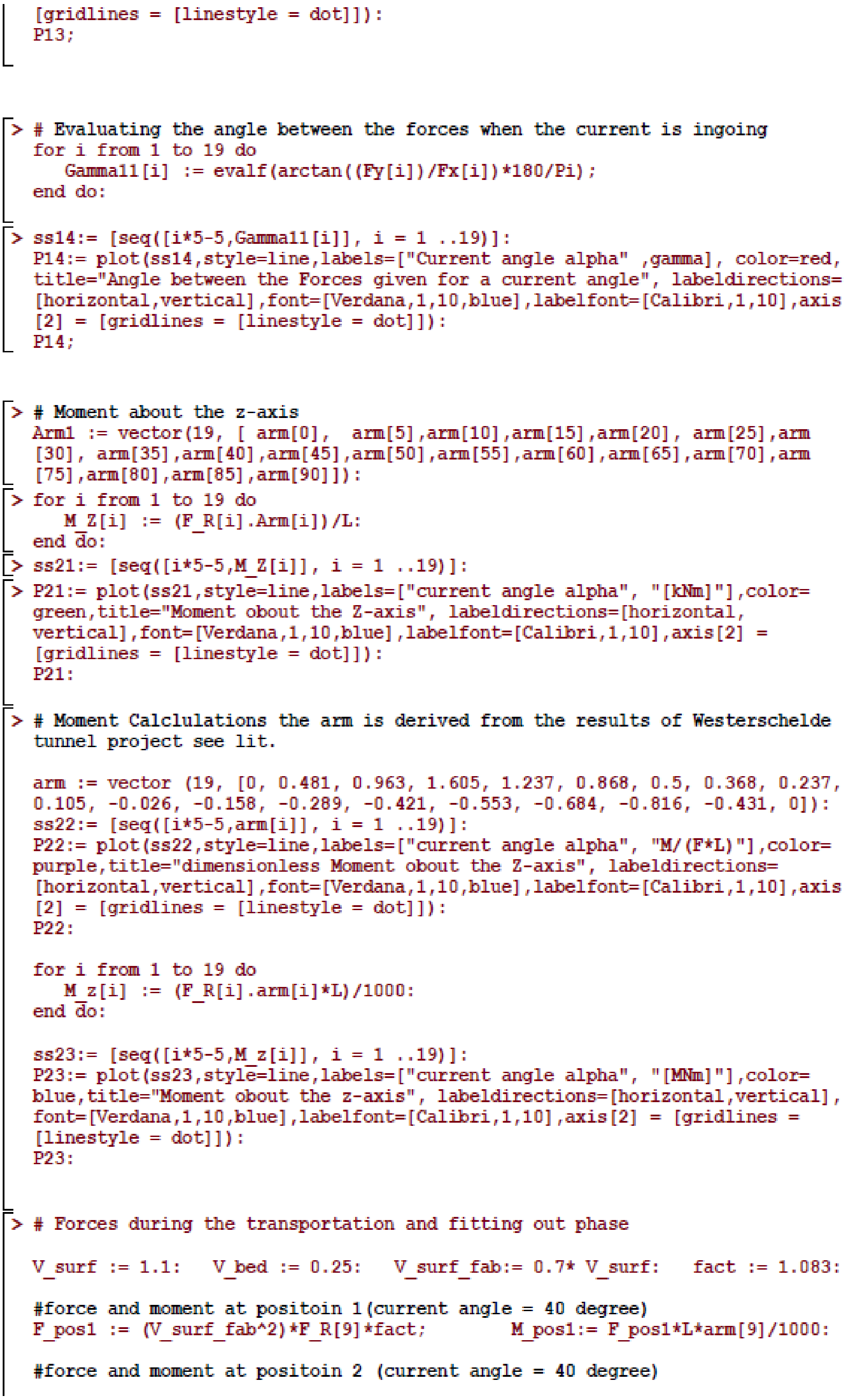
```
  font=[Verdana,1,10,blue],labelfont=[Calibri,1,12],axis[2] = [gridlines =
  [linestyle = dot]]);
> #Calculating the ratio H/Beff

  for i from 1 to 90 do
    Ratio_transport [i]:= (2*H)/Beff[i]:
  end do:

  for i from 1 to 90 do
    Ratio_immersion [i]:= (H)/Beff[i]:
  end do:

  for i from 1 to 90 do
    (H)/Beff[i]:
  end do:

  for i from 0 to 90 do
    Beff[i]
  end do:

> ss3:= [seq([i,Ratio_transport [i]], i = 1 ..90)]:
  ss4:= [seq([i,Ratio_immersion [i]], i = 1 ..90)]:
  P1:=plot(ss3,style=line, labels=["Ratio H/B", ""], color=red,title="Ratio
  (2H/Beff) Transport", labeldirections=[horizontal,vertical],font=[Verdana,1,
  10,blue],labelfont=[Calibri,1,12],axis[2] = [gridlines = [linestyle = dot]])
  :
  P2:=plot(ss4,style=line, labels=["Ratio H/B",""], color=black,title="Ratio
  (H/Beff) Immersion", labeldirections=[horizontal,vertical],font=[Verdana,1,
  10,blue],labelfont=[Calibri,1,12],axis[2] = [gridlines = [linestyle = dot]])
  :
  display({P1,P2});

> #Calculating the Cd for unconfined depth and lenght
  Cd_uncorrected:=vector(11,[0.90,0.90,0.90,0.95,1.0,1.1,1.2,1.5,1.6,1.75,
  1.75]):
  ss5:= [seq([i*0.05,Cd_uncorrected[i]], i = 1 ..11)]:
  P3:= plot(ss5,style=line, labels=["Ratio H/B","Cd"], color=green,title="Cd
  uncorecceted value for different Beff", labeldirections=[horizontal,
  vertical],font=[Verdana,1,10,blue],labelfont=[Calibri,1,10],axis[2] =
  [gridlines = [linestyle = dot]]):
  P3;

> #Calculating the Cd for unconfined depth and finite lenght (for width/Beff
  ratio)
  Cd:=vector(11,[0.90*lambda_I,0.90*lambda_I,0.90*lambda_I,0.95*lambda_I,1.0*
  lambda_I,1.1*lambda_t,1.2*lambda_t,1.5*lambda_t,1.6*lambda_t,1.75*lambda_t,
  1.75*lambda_t]):
  ss6:= [seq([i*0.05,Cd[i]], i = 1 ..11)]:
  P4:= plot(ss6,style=line, labels=["Ratio H/B","Cd"], color=purple,title="Cd
  for different values of Beff", labeldirections=[horizontal,vertical],font=
  [Verdana,1,10,blue],labelfont=[Calibri,1,10],axis[2] = [gridlines =
  [linestyle = dot]]):
  P4;
> #Calculating the Cd for unconfined depth and finite lenght (for current
  angle)

  Cd_cur_ang := vector(19,[lambda_I*0.90, lambda_I*0.90, lambda_I*0.90,
  lambda_I* 0.95,lambda_I* 0.95,lambda_I* 1.0, lambda_I*1.025,lambda_t* 1.05,
  lambda_t*1.10,lambda_t* 1.25, lambda_t*1.40, lambda_t*1.45, lambda_t*1.70,
  lambda_t*1.70, lambda_t*1.70,lambda_t* 1.75, lambda_t*1.75, lambda_t*1.80,
  lambda_t*1.80]):
  ss7:= [seq([i*5-5,Cd_cur_ang[i]], i = 1 ..19)]:
  P7:= plot(ss7,style=line, labels=["Current angle alpha","Cd"], color=orange,
  title="Cd  for different values of Beff", labeldirections=[horizontal,
  vertical],font=[Verdana,1,10,blue],labelfont=[Calibri,1,10],axis[2] =
```



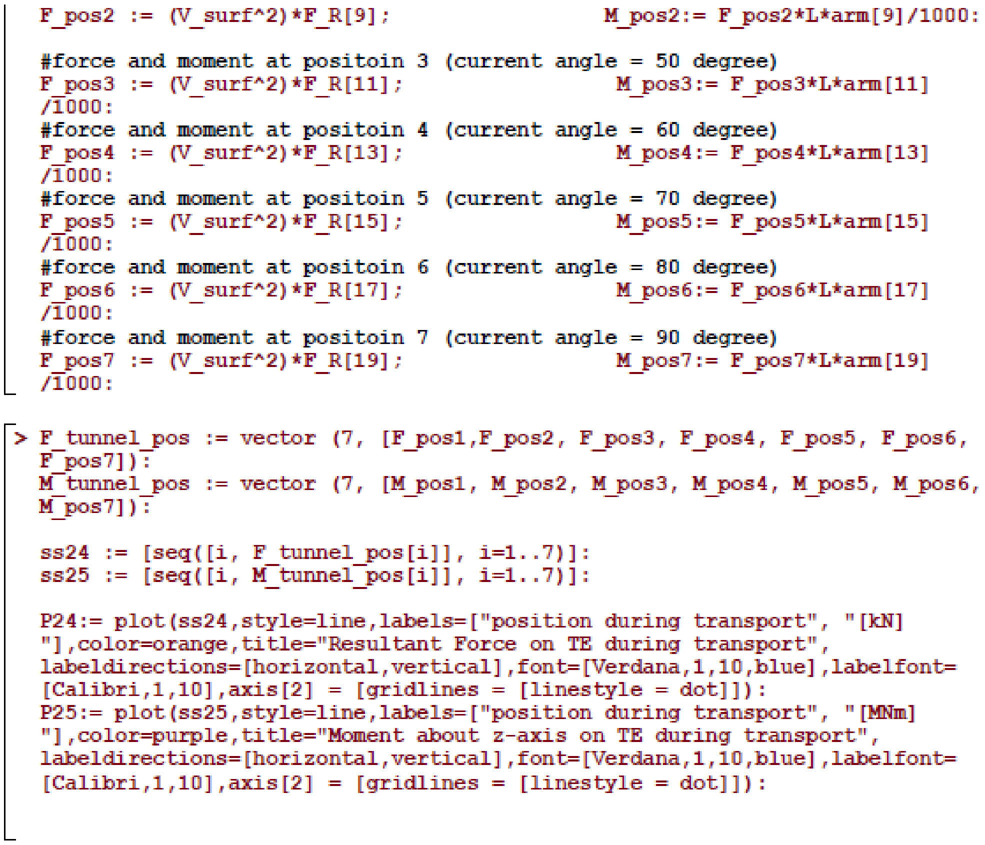
```
[gridlines = [linestyle = dot]]):
P7;

> # Calculating the Fx for unconfined depth and finite lenght (for current
angle)

Ac := vector(19, [ Area[0],  Area[5],Area[10],Area[15],Area[20], Area[25],
Area[30], Area[35],Area[40],Area[45],Area[50],Area[55],Area[60],Area[65],
Area[70],Area[75],Area[80],Area[85],Area[90]]):

for i from 1 to 19 do
  Fx[i] := (0.5*Cd_cur_ang[i]*rho_w*Ac[i]*(Vc^2))/1000:
end do:

ss8:= [seq([i*5-5,Fx[i]], i = 1 ..19)]:
P8:= plot(ss8,style=line, labels=["Current angle alpha" ,"[kN]"], color=
orange,title="Drag Force for 1 m/s flow velocity", labeldirections=
[horizontal,vertical],font=[Verdana,1,10,blue],labelfont=[Calibri,1,10],axis
[2] = [gridlines = [linestyle = dot]]):
P8;
for i from 1 to 19 do
  Fx_d[i] := (1.55*0.5*Cd_cur_ang[i]*rho_w*Ac[i]*(Vc^2))/1000:
end do:

ss15:= [seq([i*5-5,Fx_d[i]], i = 1 ..19)]:
P15:= plot(ss15,style=line, labels=["Current angle alpha" ,"[kN]"], color=
blue,title="Design value of the Drag Force for 1 m/s flow velocity",
labeldirections=[horizontal,vertical],font=[Verdana,1,10,blue],labelfont=
[Calibri,1,10],axis[2] = [gridlines = [linestyle = dot]]):
P15;
> # Calculating the Cd(y)(Values are from Lit.)

for i from 0 to 90 do
 if i <= 30 then
    Cdyy[i] :=  0.020278*i:
 fi;

if i >30 then
  Cdyy[i]:= Cdyy[30];
fi;

if i > 60 then
  Cdyy[i] := Cdyy[30]- 0.020278*(i-60):
fi;

end do;

for i from 0 to 90 do
  resultaat[i] := Cdyy[i]:
end do:

> ss9:= [seq([i,Cdyy[i]], i = 0 ..90)]:
P9:= plot(ss9,style=line, labels=["Current angle alpha" ,"Cd(y)"], color=
orange,title="Cd(y)", labeldirections=[horizontal,vertical],font=[Verdana,1,
10,blue],labelfont=[Calibri,1,10],axis[2] = [gridlines = [linestyle = dot]])
:
P9;


> CCdy := vector (19, [Cdyy[0],Cdyy[5],Cdyy[10], Cdyy[15],Cdyy[20], Cdyy[25],
Cdyy[30], Cdyy[30], Cdyy[30], Cdyy[30],Cdyy[30],Cdyy[30],Cdyy[30], Cdyy[65],
Cdyy[70],Cdyy[75], Cdyy[80], Cdyy[85], Cdyy[90] ]):
ss10:= [seq([(i*5)-5,CCdy[i]], i = 1 ..19)]:
P10:= plot(ss10,style=line,labels=["Current angle alpha" ,"Cd(y)"], color=
orange,title="Cd(y)", labeldirections=[horizontal,vertical],font=[Verdana,1,
```



```
10,blue],labelfont=[Calibri,1,10],axis[2] = [gridlines = [linestyle = dot]]
:
P10;

> # Calculating the Fy for unconfined depth and finite lenght (for current
angle)

for i from 1 to 19 do
   Fy[i] := (0.5*CCdy[i]*rho_w*Ac[i]*(Vc^2))/1000:
end do:

ss16:= [seq([i*5-5,Fy[i]], i = 1 ..19)]:
P16:= plot(ss16,style=line, labels=["Current angle alpha" ,"[kN]"], color=
purple,title="Force Fy as a funtion of the current angel alpha",
labeldirections=[horizontal,vertical],font=[Verdana,1,10,blue],labelfont=
[Calibri,1,10],axis[2] = [gridlines = [linestyle = dot]]):
P16;


for i from 1 to 19 do
   Fy_d[i] := (1.5*0.5*CCdy[i]*rho_w*Ac[i]*(Vc^2))/1000:
end do:

ss11:= [seq([i*5-5,Fy[i]], i = 1 ..19)]:
P11:= plot(ss11,style=line, labels=["Current angle alpha" ,"[kN]"], color=
red,title="Design values of Force Fy as a funtion of the current angel
alpha", labeldirections=[horizontal,vertical],font=[Verdana,1,10,blue],
labelfont=[Calibri,1,10],axis[2] = [gridlines = [linestyle = dot]]):
P11;

> #  Resultant force

for i from 1 to 19 do
   F_R[i] := sqrt((Fx[i]^2) + (Fy[i]^2)):
end do:

for i from 1 to 19 do
   F_R_d[i] := sqrt((Fx_d[i]^2) + (Fy_d[i]^2)):
end do:

> ss12:= [seq([i*5-5,F_R[i]], i = 1 ..19)]:
P11:= plot(ss12,style=line,labels=["current angle alpha", "[kN]"],color=
black,title="Resultant Force on TE", labeldirections=[horizontal,vertical],
font=[Verdana,1,10,blue],labelfont=[Calibri,1,10],axis[2] = [gridlines =
[linestyle = dot]]):
P11;

ss17:= [seq([i*5-5,F_R_d[i]], i = 1 ..19)];
P17:= plot(ss17,style=line,labels=["current angle alpha", "[kN]"], color=
red,title="Design value Resultant Force on TE", labeldirections=[horizontal,
vertical],font=[Verdana,1,10,blue],labelfont=[Calibri,1,10],axis[2] =
[gridlines = [linestyle = dot]]):
P17;

> # Evaluating the angle between the forces when the current is outgoing

for i from 1 to 19 do
   Gamma[i] := (-1)*evalf(arccos(Fx[i]/F_R[i])*180/Pi);

end do:

> ss13:= [seq([i*5-5,Gamma[i]], i = 1 ..19)]:
P13:= plot(ss13,style=line, labels=["Current angle alpha" ,gamma], color=
blue,title="Angle between the Forces", labeldirections=[horizontal,
vertical],font=[Verdana,1,10,blue],labelfont=[Calibri,1,10],axis[2] =
```

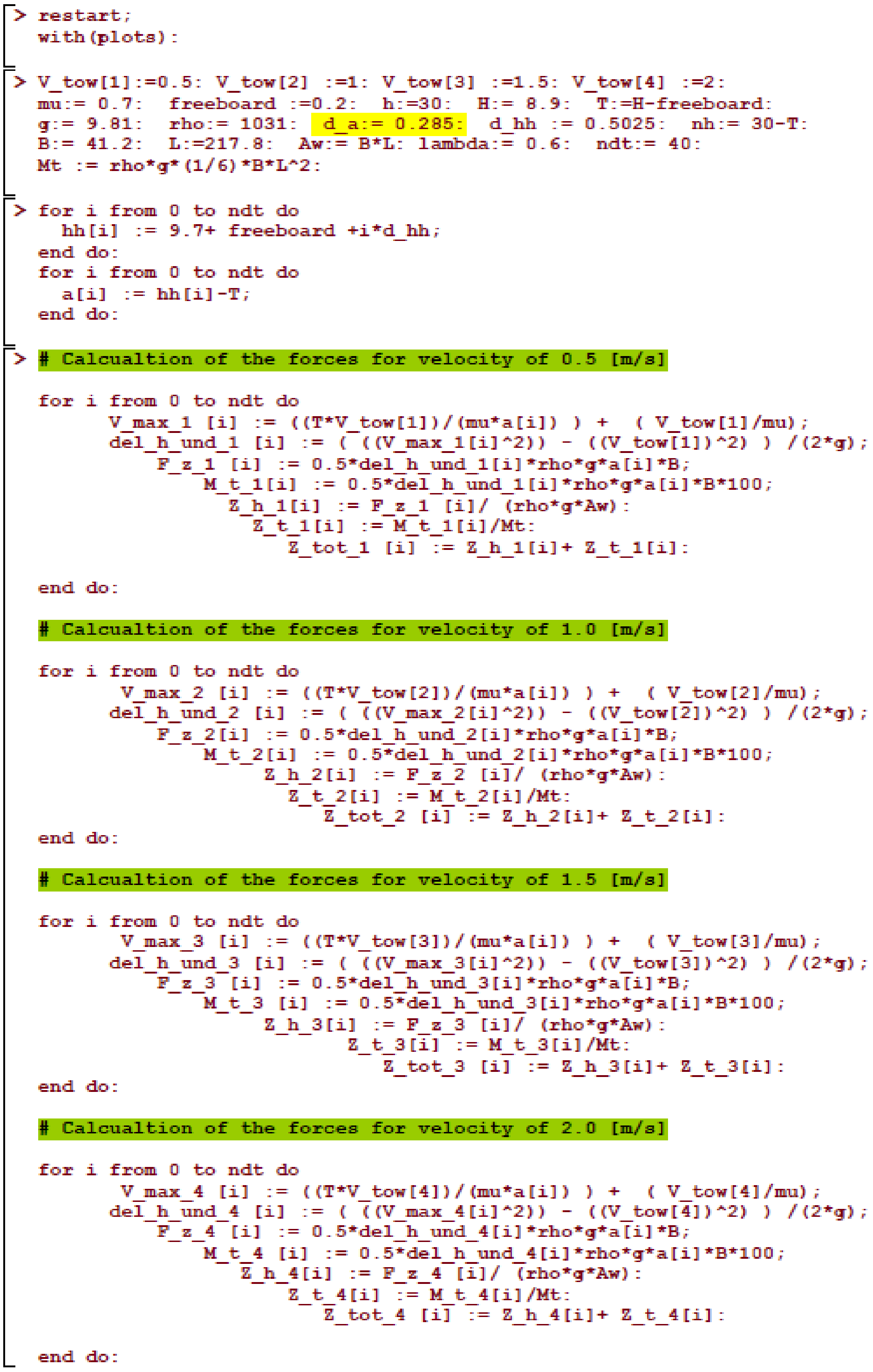


```
[gridlines = [linestyle = dot]]):
P13;

> # Evaluating the angle between the forces when the current is ingoing
for i from 1 to 19 do
   Gamma11[i] := evalf(arctan((Fy[i])/Fx[i])*180/Pi);
end do:

> ss14:= [seq([i*5-5,Gamma11[i]], i = 1 ..19)]:
P14:= plot(ss14,style=line,labels=["Current angle alpha" ,gamma], color=red,
title="Angle between the Forces given for a current angle", labeldirections=
[horizontal,vertical],font=[Verdana,1,10,blue],labelfont=[Calibri,1,10],axis
[2] = [gridlines = [linestyle = dot]]):
P14;

> # Moment about the z-axis
Arm1 := vector(19, [ arm[0],  arm[5],arm[10],arm[15],arm[20], arm[25],arm
[30], arm[35],arm[40],arm[45],arm[50],arm[55],arm[60],arm[65],arm[70],arm
[75],arm[80],arm[85],arm[90]]):
> for i from 1 to 19 do
   M_Z[i] := (F_R[i].Arm[i])/L:
end do:
> ss21:= [seq([i*5-5,M_Z[i]], i = 1 ..19)]:
> P21:= plot(ss21,style=line,labels=["current angle alpha", "[kNm]"],color=
green,title="Moment obout the Z-axis", labeldirections=[horizontal,
vertical],font=[Verdana,1,10,blue],labelfont=[Calibri,1,10],axis[2] =
[gridlines = [linestyle = dot]]):
P21:

> # Moment Calclulations the arm is derived from the results of Westerschelde
tunnel project see lit.

arm := vector (19, [0, 0.481, 0.963, 1.605, 1.237, 0.868, 0.5, 0.368, 0.237,
0.105, -0.026, -0.158, -0.289, -0.421, -0.553, -0.684, -0.816, -0.431, 0]):
ss22:= [seq([i*5-5,arm[i]], i = 1 ..19)]:
P22:= plot(ss22,style=line,labels=["current angle alpha", "M/(F*L)"],color=
purple,title="dimensionless Moment obout the Z-axis", labeldirections=
[horizontal,vertical],font=[Verdana,1,10,blue],labelfont=[Calibri,1,10],axis
[2] = [gridlines = [linestyle = dot]]):
P22:

for i from 1 to 19 do
   M_z[i] := (F_R[i].arm[i]*L)/1000:
end do:

ss23:= [seq([i*5-5,M_z[i]], i = 1 ..19)]:
P23:= plot(ss23,style=line,labels=["current angle alpha", "[MNm]"],color=
blue,title="Moment obout the z-axis", labeldirections=[horizontal,vertical],
font=[Verdana,1,10,blue],labelfont=[Calibri,1,10],axis[2] = [gridlines =
[linestyle = dot]]):
P23:

> # Forces during the transportation and fitting out phase

V_surf := 1.1:   V_bed := 0.25:   V_surf_fab:= 0.7* V_surf:   fact := 1.083:

#force and moment at positoin 1(current angle = 40 degree)
F_pos1 := (V_surf_fab^2)*F_R[9]*fact;        M_pos1:= F_pos1*L*arm[9]/1000:

#force and moment at positoin 2 (current angle = 40 degree)
```

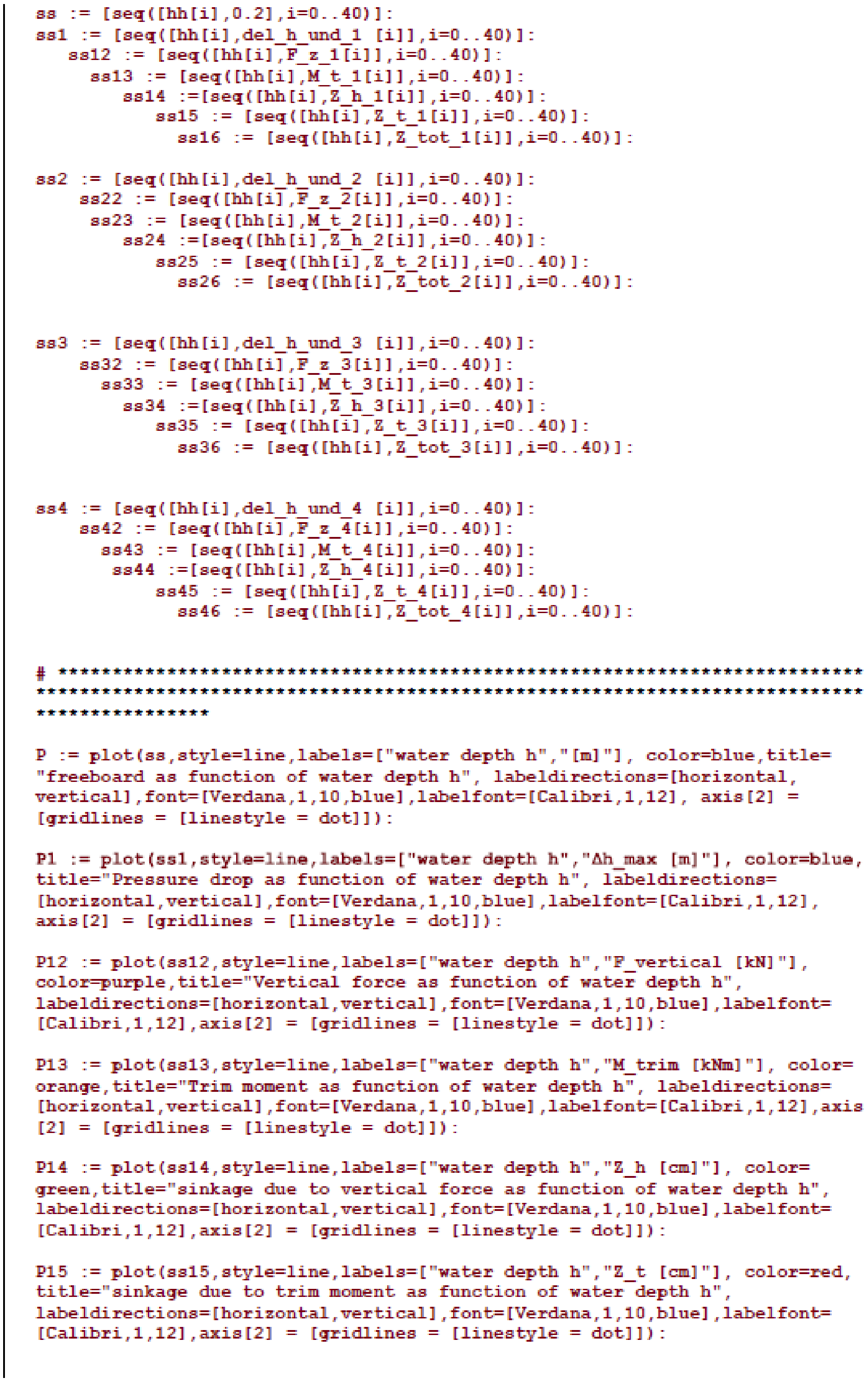


```
F_pos2 := (V_surf^2)*F_R[9];                    M_pos2:= F_pos2*L*arm[9]/1000:

#force and moment at positoin 3 (current angle = 50 degree)
F_pos3 := (V_surf^2)*F_R[11];                   M_pos3:= F_pos3*L*arm[11]
/1000:
#force and moment at positoin 4 (current angle = 60 degree)
F_pos4 := (V_surf^2)*F_R[13];                   M_pos4:= F_pos4*L*arm[13]
/1000:
#force and moment at positoin 5 (current angle = 70 degree)
F_pos5 := (V_surf^2)*F_R[15];                   M_pos5:= F_pos5*L*arm[15]
/1000:
#force and moment at positoin 6 (current angle = 80 degree)
F_pos6 := (V_surf^2)*F_R[17];                   M_pos6:= F_pos6*L*arm[17]
/1000:
#force and moment at positoin 7 (current angle = 90 degree)
F_pos7 := (V_surf^2)*F_R[19];                   M_pos7:= F_pos7*L*arm[19]
/1000:

> F_tunnel_pos := vector (7, [F_pos1,F_pos2, F_pos3, F_pos4, F_pos5, F_pos6,
F_pos7]):
M_tunnel_pos := vector (7, [M_pos1, M_pos2, M_pos3, M_pos4, M_pos5, M_pos6,
M_pos7]):

ss24 := [seq([i, F_tunnel_pos[i]], i=1..7)]:
ss25 := [seq([i, M_tunnel_pos[i]], i=1..7)]:

P24:= plot(ss24,style=line,labels=["position during transport", "[kN]
"],color=orange,title="Resultant Force on TE during transport",
labeldirections=[horizontal,vertical],font=[Verdana,1,10,blue],labelfont=
[Calibri,1,10],axis[2] = [gridlines = [linestyle = dot]]):
P25:= plot(ss25,style=line,labels=["position during transport", "[MNm]
"],color=purple,title="Moment about z-axis on TE during transport",
labeldirections=[horizontal,vertical],font=[Verdana,1,10,blue],labelfont=
[Calibri,1,10],axis[2] = [gridlines = [linestyle = dot]]):
```

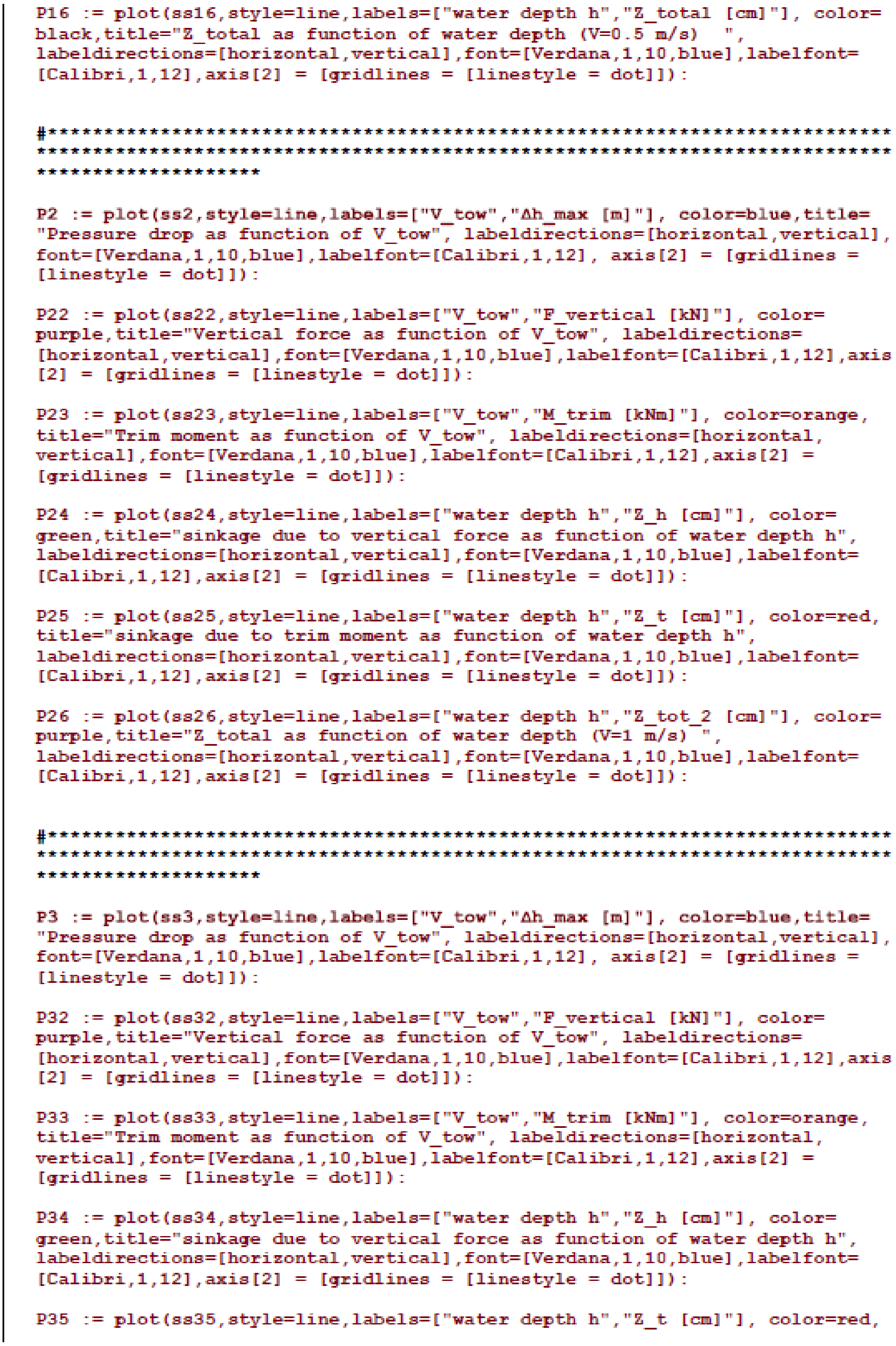
7.3 CALCULATION FILES HYDRODYNAMIC INSTABILITY



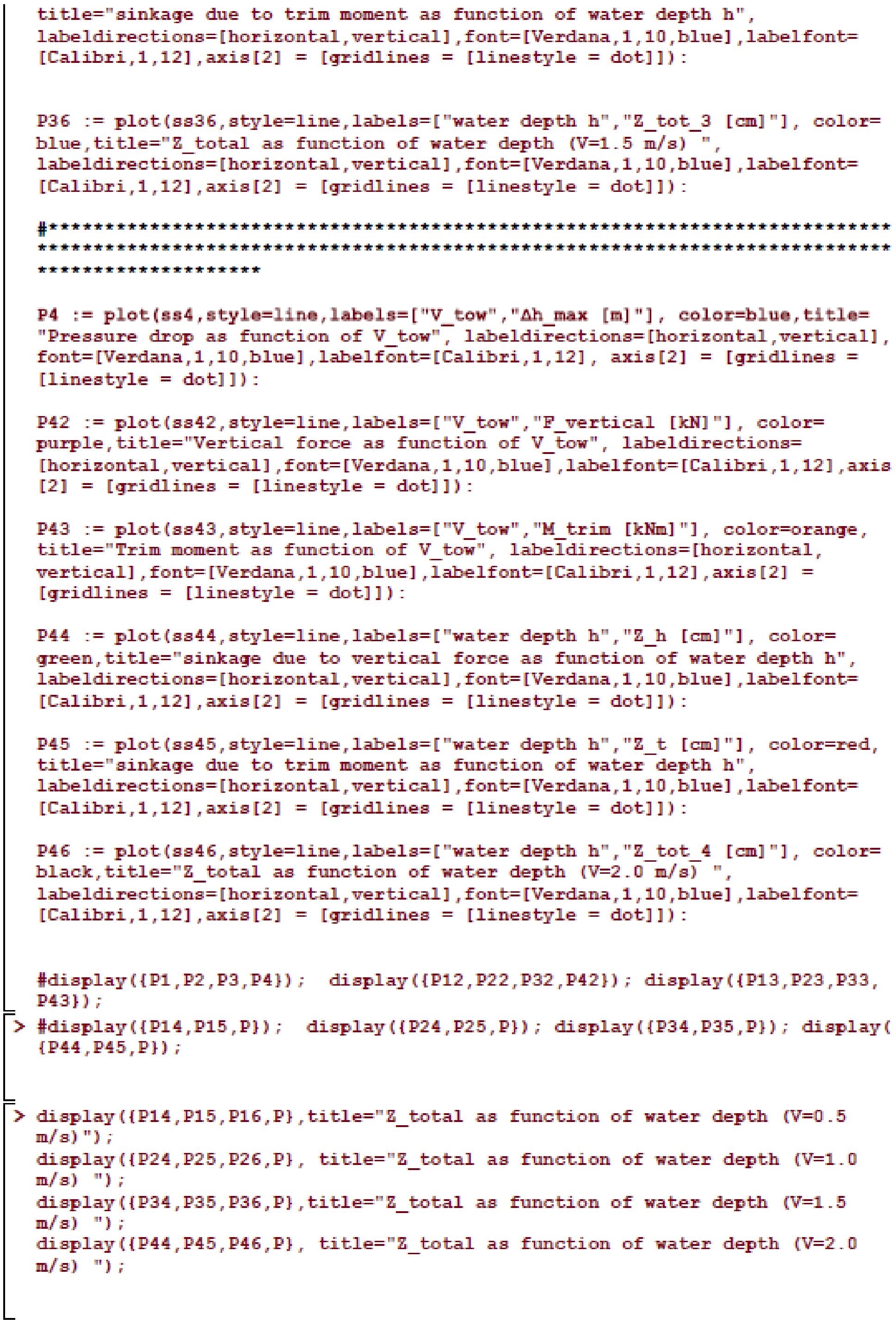
```
> restart;
  with(plots):

> V_tow[1]:=0.5: V_tow[2] :=1: V_tow[3] :=1.5: V_tow[4] :=2:
  mu:= 0.7:  freeboard :=0.2:  h:=30:  H:= 8.9:  T:=H-freeboard:
  g:= 9.81:  rho:= 1031:  d_a:= 0.285:  d_hh := 0.5025:  nh:= 30-T:
  B:= 41.2:  L:=217.8:  Aw:= B*L: lambda:= 0.6:  ndt:= 40:
  Mt := rho*g*(1/6)*B*L^2:

> for i from 0 to ndt do
    hh[i] := 9.7+ freeboard +i*d_hh;
  end do:
  for i from 0 to ndt do
    a[i] := hh[i]-T;
  end do:

> # Calcualtion of the forces for velocity of 0.5 [m/s]

  for i from 0 to ndt do
        V_max_1 [i] := ((T*V_tow[1])/(mu*a[i]) ) +  ( V_tow[1]/mu);
        del_h_und_1 [i] := ( ((V_max_1[i]^2)) - ((V_tow[1])^2) ) /(2*g);
            F_z_1 [i] := 0.5*del_h_und_1[i]*rho*g*a[i]*B;
                M_t_1[i] := 0.5*del_h_und_1[i]*rho*g*a[i]*B*100;
                  Z_h_1[i] := F_z_1 [i]/ (rho*g*Aw):
                    Z_t_1[i] := M_t_1[i]/Mt:
                       Z_tot_1 [i] := Z_h_1[i]+ Z_t_1[i]:

  end do:

  # Calcualtion of the forces for velocity of 1.0 [m/s]

  for i from 0 to ndt do
         V_max_2 [i] := ((T*V_tow[2])/(mu*a[i]) ) +  ( V_tow[2]/mu);
        del_h_und_2 [i] := ( ((V_max_2[i]^2)) - ((V_tow[2])^2) ) /(2*g);
            F_z_2[i] := 0.5*del_h_und_2[i]*rho*g*a[i]*B;
                M_t_2[i] := 0.5*del_h_und_2[i]*rho*g*a[i]*B*100;
                     Z_h_2[i] := F_z_2 [i]/ (rho*g*Aw):
                       Z_t_2[i] := M_t_2[i]/Mt:
                          Z_tot_2 [i] := Z_h_2[i]+ Z_t_2[i]:
  end do:

  # Calcualtion of the forces for velocity of 1.5 [m/s]

  for i from 0 to ndt do
         V_max_3 [i] := ((T*V_tow[3])/(mu*a[i]) ) +  ( V_tow[3]/mu);
        del_h_und_3 [i] := ( ((V_max_3[i]^2)) - ((V_tow[3])^2) ) /(2*g);
            F_z_3 [i] := 0.5*del_h_und_3[i]*rho*g*a[i]*B;
                M_t_3 [i] := 0.5*del_h_und_3[i]*rho*g*a[i]*B*100;
                     Z_h_3[i] := F_z_3 [i]/ (rho*g*Aw):
                            Z_t_3[i] := M_t_3[i]/Mt:
                               Z_tot_3 [i] := Z_h_3[i]+ Z_t_3[i]:
  end do:

  # Calcualtion of the forces for velocity of 2.0 [m/s]

  for i from 0 to ndt do
         V_max_4 [i] := ((T*V_tow[4])/(mu*a[i]) ) +  ( V_tow[4]/mu);
        del_h_und_4 [i] := ( ((V_max_4[i]^2)) - ((V_tow[4])^2) ) /(2*g);
            F_z_4 [i] := 0.5*del_h_und_4[i]*rho*g*a[i]*B;
                M_t_4 [i] := 0.5*del_h_und_4[i]*rho*g*a[i]*B*100;
                   Z_h_4[i] := F_z_4 [i]/ (rho*g*Aw):
                       Z_t_4[i] := M_t_4[i]/Mt:
                          Z_tot_4 [i] := Z_h_4[i]+ Z_t_4[i]:

  end do:
```



```
ss := [seq([hh[i],0.2],i=0..40)]:
ss1 := [seq([hh[i],del_h_und_1 [i]],i=0..40)]:
   ss12 := [seq([hh[i],F_z_1[i]],i=0..40)]:
     ss13 := [seq([hh[i],M_t_1[i]],i=0..40)]:
        ss14 :=[seq([hh[i],Z_h_1[i]],i=0..40)]:
           ss15 := [seq([hh[i],Z_t_1[i]],i=0..40)]:
             ss16 := [seq([hh[i],Z_tot_1[i]],i=0..40)]:

ss2 := [seq([hh[i],del_h_und_2 [i]],i=0..40)]:
    ss22 := [seq([hh[i],F_z_2[i]],i=0..40)]:
     ss23 := [seq([hh[i],M_t_2[i]],i=0..40)]:
        ss24 :=[seq([hh[i],Z_h_2[i]],i=0..40)]:
           ss25 := [seq([hh[i],Z_t_2[i]],i=0..40)]:
             ss26 := [seq([hh[i],Z_tot_2[i]],i=0..40)]:


ss3 := [seq([hh[i],del_h_und_3 [i]],i=0..40)]:
    ss32 := [seq([hh[i],F_z_3[i]],i=0..40)]:
      ss33 := [seq([hh[i],M_t_3[i]],i=0..40)]:
        ss34 :=[seq([hh[i],Z_h_3[i]],i=0..40)]:
           ss35 := [seq([hh[i],Z_t_3[i]],i=0..40)]:
             ss36 := [seq([hh[i],Z_tot_3[i]],i=0..40)]:


ss4 := [seq([hh[i],del_h_und_4 [i]],i=0..40)]:
    ss42 := [seq([hh[i],F_z_4[i]],i=0..40)]:
      ss43 := [seq([hh[i],M_t_4[i]],i=0..40)]:
       ss44 :=[seq([hh[i],Z_h_4[i]],i=0..40)]:
           ss45 := [seq([hh[i],Z_t_4[i]],i=0..40)]:
             ss46 := [seq([hh[i],Z_tot_4[i]],i=0..40)]:


# ***************************************************************************
*****************************************************************************
****************

P := plot(ss,style=line,labels=["water depth h","[m]"], color=blue,title=
"freeboard as function of water depth h", labeldirections=[horizontal,
vertical],font=[Verdana,1,10,blue],labelfont=[Calibri,1,12], axis[2] =
[gridlines = [linestyle = dot]]):

P1 := plot(ss1,style=line,labels=["water depth h","Δh_max [m]"], color=blue,
title="Pressure drop as function of water depth h", labeldirections=
[horizontal,vertical],font=[Verdana,1,10,blue],labelfont=[Calibri,1,12],
axis[2] = [gridlines = [linestyle = dot]]):

P12 := plot(ss12,style=line,labels=["water depth h","F_vertical [kN]"],
color=purple,title="Vertical force as function of water depth h",
labeldirections=[horizontal,vertical],font=[Verdana,1,10,blue],labelfont=
[Calibri,1,12],axis[2] = [gridlines = [linestyle = dot]]):

P13 := plot(ss13,style=line,labels=["water depth h","M_trim [kNm]"], color=
orange,title="Trim moment as function of water depth h", labeldirections=
[horizontal,vertical],font=[Verdana,1,10,blue],labelfont=[Calibri,1,12],axis
[2] = [gridlines = [linestyle = dot]]):

P14 := plot(ss14,style=line,labels=["water depth h","Z_h [cm]"], color=
green,title="sinkage due to vertical force as function of water depth h",
labeldirections=[horizontal,vertical],font=[Verdana,1,10,blue],labelfont=
[Calibri,1,12],axis[2] = [gridlines = [linestyle = dot]]):

P15 := plot(ss15,style=line,labels=["water depth h","Z_t [cm]"], color=red,
title="sinkage due to trim moment as function of water depth h",
labeldirections=[horizontal,vertical],font=[Verdana,1,10,blue],labelfont=
[Calibri,1,12],axis[2] = [gridlines = [linestyle = dot]]):
```



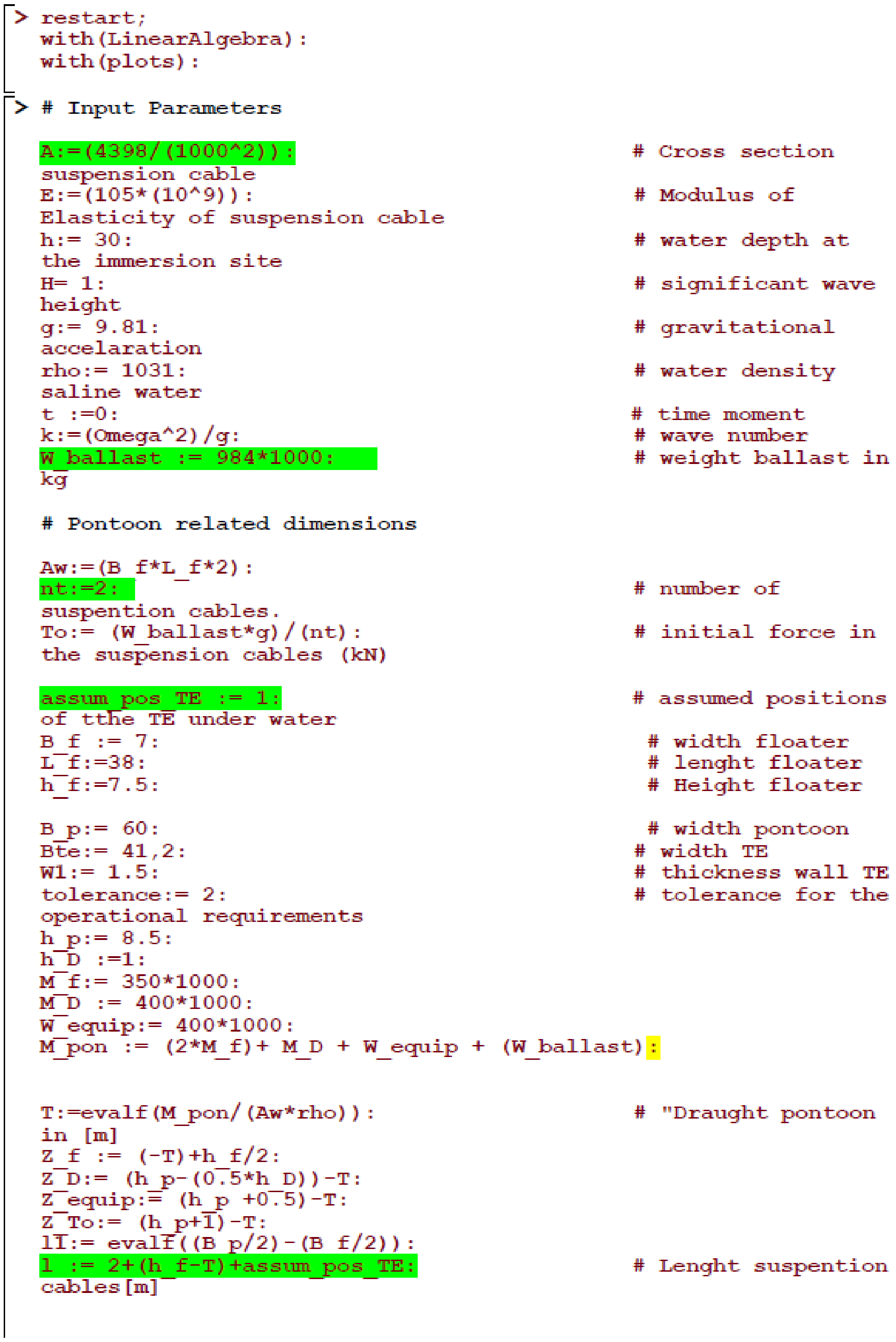
```
P16 := plot(ss16,style=line,labels=["water depth h","Z_total [cm]"], color=
black,title="Z_total as function of water depth (V=0.5 m/s)  ",
labeldirections=[horizontal,vertical],font=[Verdana,1,10,blue],labelfont=
[Calibri,1,12],axis[2] = [gridlines = [linestyle = dot]]):


#***********************************************************************
************************************************************************
********************

P2 := plot(ss2,style=line,labels=["V_tow","Δh_max [m]"], color=blue,title=
"Pressure drop as function of V_tow", labeldirections=[horizontal,vertical],
font=[Verdana,1,10,blue],labelfont=[Calibri,1,12], axis[2] = [gridlines =
[linestyle = dot]]):

P22 := plot(ss22,style=line,labels=["V_tow","F_vertical [kN]"], color=
purple,title="Vertical force as function of V_tow", labeldirections=
[horizontal,vertical],font=[Verdana,1,10,blue],labelfont=[Calibri,1,12],axis
[2] = [gridlines = [linestyle = dot]]):

P23 := plot(ss23,style=line,labels=["V_tow","M_trim [kNm]"], color=orange,
title="Trim moment as function of V_tow", labeldirections=[horizontal,
vertical],font=[Verdana,1,10,blue],labelfont=[Calibri,1,12],axis[2] =
[gridlines = [linestyle = dot]]):

P24 := plot(ss24,style=line,labels=["water depth h","Z_h [cm]"], color=
green,title="sinkage due to vertical force as function of water depth h",
labeldirections=[horizontal,vertical],font=[Verdana,1,10,blue],labelfont=
[Calibri,1,12],axis[2] = [gridlines = [linestyle = dot]]):

P25 := plot(ss25,style=line,labels=["water depth h","Z_t [cm]"], color=red,
title="sinkage due to trim moment as function of water depth h",
labeldirections=[horizontal,vertical],font=[Verdana,1,10,blue],labelfont=
[Calibri,1,12],axis[2] = [gridlines = [linestyle = dot]]):

P26 := plot(ss26,style=line,labels=["water depth h","Z_tot_2 [cm]"], color=
purple,title="Z_total as function of water depth (V=1 m/s) ",
labeldirections=[horizontal,vertical],font=[Verdana,1,10,blue],labelfont=
[Calibri,1,12],axis[2] = [gridlines = [linestyle = dot]]):


#***********************************************************************
************************************************************************
********************

P3 := plot(ss3,style=line,labels=["V_tow","Δh_max [m]"], color=blue,title=
"Pressure drop as function of V_tow", labeldirections=[horizontal,vertical],
font=[Verdana,1,10,blue],labelfont=[Calibri,1,12], axis[2] = [gridlines =
[linestyle = dot]]):

P32 := plot(ss32,style=line,labels=["V_tow","F_vertical [kN]"], color=
purple,title="Vertical force as function of V_tow", labeldirections=
[horizontal,vertical],font=[Verdana,1,10,blue],labelfont=[Calibri,1,12],axis
[2] = [gridlines = [linestyle = dot]]):

P33 := plot(ss33,style=line,labels=["V_tow","M_trim [kNm]"], color=orange,
title="Trim moment as function of V_tow", labeldirections=[horizontal,
vertical],font=[Verdana,1,10,blue],labelfont=[Calibri,1,12],axis[2] =
[gridlines = [linestyle = dot]]):

P34 := plot(ss34,style=line,labels=["water depth h","Z_h [cm]"], color=
green,title="sinkage due to vertical force as function of water depth h",
labeldirections=[horizontal,vertical],font=[Verdana,1,10,blue],labelfont=
[Calibri,1,12],axis[2] = [gridlines = [linestyle = dot]]):

P35 := plot(ss35,style=line,labels=["water depth h","Z_t [cm]"], color=red,
```



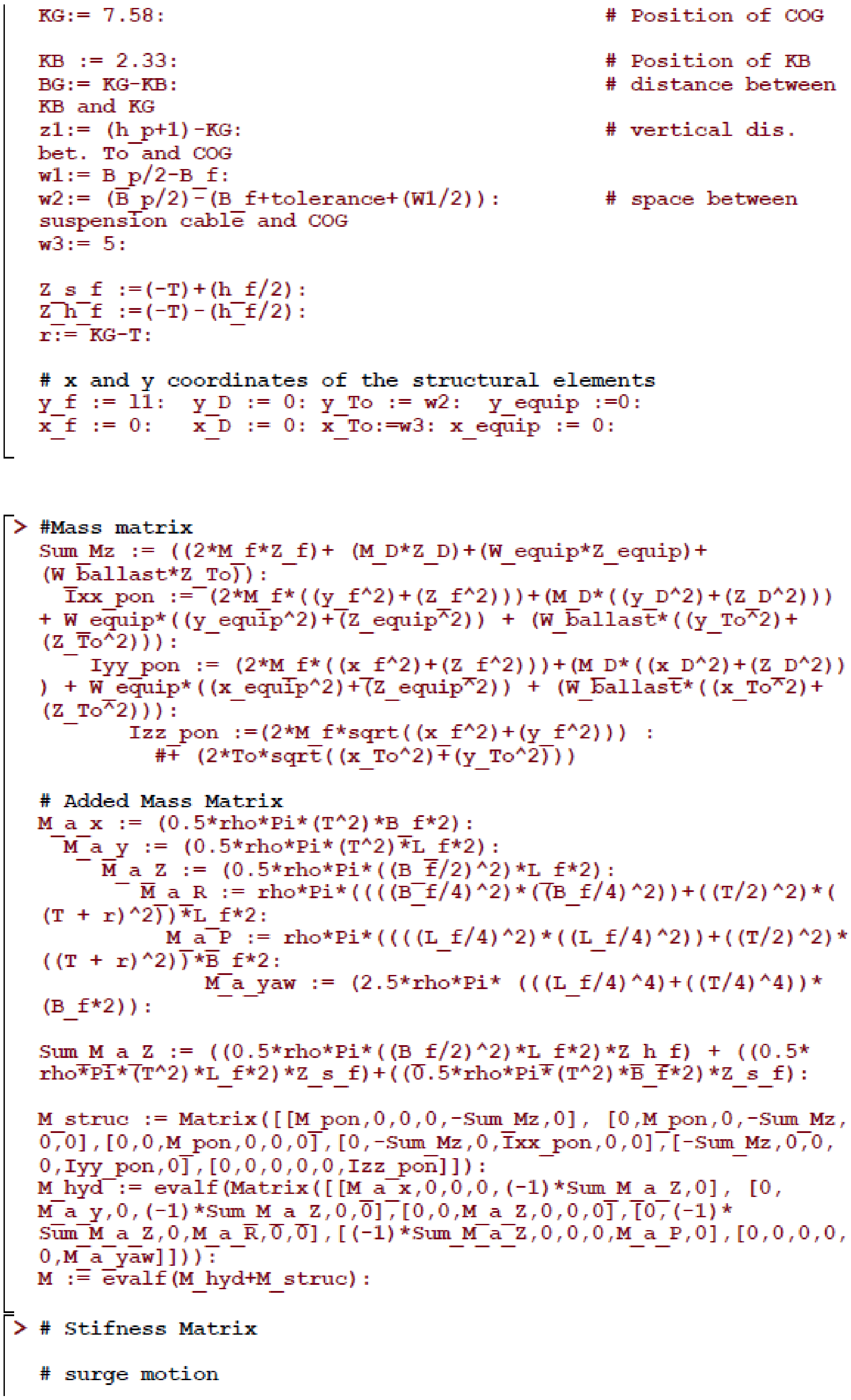
```
title="sinkage due to trim moment as function of water depth h",
labeldirections=[horizontal,vertical],font=[Verdana,1,10,blue],labelfont=
[Calibri,1,12],axis[2] = [gridlines = [linestyle = dot]]):


P36 := plot(ss36,style=line,labels=["water depth h","Z_tot_3 [cm]"], color=
blue,title="Z_total as function of water depth (V=1.5 m/s) ",
labeldirections=[horizontal,vertical],font=[Verdana,1,10,blue],labelfont=
[Calibri,1,12],axis[2] = [gridlines = [linestyle = dot]]):

#**************************************************************************
**************************************************************************
********************

P4 := plot(ss4,style=line,labels=["V_tow","Δh_max [m]"], color=blue,title=
"Pressure drop as function of V_tow", labeldirections=[horizontal,vertical],
font=[Verdana,1,10,blue],labelfont=[Calibri,1,12], axis[2] = [gridlines =
[linestyle = dot]]):

P42 := plot(ss42,style=line,labels=["V_tow","F_vertical [kN]"], color=
purple,title="Vertical force as function of V_tow", labeldirections=
[horizontal,vertical],font=[Verdana,1,10,blue],labelfont=[Calibri,1,12],axis
[2] = [gridlines = [linestyle = dot]]):

P43 := plot(ss43,style=line,labels=["V_tow","M_trim [kNm]"], color=orange,
title="Trim moment as function of V_tow", labeldirections=[horizontal,
vertical],font=[Verdana,1,10,blue],labelfont=[Calibri,1,12],axis[2] =
[gridlines = [linestyle = dot]]):

P44 := plot(ss44,style=line,labels=["water depth h","Z_h [cm]"], color=
green,title="sinkage due to vertical force as function of water depth h",
labeldirections=[horizontal,vertical],font=[Verdana,1,10,blue],labelfont=
[Calibri,1,12],axis[2] = [gridlines = [linestyle = dot]]):

P45 := plot(ss45,style=line,labels=["water depth h","Z_t [cm]"], color=red,
title="sinkage due to trim moment as function of water depth h",
labeldirections=[horizontal,vertical],font=[Verdana,1,10,blue],labelfont=
[Calibri,1,12],axis[2] = [gridlines = [linestyle = dot]]):

P46 := plot(ss46,style=line,labels=["water depth h","Z_tot_4 [cm]"], color=
black,title="Z_total as function of water depth (V=2.0 m/s) ",
labeldirections=[horizontal,vertical],font=[Verdana,1,10,blue],labelfont=
[Calibri,1,12],axis[2] = [gridlines = [linestyle = dot]]):


#display({P1,P2,P3,P4});  display({P12,P22,P32,P42}); display({P13,P23,P33,
P43});
> #display({P14,P15,P});  display({P24,P25,P}); display({P34,P35,P}); display(
{P44,P45,P});


> display({P14,P15,P16,P},title="Z_total as function of water depth (V=0.5
m/s)");
display({P24,P25,P26,P}, title="Z_total as function of water depth (V=1.0
m/s) ");
display({P34,P35,P36,P},title="Z_total as function of water depth (V=1.5
m/s) ");
display({P44,P45,P46,P}, title="Z_total as function of water depth (V=2.0
m/s) ");
```

7.4 CALCULATION FILES NATURAL PERIODS OF PONTOONS

Natural Periods Catamaran



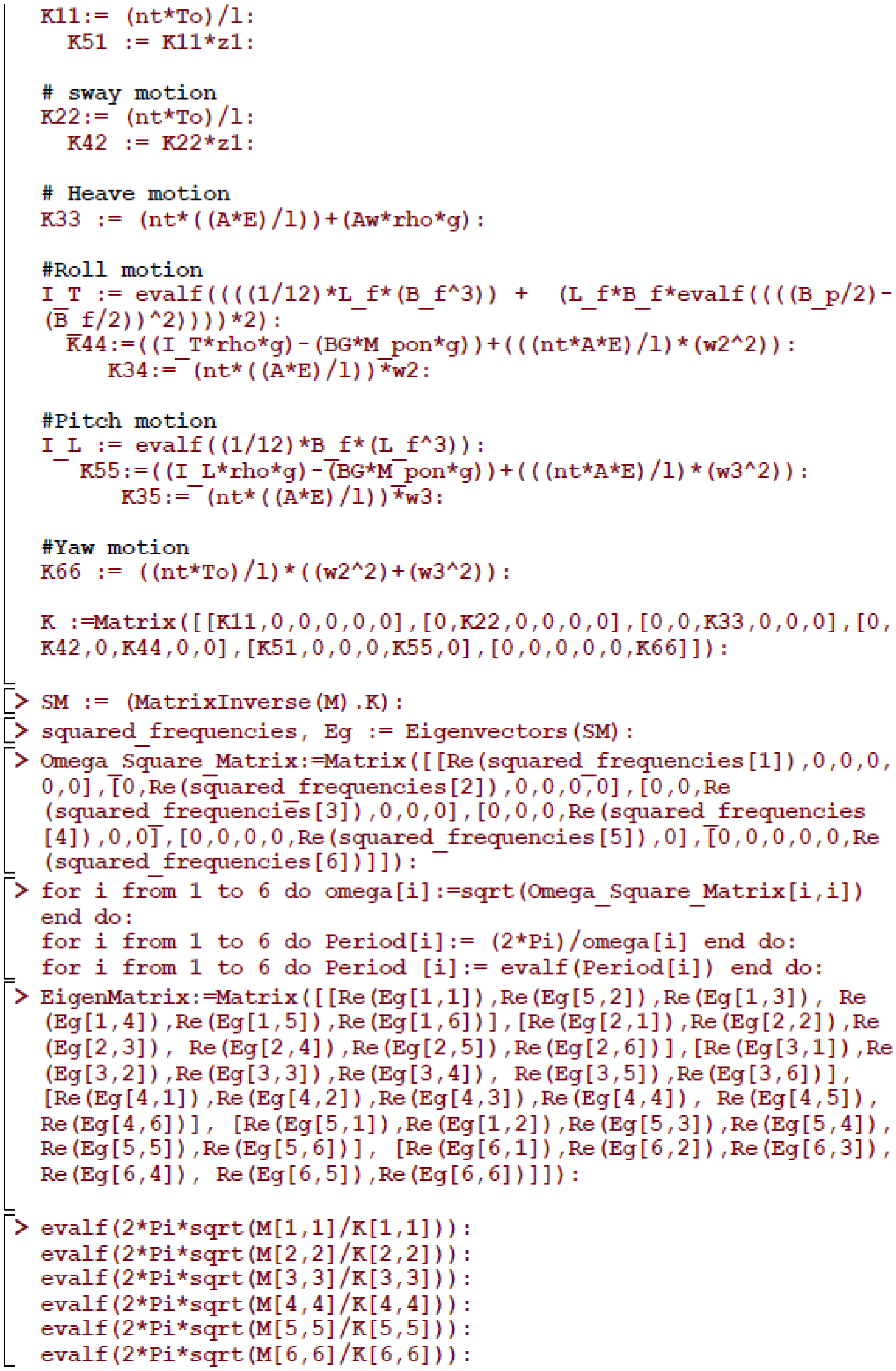
```
> restart;
  with(LinearAlgebra):
  with(plots):

> # Input Parameters

  A:=(4398/(1000^2)):                        # Cross section
  suspension cable
  E:=(105*(10^9)):                           # Modulus of
  Elasticity of suspension cable
  h:= 30:                                    # water depth at
  the immersion site
  H= 1:                                      # significant wave
  height
  g:= 9.81:                                  # gravitational
  accelaration
  rho:= 1031:                                # water density
  saline water
  t :=0:                                     # time moment
  k:=(Omega^2)/g:                            # wave number
  W_ballast := 984*1000:                     # weight ballast in
  kg

  # Pontoon related dimensions

  Aw:=(B_f*L_f*2):
  nt:=2:                                     # number of
  suspention cables.
  To:= (W_ballast*g)/(nt):                   # initial force in
  the suspension cables (kN)

  assum_pos_TE := 1:                         # assumed positions
  of tthe TE under water
  B_f := 7:                                   # width floater
  L_f:=38:                                    # lenght floater
  h_f:=7.5:                                   # Height floater

  B_p:= 60:                                   # width pontoon
  Bte:= 41,2:                                # width TE
  W1:= 1.5:                                  # thickness wall TE
  tolerance:= 2:                             # tolerance for the
  operational requirements
  h_p:= 8.5:
  h_D :=1:
  M_f:= 350*1000:
  M_D := 400*1000:
  W_equip:= 400*1000:
  M_pon := (2*M_f)+ M_D + W_equip + (W_ballast):


  T:=evalf(M_pon/(Aw*rho)):                  # "Draught pontoon
  in [m]
  Z_f := (-T)+h_f/2:
  Z_D:= (h_p-(0.5*h_D))-T:
  Z_equip:= (h_p +0.5)-T:
  Z_To:= (h_p+1)-T:
  l1:= evalf((B_p/2)-(B_f/2)):
  l := 2+(h_f-T)+assum_pos_TE:               # Lenght suspention
  cables[m]
```



```
KG:= 7.58:                                  # Position of COG

KB := 2.33:                                 # Position of KB
BG:= KG-KB:                                 # distance between
KB and KG
z1:= (h_p+1)-KG:                            # vertical dis.
bet. To and COG
w1:= B_p/2-B_f:
w2:= (B_p/2)-(B_f+tolerance+(W1/2)):        # space between
suspension cable and COG
w3:= 5:

Z_s_f :=(-T)+(h_f/2):
Z_h_f :=(-T)-(h_f/2):
r:= KG-T:

# x and y coordinates of the structural elements
y_f := l1:   y_D := 0: y_To := w2:  y_equip :=0:
x_f := 0:    x_D := 0: x_To:=w3: x_equip := 0:
```

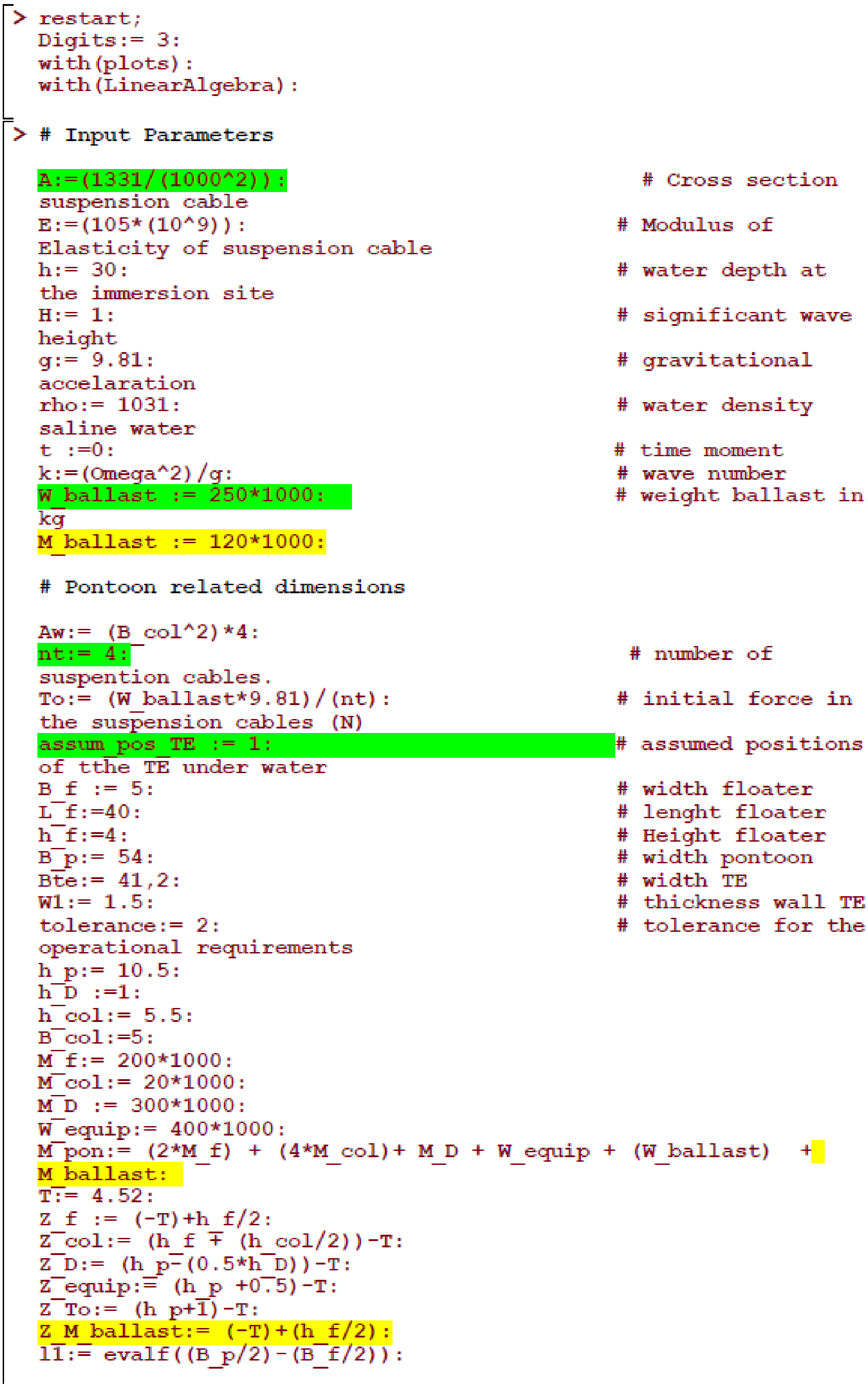
```
> #Mass matrix
Sum_Mz := ((2*M_f*Z_f)+ (M_D*Z_D)+(W_equip*Z_equip)+
(W_ballast*Z_To)):
  Ixx_pon := (2*M_f*((y_f^2)+(Z_f^2)))+(M_D*((y_D^2)+(Z_D^2)))
+ W_equip*((y_equip^2)+(Z_equip^2)) + (W_ballast*((y_To^2)+
(Z_To^2))):
    Iyy_pon := (2*M_f*((x_f^2)+(Z_f^2)))+(M_D*((x_D^2)+(Z_D^2))
) + W_equip*((x_equip^2)+(Z_equip^2)) + (W_ballast*((x_To^2)+
(Z_To^2))):
       Izz_pon :=(2*M_f*sqrt((x_f^2)+(y_f^2))) :
         #+ (2*To*sqrt((x_To^2)+(y_To^2)))

# Added Mass Matrix
M_a_x := (0.5*rho*Pi*(T^2)*B_f*2):
  M_a_y := (0.5*rho*Pi*(T^2)*L_f*2):
     M_a_Z := (0.5*rho*Pi*((B_f/2)^2)*L_f*2):
        M_a_R := rho*Pi*((((B_f/4)^2)*((B_f/4)^2))+((T/2)^2)*(
(T + r)^2))*L_f*2:
           M_a_P := rho*Pi*((((L_f/4)^2)*((L_f/4)^2))+((T/2)^2)*
((T + r)^2))*B_f*2:
              M_a_yaw := (2.5*rho*Pi* (((L_f/4)^4)+((T/4)^4))*
(B_f*2)):

Sum_M_a_Z := ((0.5*rho*Pi*((B_f/2)^2)*L_f*2)*Z_h_f) + ((0.5*
rho*Pi*(T^2)*L_f*2)*Z_s_f)+((0.5*rho*Pi*(T^2)*B_f*2)*Z_s_f):

M_struc := Matrix([[M_pon,0,0,0,-Sum_Mz,0], [0,M_pon,0,-Sum_Mz,
0,0],[0,0,M_pon,0,0,0],[0,-Sum_Mz,0,Ixx_pon,0,0],[-Sum_Mz,0,0,
0,Iyy_pon,0],[0,0,0,0,0,Izz_pon]]):
M_hyd := evalf(Matrix([[M_a_x,0,0,0,(-1)*Sum_M_a_Z,0], [0,
M_a_y,0,(-1)*Sum_M_a_Z,0,0],[0,0,M_a_Z,0,0,0],[0,(-1)*
Sum_M_a_Z,0,M_a_R,0,0],[(-1)*Sum_M_a_Z,0,0,0,M_a_P,0],[0,0,0,0,
0,M_a_yaw]])):
M := evalf(M_hyd+M_struc):
```

```
> # Stifness Matrix

  # surge motion
```

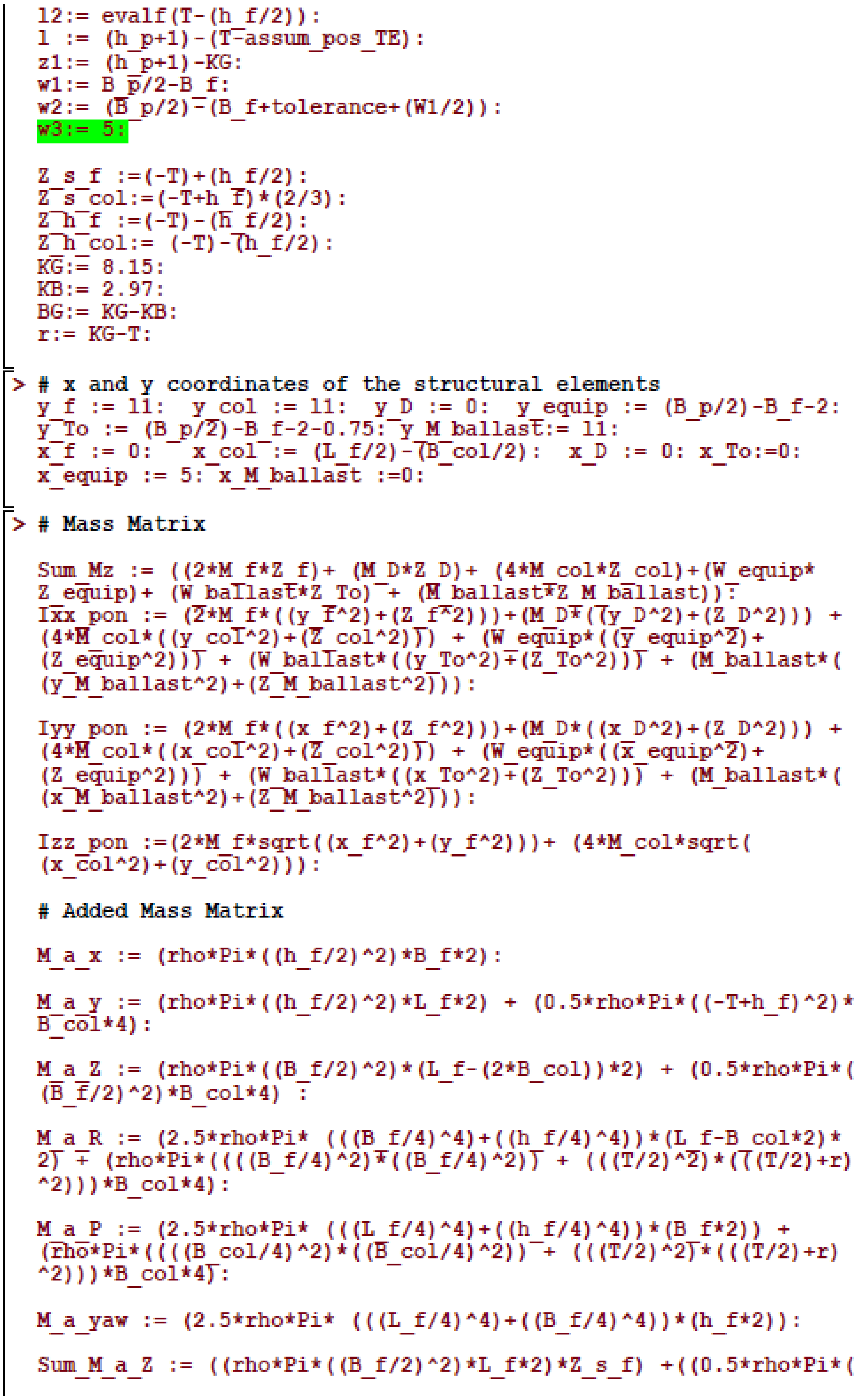


```
K11:= (nt*To)/l:
  K51 := K11*z1:

# sway motion
K22:= (nt*To)/l:
  K42 := K22*z1:

# Heave motion
K33 := (nt*((A*E)/l))+(Aw*rho*g):

#Roll motion
I_T := evalf((((1/12)*L_f*(B_f^3)) +  (L_f*B_f*evalf((((B_p/2)-
(B_f/2))^2))))*2):
  K44:=((I_T*rho*g)-(BG*M_pon*g))+(((nt*A*E)/l)*(w2^2)):
     K34:= (nt*((A*E)/l))*w2:

#Pitch motion
I_L := evalf((1/12)*B_f*(L_f^3)):
   K55:=((I_L*rho*g)-(BG*M_pon*g))+(((nt*A*E)/l)*(w3^2)):
      K35:= (nt*((A*E)/l))*w3:

#Yaw motion
K66 := ((nt*To)/l)*((w2^2)+(w3^2)):

K :=Matrix([[K11,0,0,0,0,0],[0,K22,0,0,0,0],[0,0,K33,0,0,0],[0,
K42,0,K44,0,0],[K51,0,0,0,K55,0],[0,0,0,0,0,K66]]):

> SM := (MatrixInverse(M).K):
> squared_frequencies, Eg := Eigenvectors(SM):
> Omega_Square_Matrix:=Matrix([[Re(squared_frequencies[1]),0,0,0,
0,0],[0,Re(squared_frequencies[2]),0,0,0,0],[0,0,Re
(squared_frequencies[3]),0,0,0],[0,0,0,Re(squared_frequencies
[4]),0,0],[0,0,0,0,Re(squared_frequencies[5]),0],[0,0,0,0,0,Re
(squared_frequencies[6])]]):
> for i from 1 to 6 do omega[i]:=sqrt(Omega_Square_Matrix[i,i])
end do:
for i from 1 to 6 do Period[i]:= (2*Pi)/omega[i] end do:
for i from 1 to 6 do Period [i]:= evalf(Period[i]) end do:
> EigenMatrix:=Matrix([[Re(Eg[1,1]),Re(Eg[5,2]),Re(Eg[1,3]), Re
(Eg[1,4]),Re(Eg[1,5]),Re(Eg[1,6])],[Re(Eg[2,1]),Re(Eg[2,2]),Re
(Eg[2,3]), Re(Eg[2,4]),Re(Eg[2,5]),Re(Eg[2,6])],[Re(Eg[3,1]),Re
(Eg[3,2]),Re(Eg[3,3]),Re(Eg[3,4]), Re(Eg[3,5]),Re(Eg[3,6])],
[Re(Eg[4,1]),Re(Eg[4,2]),Re(Eg[4,3]),Re(Eg[4,4]), Re(Eg[4,5]),
Re(Eg[4,6])], [Re(Eg[5,1]),Re(Eg[1,2]),Re(Eg[5,3]),Re(Eg[5,4]),
Re(Eg[5,5]),Re(Eg[5,6])], [Re(Eg[6,1]),Re(Eg[6,2]),Re(Eg[6,3]),
Re(Eg[6,4]), Re(Eg[6,5]),Re(Eg[6,6])]]):

> evalf(2*Pi*sqrt(M[1,1]/K[1,1])):
evalf(2*Pi*sqrt(M[2,2]/K[2,2])):
evalf(2*Pi*sqrt(M[3,3]/K[3,3])):
evalf(2*Pi*sqrt(M[4,4]/K[4,4])):
evalf(2*Pi*sqrt(M[5,5]/K[5,5])):
evalf(2*Pi*sqrt(M[6,6]/K[6,6])):
```

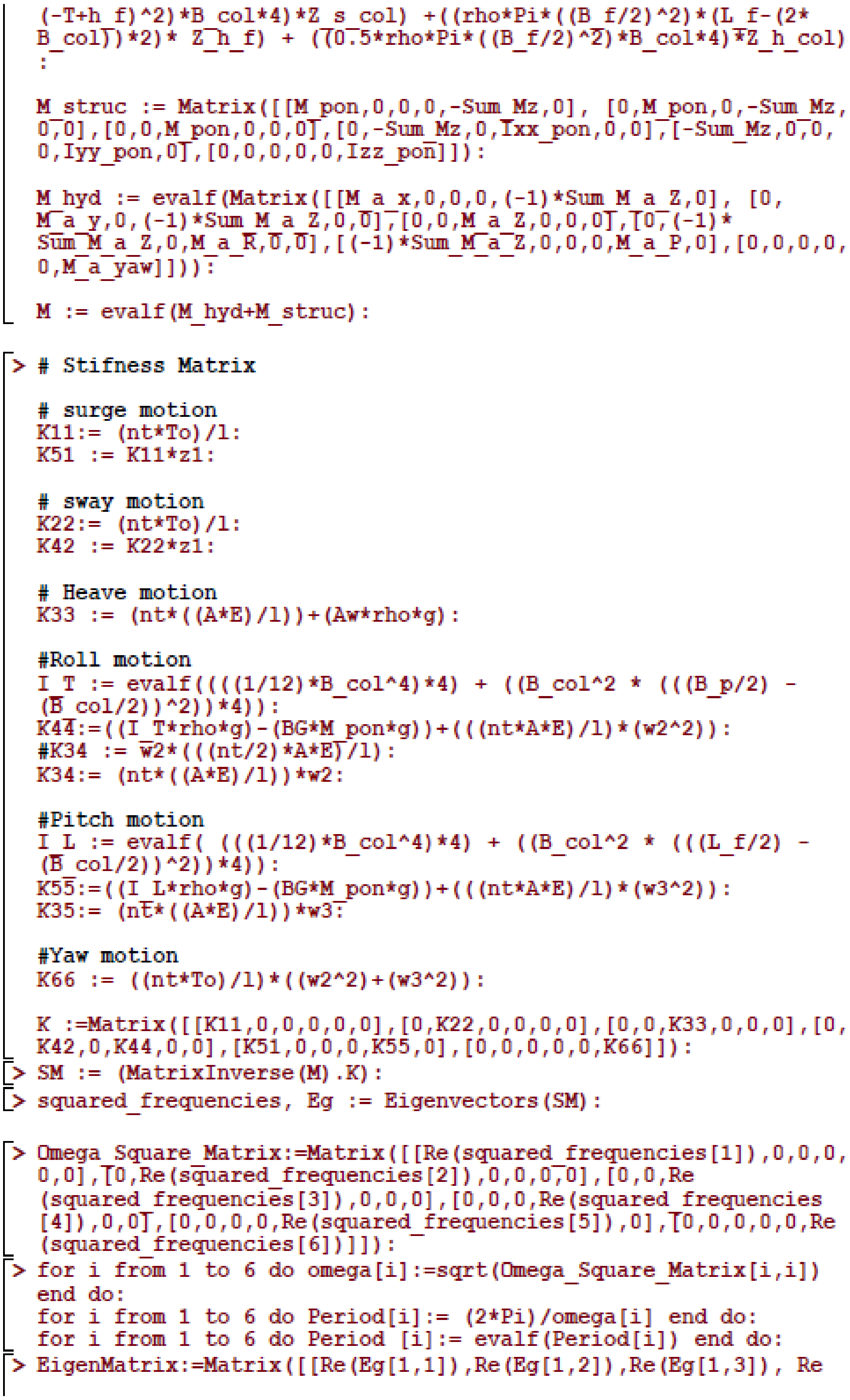
Calculation files Natural periods Semi-submersible pontoon



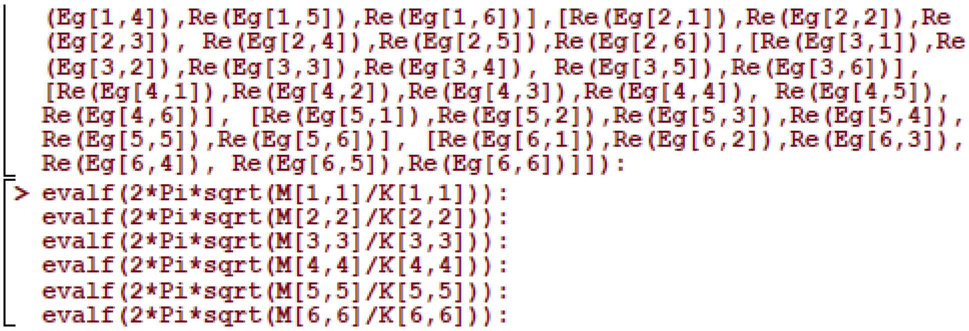
```
> restart;
  Digits:= 3:
  with(plots):
  with(LinearAlgebra):

> # Input Parameters

  A:=(1331/(1000^2)):                          # Cross section
  suspension cable
  E:=(105*(10^9)):                           # Modulus of
  Elasticity of suspension cable
  h:= 30:                                    # water depth at
  the immersion site
  H:= 1:                                     # significant wave
  height
  g:= 9.81:                                  # gravitational
  accelaration
  rho:= 1031:                                # water density
  saline water
  t :=0:                                     # time moment
  k:=(Omega^2)/g:                            # wave number
  W_ballast := 250*1000:                     # weight ballast in
  kg
  M_ballast := 120*1000:

  # Pontoon related dimensions

  Aw:= (B_col^2)*4:
  nt:= 4:                                     # number of
  suspention cables.
  To:= (W_ballast*9.81)/(nt):                # initial force in
  the suspension cables (N)
  assum_pos_TE := 1:                         # assumed positions
  of tthe TE under water
  B_f := 5:                                  # width floater
  L_f:=40:                                   # lenght floater
  h_f:=4:                                    # Height floater
  B_p:= 54:                                  # width pontoon
  Bte:= 41,2:                                # width TE
  W1:= 1.5:                                  # thickness wall TE
  tolerance:= 2:                             # tolerance for the
  operational requirements
  h_p:= 10.5:
  h_D :=1:
  h_col:= 5.5:
  B_col:=5:
  M_f:= 200*1000:
  M_col:= 20*1000:
  M_D := 300*1000:
  W_equip:= 400*1000:
  M_pon:= (2*M_f) + (4*M_col)+ M_D + W_equip + (W_ballast)   +
  M_ballast:
  T:= 4.52:
  Z_f := (-T)+h_f/2:
  Z_col:= (h_f + (h_col/2))-T:
  Z_D:= (h_p-(0.5*h_D))-T:
  Z_equip:= (h_p +0.5)-T:
  Z_To:= (h_p+1)-T:
  Z_M_ballast:= (-T)+(h_f/2):
  l1:= evalf((B_p/2)-(B_f/2)):
```



```
  l2:= evalf(T-(h_f/2)):
  l := (h_p+1)-(T-assum_pos_TE):
  z1:= (h_p+1)-KG:
  w1:= B_p/2-B_f:
  w2:= (B_p/2)-(B_f+tolerance+(W1/2)):
  w3:= 5:

  Z_s_f :=(-T)+(h_f/2):
  Z_s_col:=(-T+h_f)*(2/3):
  Z_h_f :=(-T)-(h_f/2):
  Z_h_col:= (-T)-(h_f/2):
  KG:= 8.15:
  KB:= 2.97:
  BG:= KG-KB:
  r:= KG-T:

> # x and y coordinates of the structural elements
  y_f := l1:  y_col := l1:  y_D := 0:  y_equip := (B_p/2)-B_f-2:
  y_To := (B_p/2)-B_f-2-0.75: y_M_ballast:= l1:
  x_f := 0:   x_col := (L_f/2)-(B_col/2):  x_D := 0: x_To:=0:
  x_equip := 5: x_M_ballast :=0:

> # Mass Matrix

  Sum_Mz := ((2*M_f*Z_f)+ (M_D*Z_D)+ (4*M_col*Z_col)+(W_equip*
  Z_equip)+ (W_ballast*Z_To) + (M_ballast*Z_M_ballast)):
  Ixx_pon := (2*M_f*((y_f^2)+(Z_f^2)))+(M_D*((y_D^2)+(Z_D^2))) +
  (4*M_col*((y_col^2)+(Z_col^2))) + (W_equip*((y_equip^2)+
  (Z_equip^2))) + (W_ballast*((y_To^2)+(Z_To^2))) + (M_ballast*(
  (y_M_ballast^2)+(Z_M_ballast^2))):

  Iyy_pon := (2*M_f*((x_f^2)+(Z_f^2)))+(M_D*((x_D^2)+(Z_D^2))) +
  (4*M_col*((x_col^2)+(Z_col^2))) + (W_equip*((x_equip^2)+
  (Z_equip^2))) + (W_ballast*((x_To^2)+(Z_To^2))) + (M_ballast*(
  (x_M_ballast^2)+(Z_M_ballast^2))):

  Izz_pon :=(2*M_f*sqrt((x_f^2)+(y_f^2)))+ (4*M_col*sqrt(
  (x_col^2)+(y_col^2))):

  # Added Mass Matrix

  M_a_x := (rho*Pi*((h_f/2)^2)*B_f*2):

  M_a_y := (rho*Pi*((h_f/2)^2)*L_f*2) + (0.5*rho*Pi*((-T+h_f)^2)*
  B_col*4):

  M_a_Z := (rho*Pi*((B_f/2)^2)*(L_f-(2*B_col))*2) + (0.5*rho*Pi*(
  (B_f/2)^2)*B_col*4)  :

  M_a_R := (2.5*rho*Pi* (((B_f/4)^4)+((h_f/4)^4))*(L_f-B_col*2)*
  2) + (rho*Pi*((((B_f/4)^2)*((B_f/4)^2)) + (((T/2)^2)*(((T/2)+r)
  ^2)))*B_col*4):

  M_a_P := (2.5*rho*Pi* (((L_f/4)^4)+((h_f/4)^4))*(B_f*2)) +
  (rho*Pi*((((B_col/4)^2)*((B_col/4)^2)) + (((T/2)^2)*(((T/2)+r)
  ^2)))*B_col*4):

  M_a_yaw := (2.5*rho*Pi* (((L_f/4)^4)+((B_f/4)^4))*(h_f*2)):

  Sum_M_a_Z := ((rho*Pi*((B_f/2)^2)*L_f*2)*Z_s_f) +((0.5*rho*Pi*(
```



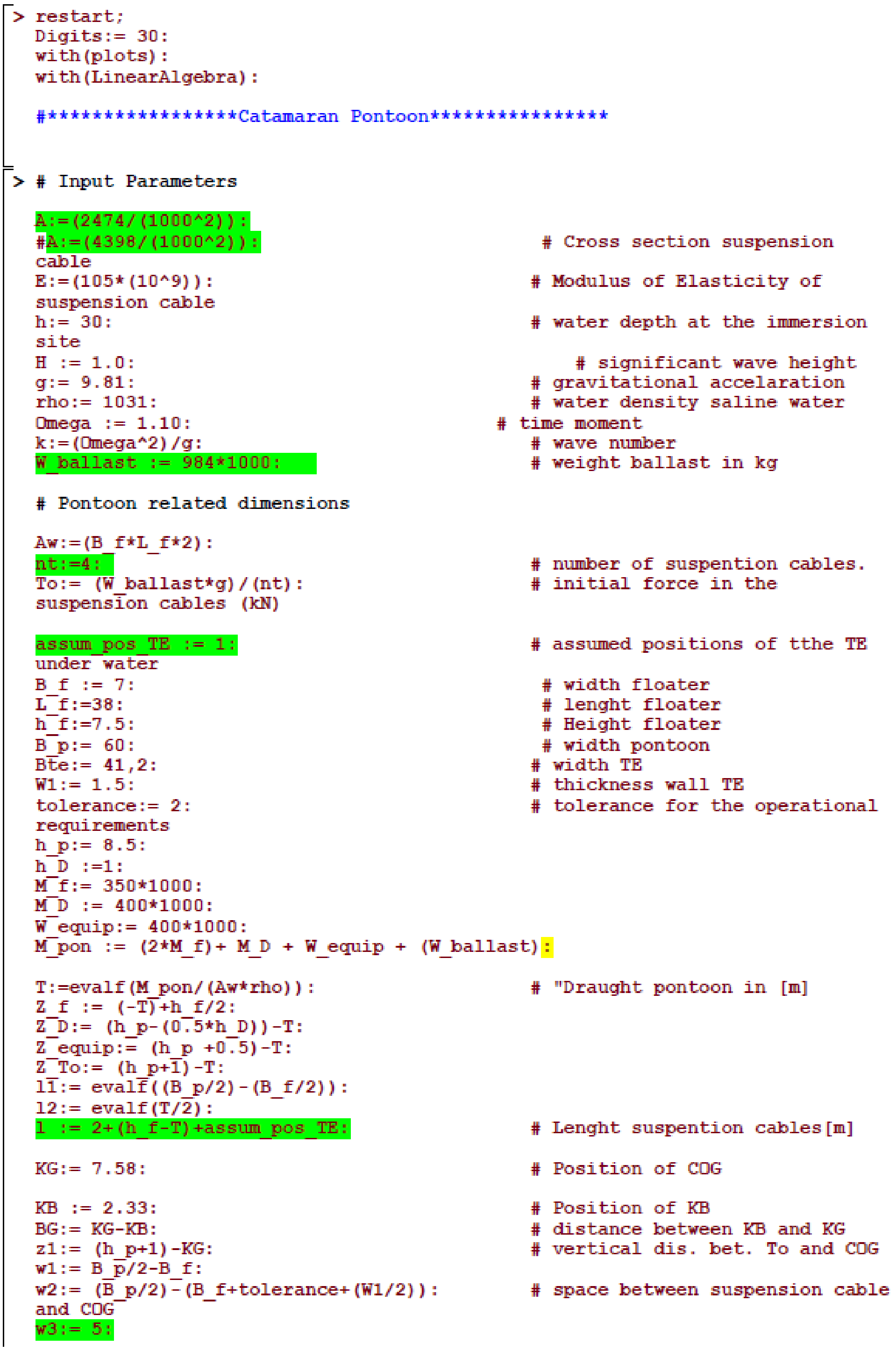
```
(-T+h_f)^2)*B_col*4)*Z_s_col) +((rho*Pi*((B_f/2)^2)*(L_f-(2*
B_col))*2)* Z_h_f) + ((0.5*rho*Pi*((B_f/2)^2)*B_col*4)*Z_h_col)
:

M_struc := Matrix([[M_pon,0,0,0,-Sum_Mz,0], [0,M_pon,0,-Sum_Mz,
0,0],[0,0,M_pon,0,0,0],[0,-Sum_Mz,0,Ixx_pon,0,0],[-Sum_Mz,0,0,
0,Iyy_pon,0],[0,0,0,0,0,Izz_pon]]):

M_hyd := evalf(Matrix([[M_a_x,0,0,0,(-1)*Sum_M_a_Z,0], [0,
M_a_y,0,(-1)*Sum_M_a_Z,0,0],[0,0,M_a_Z,0,0,0],[0,(-1)*
Sum_M_a_Z,0,M_a_R,0,0],[(-1)*Sum_M_a_Z,0,0,0,M_a_P,0],[0,0,0,0,
0,M_a_yaw]])):

M := evalf(M_hyd+M_struc):

> # Stifness Matrix

  # surge motion
  K11:= (nt*To)/l:
  K51 := K11*z1:

  # sway motion
  K22:= (nt*To)/l:
  K42 := K22*z1:

  # Heave motion
  K33 := (nt*((A*E)/l))+(Aw*rho*g):

  #Roll motion
  I_T := evalf((((1/12)*B_col^4)*4) + ((B_col^2 * (((B_p/2) -
  (B_col/2))^2))*4)):
  K44:=((I_T*rho*g)-(BG*M_pon*g))+(((nt*A*E)/l)*(w2^2)):
  #K34 := w2*(((nt/2)*A*E)/l):
  K34:= (nt*((A*E)/l))*w2:

  #Pitch motion
  I_L := evalf( (((1/12)*B_col^4)*4) + ((B_col^2 * (((L_f/2) -
  (B_col/2))^2))*4)):
  K55:=((I_L*rho*g)-(BG*M_pon*g))+(((nt*A*E)/l)*(w3^2)):
  K35:= (nt*((A*E)/l))*w3:

  #Yaw motion
  K66 := ((nt*To)/l)*((w2^2)+(w3^2)):

  K :=Matrix([[K11,0,0,0,0,0],[0,K22,0,0,0,0],[0,0,K33,0,0,0],[0,
  K42,0,K44,0,0],[K51,0,0,0,K55,0],[0,0,0,0,0,K66]]):
> SM := (MatrixInverse(M).K):
> squared_frequencies, Eg := Eigenvectors(SM):

> Omega_Square_Matrix:=Matrix([[Re(squared_frequencies[1]),0,0,0,
  0,0],[0,Re(squared_frequencies[2]),0,0,0,0],[0,0,Re
  (squared_frequencies[3]),0,0,0],[0,0,0,Re(squared_frequencies
  [4]),0,0],[0,0,0,0,Re(squared_frequencies[5]),0],[0,0,0,0,0,Re
  (squared_frequencies[6])]]):
> for i from 1 to 6 do omega[i]:=sqrt(Omega_Square_Matrix[i,i])
  end do:
  for i from 1 to 6 do Period[i]:= (2*Pi)/omega[i] end do:
  for i from 1 to 6 do Period [i]:= evalf(Period[i]) end do:
> EigenMatrix:=Matrix([[Re(Eg[1,1]),Re(Eg[1,2]),Re(Eg[1,3]), Re
```



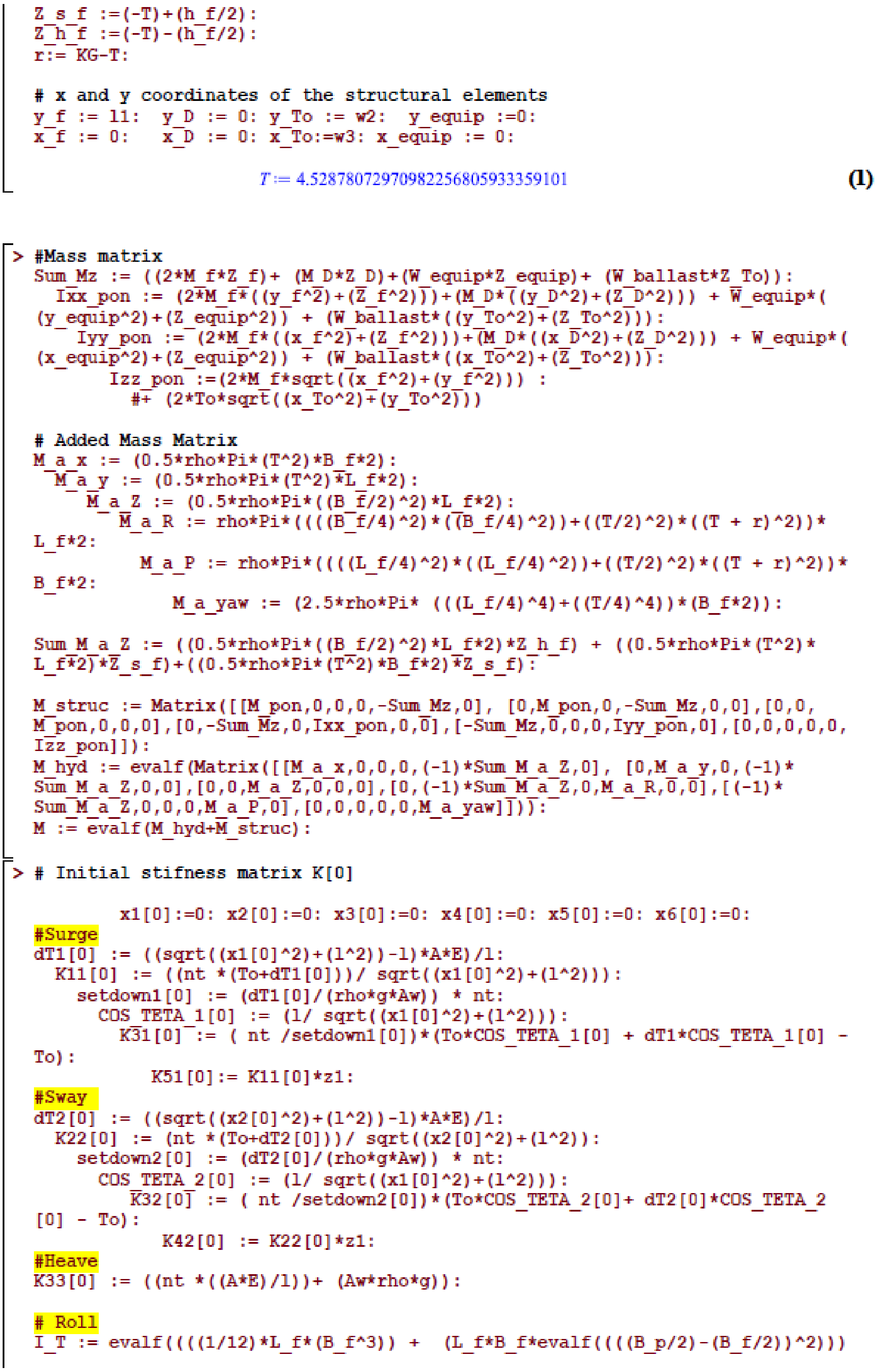
```
 (Eg[1,4]),Re(Eg[1,5]),Re(Eg[1,6])],[Re(Eg[2,1]),Re(Eg[2,2]),Re
 (Eg[2,3]), Re(Eg[2,4]),Re(Eg[2,5]),Re(Eg[2,6])],[Re(Eg[3,1]),Re
 (Eg[3,2]),Re(Eg[3,3]),Re(Eg[3,4]), Re(Eg[3,5]),Re(Eg[3,6])],
 [Re(Eg[4,1]),Re(Eg[4,2]),Re(Eg[4,3]),Re(Eg[4,4]), Re(Eg[4,5]),
 Re(Eg[4,6])], [Re(Eg[5,1]),Re(Eg[5,2]),Re(Eg[5,3]),Re(Eg[5,4]),
 Re(Eg[5,5]),Re(Eg[5,6])], [Re(Eg[6,1]),Re(Eg[6,2]),Re(Eg[6,3]),
 Re(Eg[6,4]), Re(Eg[6,5]),Re(Eg[6,6])]]):
> evalf(2*Pi*sqrt(M[1,1]/K[1,1])):
  evalf(2*Pi*sqrt(M[2,2]/K[2,2])):
  evalf(2*Pi*sqrt(M[3,3]/K[3,3])):
  evalf(2*Pi*sqrt(M[4,4]/K[4,4])):
  evalf(2*Pi*sqrt(M[5,5]/K[5,5])):
  evalf(2*Pi*sqrt(M[6,6]/K[6,6])):
```

7.5 CALCULATION FILES TIME DOMAIN ANALYSIS

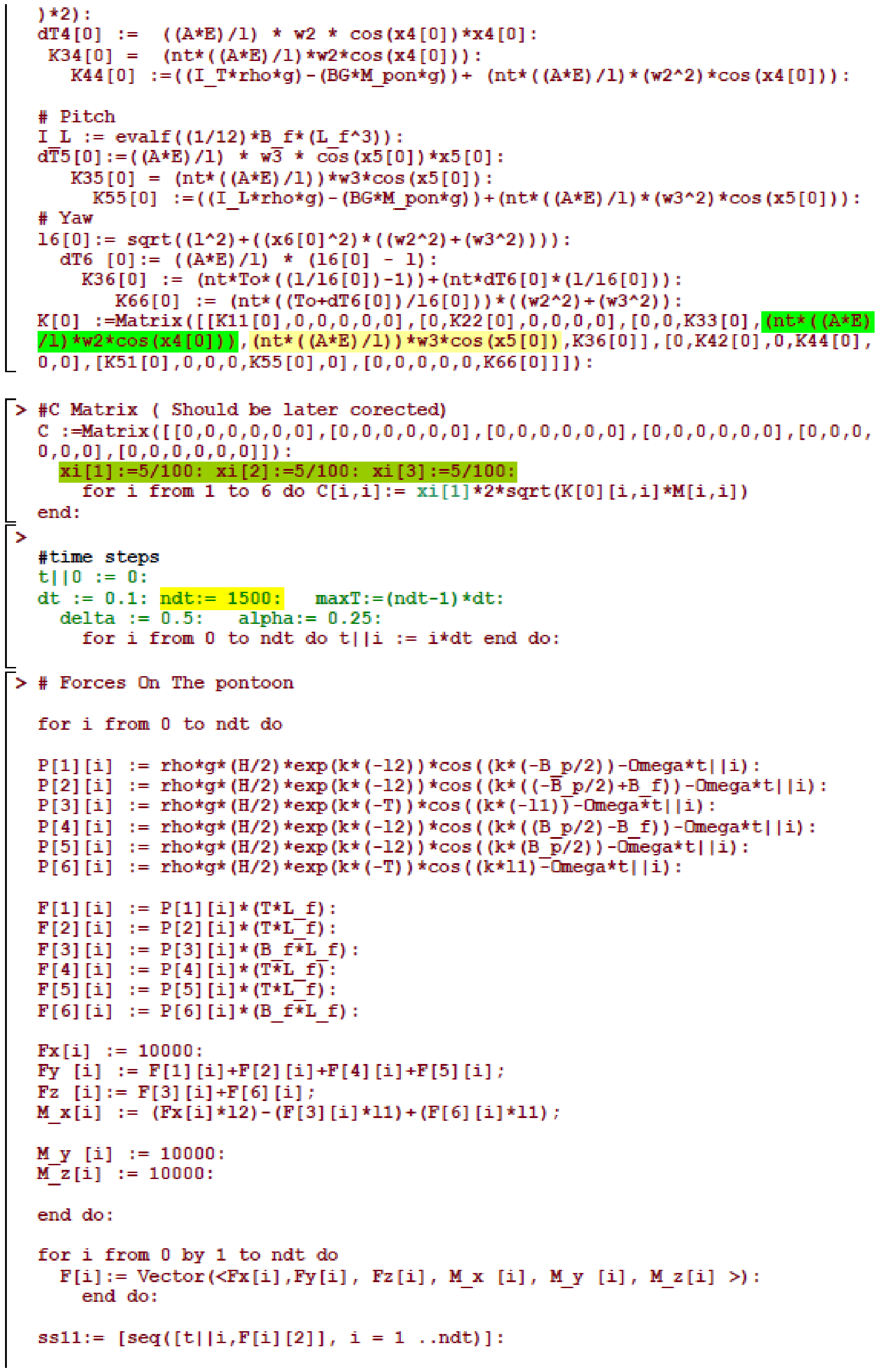
Catamaran pontoon



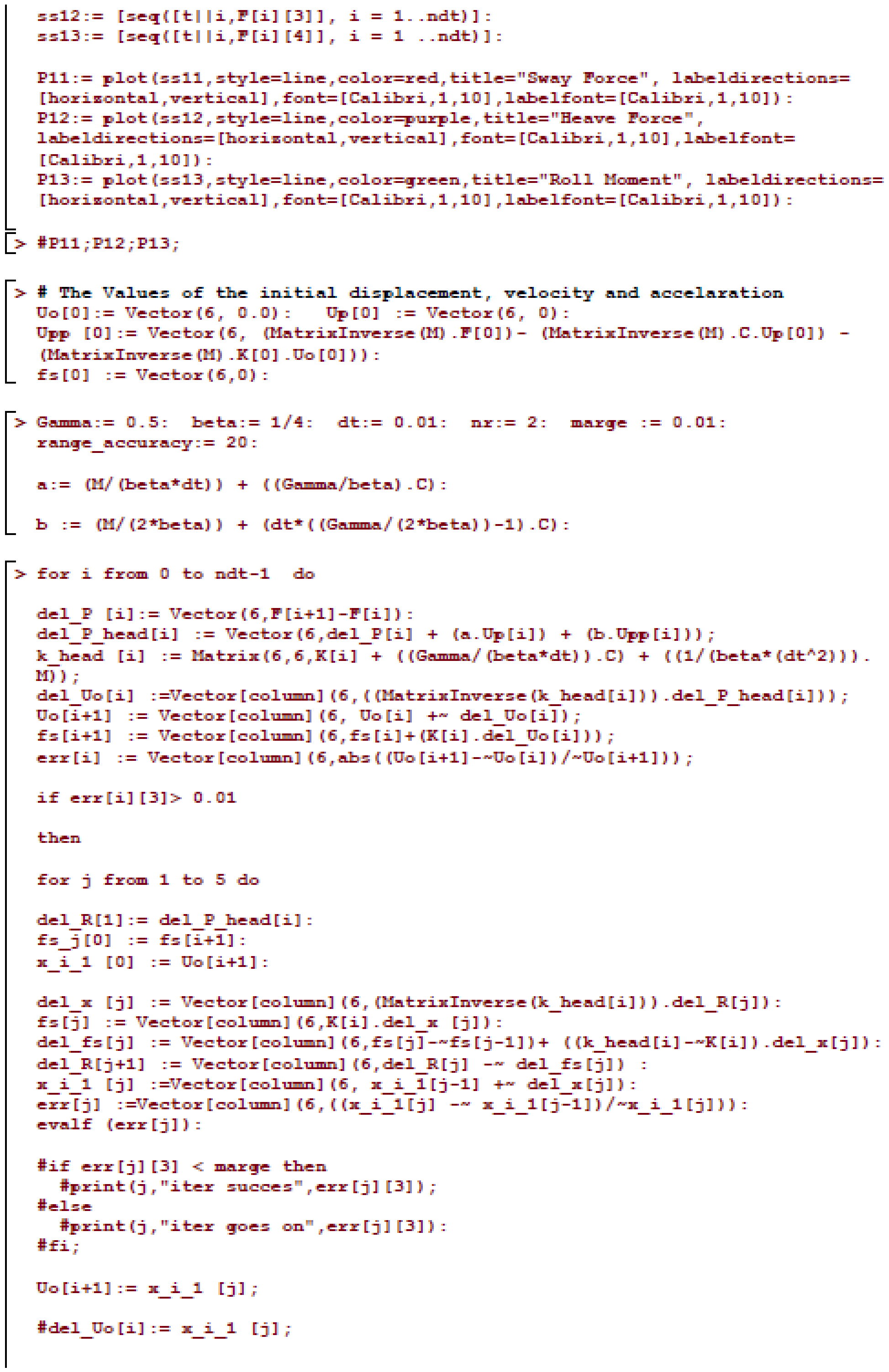
```
> restart;
  Digits:= 30:
  with(plots):
  with(LinearAlgebra):

  #*****************Catamaran Pontoon****************

> # Input Parameters

  A:=(2474/(1000^2)):
  #A:=(4398/(1000^2)):                        # Cross section suspension
  cable
  E:=(105*(10^9)):                           # Modulus of Elasticity of
  suspension cable
  h:= 30:                                    # water depth at the immersion
  site
  H := 1.0:                                      # significant wave height
  g:= 9.81:                                  # gravitational accelaration
  rho:= 1031:                                # water density saline water
  Omega := 1.10:                          # time moment
  k:=(Omega^2)/g:                            # wave number
  W_ballast := 984*1000:                     # weight ballast in kg

  # Pontoon related dimensions

  Aw:=(B_f*L_f*2):
  nt:=4:                                     # number of suspention cables.
  To:= (W_ballast*g)/(nt):                   # initial force in the
  suspension cables (kN)

  assum_pos_TE := 1:                         # assumed positions of tthe TE
  under water
  B_f := 7:                                   # width floater
  L_f:=38:                                    # lenght floater
  h_f:=7.5:                                   # Height floater
  B_p:= 60:                                   # width pontoon
  Bte:= 41,2:                                # width TE
  W1:= 1.5:                                  # thickness wall TE
  tolerance:= 2:                             # tolerance for the operational
  requirements
  h_p:= 8.5:
  h_D :=1:
  M_f:= 350*1000:
  M_D := 400*1000:
  W_equip:= 400*1000:
  M_pon := (2*M_f)+ M_D + W_equip + (W_ballast):

  T:=evalf(M_pon/(Aw*rho)):                  # "Draught pontoon in [m]
  Z_f := (-T)+h_f/2:
  Z_D:= (h_p-(0.5*h_D))-T:
  Z_equip:= (h_p +0.5)-T:
  Z_To:= (h_p+1)-T:
  l1:= evalf((B_p/2)-(B_f/2)):
  l2:= evalf(T/2):
  l := 2+(h_f-T)+assum_pos_TE:               # Lenght suspention cables[m]

  KG:= 7.58:                                 # Position of COG

  KB := 2.33:                                # Position of KB
  BG:= KG-KB:                                # distance between KB and KG
  z1:= (h_p+1)-KG:                           # vertical dis. bet. To and COG
  w1:= B_p/2-B_f:
  w2:= (B_p/2)-(B_f+tolerance+(W1/2)):       # space between suspension cable
  and COG
  w3:= 5:
```



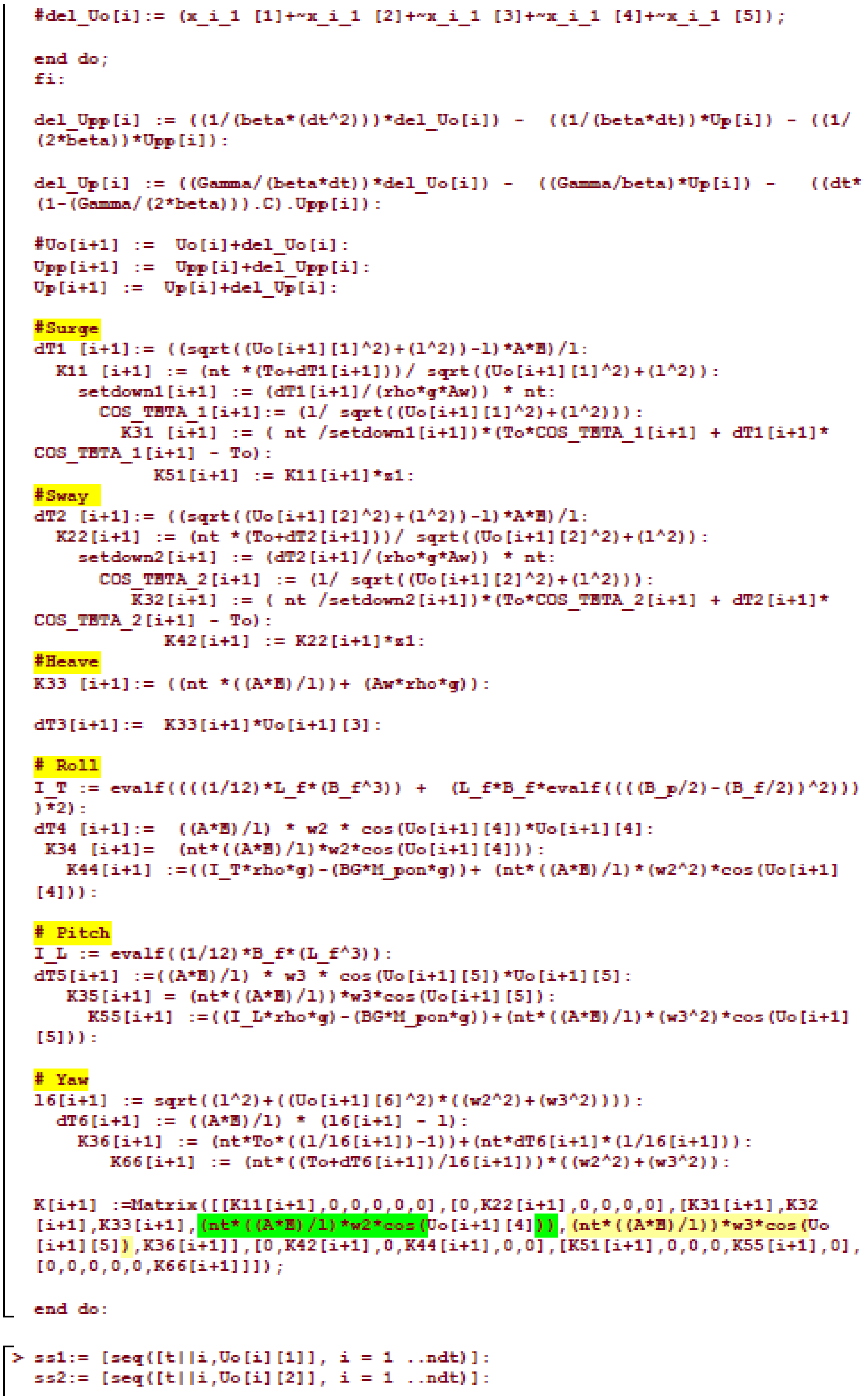
```
  Z_s_f :=(-T)+(h_f/2):
  Z_h_f :=(-T)-(h_f/2):
  r:= KG-T:

  # x and y coordinates of the structural elements
  y_f := 11:  y_D := 0: y_To := w2:  y_equip :=0:
  x_f := 0:   x_D := 0: x_To:=w3: x_equip := 0:

                    T := 4.52878072970982256805933359101                    (1)

> #Mass matrix
  Sum_Mz := ((2*M_f*Z_f)+ (M_D*Z_D)+(W_equip*Z_equip)+ (W_ballast*Z_To)):
    Ixx_pon := (2*M_f*((y_f^2)+(Z_f^2)))+(M_D*((y_D^2)+(Z_D^2))) + W_equip*(
  (y_equip^2)+(Z_equip^2)) + (W_ballast*((y_To^2)+(Z_To^2))):
      Iyy_pon := (2*M_f*((x_f^2)+(Z_f^2)))+(M_D*((x_D^2)+(Z_D^2))) + W_equip*(
  (x_equip^2)+(Z_equip^2)) + (W_ballast*((x_To^2)+(Z_To^2))):
         Izz_pon :=(2*M_f*sqrt((x_f^2)+(y_f^2))) :
           #+ (2*To*sqrt((x_To^2)+(y_To^2)))

  # Added Mass Matrix
  M_a_x := (0.5*rho*Pi*(T^2)*B_f*2):
    M_a_y := (0.5*rho*Pi*(T^2)*L_f*2):
       M_a_Z := (0.5*rho*Pi*((B_f/2)^2)*L_f*2):
          M_a_R := rho*Pi*((((B_f/4)^2)*((B_f/4)^2))+((T/2)^2)*((T + r)^2))*
  L_f*2:
            M_a_P := rho*Pi*((((L_f/4)^2)*((L_f/4)^2))+((T/2)^2)*((T + r)^2))*
  B_f*2:
               M_a_yaw := (2.5*rho*Pi* (((L_f/4)^4)+((T/4)^4))*(B_f*2)):

  Sum_M_a_Z := ((0.5*rho*Pi*((B_f/2)^2)*L_f*2)*Z_h_f) + ((0.5*rho*Pi*(T^2)*
  L_f*2)*Z_s_f)+((0.5*rho*Pi*(T^2)*B_f*2)*Z_s_f):

  M_struc := Matrix([[M_pon,0,0,0,-Sum_Mz,0], [0,M_pon,0,-Sum_Mz,0,0],[0,0,
  M_pon,0,0,0],[0,-Sum_Mz,0,Ixx_pon,0,0],[-Sum_Mz,0,0,0,Iyy_pon,0],[0,0,0,0,0,
  Izz_pon]]):
  M_hyd := evalf(Matrix([[M_a_x,0,0,0,(-1)*Sum_M_a_Z,0], [0,M_a_y,0,(-1)*
  Sum_M_a_Z,0,0],[0,0,M_a_Z,0,0,0],[0,(-1)*Sum_M_a_Z,0,M_a_R,0,0],[(-1)*
  Sum_M_a_Z,0,0,0,M_a_P,0],[0,0,0,0,0,M_a_yaw]])):
  M := evalf(M_hyd+M_struc):

> # Initial stifness matrix K[0]

          x1[0]:=0: x2[0]:=0: x3[0]:=0: x4[0]:=0: x5[0]:=0: x6[0]:=0:
  #Surge
  dT1[0] := ((sqrt((x1[0]^2)+(l^2))-l)*A*E)/l:
    K11[0] := ((nt *(To+dT1[0]))/ sqrt((x1[0]^2)+(l^2))):
      setdown1[0] := (dT1[0]/(rho*g*Aw)) * nt:
        COS_TETA_1[0] := (l/ sqrt((x1[0]^2)+(l^2))):
          K31[0] := ( nt /setdown1[0])*(To*COS_TETA_1[0] + dT1*COS_TETA_1[0] -
  To):
             K51[0]:= K11[0]*z1:
  #Sway
  dT2[0] := ((sqrt((x2[0]^2)+(l^2))-l)*A*E)/l:
    K22[0] := (nt *(To+dT2[0]))/ sqrt((x2[0]^2)+(l^2)):
      setdown2[0] := (dT2[0]/(rho*g*Aw)) * nt:
        COS_TETA_2[0] := (l/ sqrt((x1[0]^2)+(l^2))):
           K32[0] := ( nt /setdown2[0])*(To*COS_TETA_2[0]+ dT2[0]*COS_TETA_2
  [0] - To):
              K42[0] := K22[0]*z1:
  #Heave
  K33[0] := ((nt *((A*E)/l))+ (Aw*rho*g)):

  # Roll
  I_T := evalf((((1/12)*L_f*(B_f^3)) +  (L_f*B_f*evalf((((B_p/2)-(B_f/2))^2)))
```



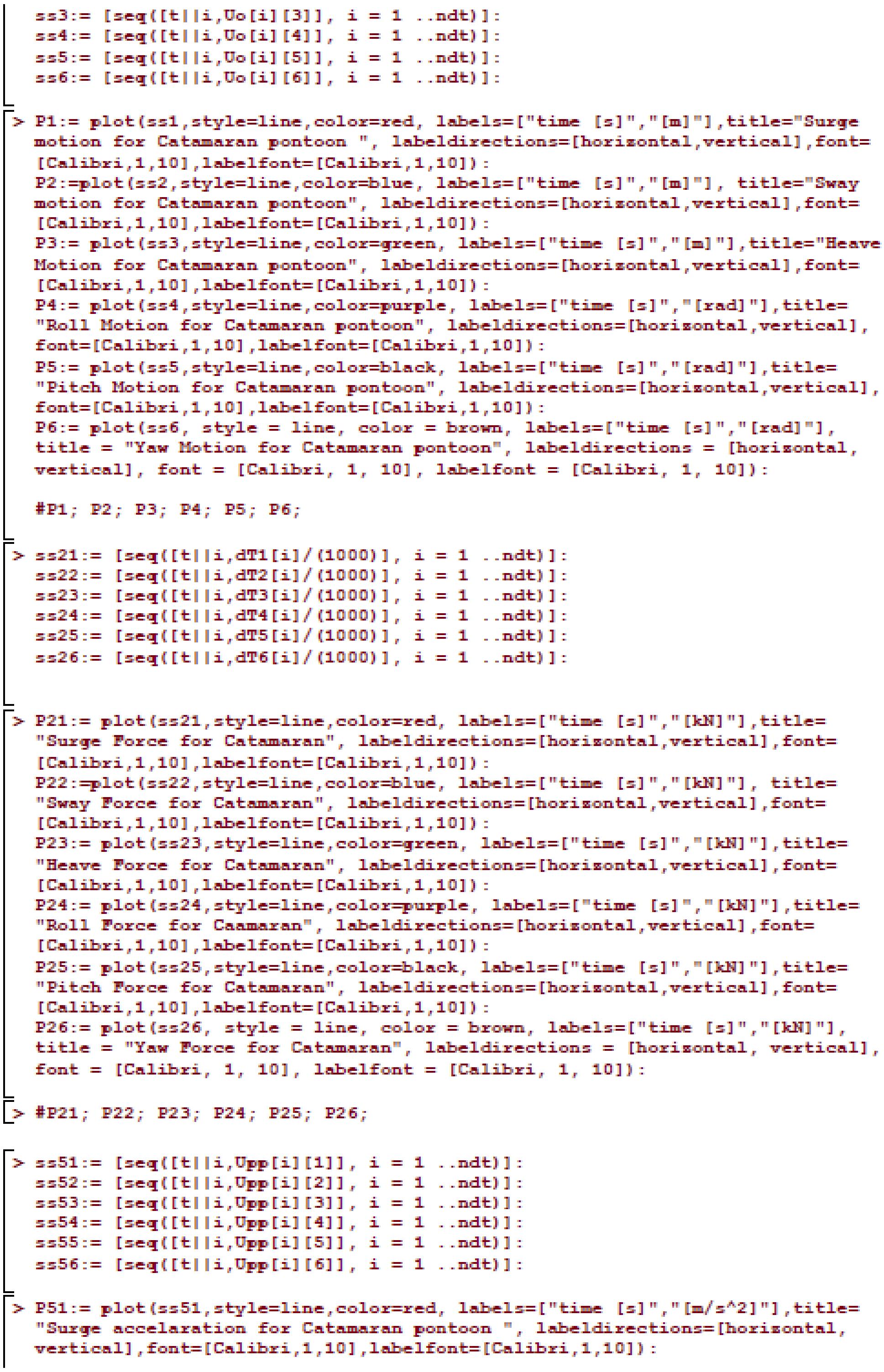
```
 )*2):
 dT4[0] :=  ((A*E)/l) * w2 * cos(x4[0])*x4[0]:
  K34[0] =  (nt*((A*E)/l)*w2*cos(x4[0])):
    K44[0] :=((I_T*rho*g)-(BG*M_pon*g))+ (nt*((A*E)/l)*(w2^2)*cos(x4[0])):

 # Pitch
 I_L := evalf((1/12)*B_f*(L_f^3)):
 dT5[0]:=((A*E)/l) * w3 * cos(x5[0])*x5[0]:
    K35[0] = (nt*((A*E)/l))*w3*cos(x5[0]):
      K55[0] :=((I_L*rho*g)-(BG*M_pon*g))+(nt*((A*E)/l)*(w3^2)*cos(x5[0])):
 # Yaw
 l6[0]:= sqrt((l^2)+((x6[0]^2)*((w2^2)+(w3^2)))):
   dT6 [0]:= ((A*E)/l) * (l6[0] - l):
     K36[0] := (nt*To*((1/l6[0])-1))+(nt*dT6[0]*(1/l6[0])):
        K66[0] := (nt*((To+dT6[0])/l6[0]))*((w2^2)+(w3^2)):
 K[0] :=Matrix([[K11[0],0,0,0,0,0],[0,K22[0],0,0,0,0],[0,0,K33[0],(nt*((A*E)
 /l)*w2*cos(x4[0])),(nt*((A*E)/l))*w3*cos(x5[0]),K36[0]],[0,K42[0],0,K44[0],
 0,0],[K51[0],0,0,0,K55[0],0],[0,0,0,0,0,K66[0]]]):

> #C Matrix ( Should be later corected)
 C :=Matrix([[0,0,0,0,0,0],[0,0,0,0,0,0],[0,0,0,0,0,0],[0,0,0,0,0,0],[0,0,0,
 0,0,0],[0,0,0,0,0,0]]):
   xi[1]:=5/100: xi[2]:=5/100: xi[3]:=5/100:
     for i from 1 to 6 do C[i,i]:= xi[1]*2*sqrt(K[0][i,i]*M[i,i])
 end:
>
 #time steps
 t||0 := 0:
 dt := 0.1: ndt:= 1500:   maxT:=(ndt-1)*dt:
   delta := 0.5:   alpha:= 0.25:
     for i from 0 to ndt do t||i := i*dt end do:

> # Forces On The pontoon

 for i from 0 to ndt do

 P[1][i] := rho*g*(H/2)*exp(k*(-l2))*cos((k*(-B_p/2))-Omega*t||i):
 P[2][i] := rho*g*(H/2)*exp(k*(-l2))*cos((k*((-B_p/2)+B_f))-Omega*t||i):
 P[3][i] := rho*g*(H/2)*exp(k*(-T))*cos((k*(-l1))-Omega*t||i):
 P[4][i] := rho*g*(H/2)*exp(k*(-l2))*cos((k*((B_p/2)-B_f))-Omega*t||i):
 P[5][i] := rho*g*(H/2)*exp(k*(-l2))*cos((k*(B_p/2))-Omega*t||i):
 P[6][i] := rho*g*(H/2)*exp(k*(-T))*cos((k*l1)-Omega*t||i):

 F[1][i] := P[1][i]*(T*L_f):
 F[2][i] := P[2][i]*(T*L_f):
 F[3][i] := P[3][i]*(B_f*L_f):
 F[4][i] := P[4][i]*(T*L_f):
 F[5][i] := P[5][i]*(T*L_f):
 F[6][i] := P[6][i]*(B_f*L_f):

 Fx[i] := 10000:
 Fy [i] := F[1][i]+F[2][i]+F[4][i]+F[5][i];
 Fz [i]:= F[3][i]+F[6][i];
 M_x[i] := (Fx[i]*l2)-(F[3][i]*l1)+(F[6][i]*l1);

 M_y [i] := 10000:
 M_z[i] := 10000:

 end do:

 for i from 0 by 1 to ndt do
   F[i]:= Vector(<Fx[i],Fy[i], Fz[i], M_x [i], M_y [i], M_z[i] >):
     end do:

 ss11:= [seq([t||i,F[i][2]], i = 1 ..ndt)]:
```



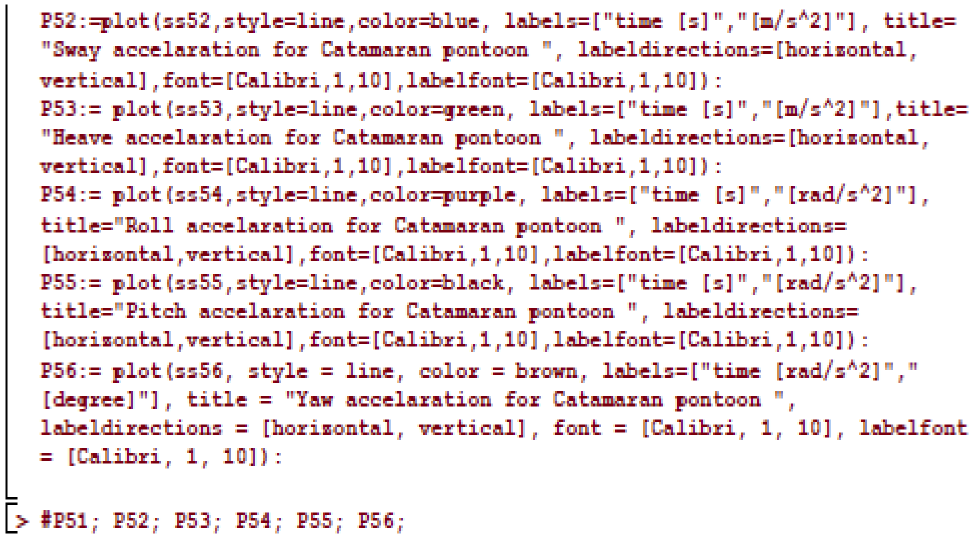
```
ss12:= [seq([t||i,F[i][3]], i = 1..ndt)]:
ss13:= [seq([t||i,F[i][4]], i = 1 ..ndt)]:

P11:= plot(ss11,style=line,color=red,title="Sway Force", labeldirections=
[horizontal,vertical],font=[Calibri,1,10],labelfont=[Calibri,1,10]):
P12:= plot(ss12,style=line,color=purple,title="Heave Force",
labeldirections=[horizontal,vertical],font=[Calibri,1,10],labelfont=
[Calibri,1,10]):
P13:= plot(ss13,style=line,color=green,title="Roll Moment", labeldirections=
[horizontal,vertical],font=[Calibri,1,10],labelfont=[Calibri,1,10]):

> #P11;P12;P13;

> # The Values of the initial displacement, velocity and accelaration
Uo[0]:= Vector(6, 0.0):   Up[0] := Vector(6, 0):
Upp [0]:= Vector(6, (MatrixInverse(M).F[0])- (MatrixInverse(M).C.Up[0]) -
(MatrixInverse(M).K[0].Uo[0])):
fs[0] := Vector(6,0):

> Gamma:= 0.5:  beta:= 1/4:  dt:= 0.01:  nr:= 2:  marge := 0.01:
range_accuracy:= 20:

a:= (M/(beta*dt)) + ((Gamma/beta).C):

b := (M/(2*beta)) + (dt*((Gamma/(2*beta))-1).C):

> for i from 0 to ndt-1  do

del_P [i]:= Vector(6,F[i+1]-F[i]):
del_P_head[i] := Vector(6,del_P[i] + (a.Up[i]) + (b.Upp[i]));
k_head [i] := Matrix(6,6,K[i] + ((Gamma/(beta*dt)).C) + ((1/(beta*(dt^2))).
M));
del_Uo[i] :=Vector[column](6,((MatrixInverse(k_head[i])).del_P_head[i]));
Uo[i+1] := Vector[column](6, Uo[i] +~ del_Uo[i]);
fs[i+1] := Vector[column](6,fs[i]+(K[i].del_Uo[i]));
err[i] := Vector[column](6,abs((Uo[i+1]-~Uo[i])/~Uo[i+1]));

if err[i][3]> 0.01

then

for j from 1 to 5 do

del_R[1]:= del_P_head[i]:
fs_j[0] := fs[i+1]:
x_i_1 [0] := Uo[i+1]:

del_x [j] := Vector[column](6,(MatrixInverse(k_head[i])).del_R[j]):
fs[j] := Vector[column](6,K[i].del_x [j]):
del_fs[j] := Vector[column](6,fs[j]-~fs[j-1])+ ((k_head[i]-~K[i]).del_x[j]):
del_R[j+1] := Vector[column](6,del_R[j] -~ del_fs[j]) :
x_i_1 [j] :=Vector[column](6, x_i_1[j-1] +~ del_x[j]):
err[j] :=Vector[column](6,((x_i_1[j] -~ x_i_1[j-1])/~x_i_1[j])):
evalf (err[j]):

#if err[j][3] < marge then
  #print(j,"iter succes",err[j][3]);
#else
  #print(j,"iter goes on",err[j][3]):
#fi;

Uo[i+1]:= x_i_1 [j];

#del_Uo[i]:= x_i_1 [j];
```



```
#del_Uo[i]:= (x_i_1 [1]+~x_i_1 [2]+~x_i_1 [3]+~x_i_1 [4]+~x_i_1 [5]);

end do;
fi:

del_Upp[i] := ((1/(beta*(dt^2)))*del_Uo[i]) -  ((1/(beta*dt))*Up[i]) - ((1/
(2*beta))*Upp[i]):

del_Up[i] := ((Gamma/(beta*dt))*del_Uo[i]) -  ((Gamma/beta)*Up[i]) -   ((dt*
(1-(Gamma/(2*beta))).C).Upp[i]):

#Uo[i+1] :=  Uo[i]+del_Uo[i]:
Upp[i+1] :=  Upp[i]+del_Upp[i]:
Up[i+1] :=  Up[i]+del_Up[i]:

#Surge
dT1 [i+1]:= ((sqrt((Uo[i+1][1]^2)+(l^2))-l)*A*E)/l:
  K11 [i+1] := (nt *(To+dT1[i+1]))/ sqrt((Uo[i+1][1]^2)+(l^2)):
    setdown1[i+1] := (dT1[i+1]/(rho*g*Aw)) * nt:
      COS_TETA_1[i+1]:= (1/ sqrt((Uo[i+1][1]^2)+(l^2))):
        K31 [i+1] := ( nt /setdown1[i+1])*(To*COS_TETA_1[i+1] + dT1[i+1]*
COS_TETA_1[i+1] - To):
           K51[i+1] := K11[i+1]*z1:
#Sway
dT2 [i+1]:= ((sqrt((Uo[i+1][2]^2)+(l^2))-l)*A*E)/l:
  K22[i+1] := (nt *(To+dT2[i+1]))/ sqrt((Uo[i+1][2]^2)+(l^2)):
    setdown2[i+1] := (dT2[i+1]/(rho*g*Aw)) * nt:
      COS_TETA_2[i+1] := (1/ sqrt((Uo[i+1][2]^2)+(l^2))):
         K32[i+1] := ( nt /setdown2[i+1])*(To*COS_TETA_2[i+1] + dT2[i+1]*
COS_TETA_2[i+1] - To):
            K42[i+1] := K22[i+1]*z1:
#Heave
K33 [i+1]:= ((nt *((A*E)/l))+ (Aw*rho*g)):

dT3[i+1]:=  K33[i+1]*Uo[i+1][3]:

# Roll
I_T := evalf((((1/12)*L_f*(B_f^3)) +  (L_f*B_f*evalf((((B_p/2)-(B_f/2))^2)))
)*2):
dT4 [i+1]:=  ((A*E)/l) * w2 * cos(Uo[i+1][4])*Uo[i+1][4]:
 K34 [i+1]=  (nt*((A*E)/l)*w2*cos(Uo[i+1][4])):
   K44[i+1] :=((I_T*rho*g)-(BG*M_pon*g))+ (nt*((A*E)/l)*(w2^2)*cos(Uo[i+1]
[4])):

# Pitch
I_L := evalf((1/12)*B_f*(L_f^3)):
dT5[i+1] :=((A*E)/l) * w3 * cos(Uo[i+1][5])*Uo[i+1][5]:
   K35[i+1] = (nt*((A*E)/l))*w3*cos(Uo[i+1][5]):
     K55[i+1] :=((I_L*rho*g)-(BG*M_pon*g))+(nt*((A*E)/l)*(w3^2)*cos(Uo[i+1]
[5])):

# Yaw
l6[i+1] := sqrt((l^2)+((Uo[i+1][6]^2)*((w2^2)+(w3^2)))):
  dT6[i+1] := ((A*E)/l) * (l6[i+1] - l):
    K36[i+1] := (nt*To*((l/l6[i+1])-1))+(nt*dT6[i+1]*(l/l6[i+1])):
       K66[i+1] := (nt*((To+dT6[i+1])/l6[i+1]))*((w2^2)+(w3^2)):

K[i+1] :=Matrix([[K11[i+1],0,0,0,0,0],[0,K22[i+1],0,0,0,0],[K31[i+1],K32
[i+1],K33[i+1],(nt*((A*E)/l)*w2*cos(Uo[i+1][4])),(nt*((A*E)/l))*w3*cos(Uo
[i+1][5]),K36[i+1]],[0,K42[i+1],0,K44[i+1],0,0],[K51[i+1],0,0,0,K55[i+1],0],
[0,0,0,0,0,K66[i+1]]]);

end do:

> ss1:= [seq([t||i,Uo[i][1]], i = 1 ..ndt)]:
  ss2:= [seq([t||i,Uo[i][2]], i = 1 ..ndt)]:
```

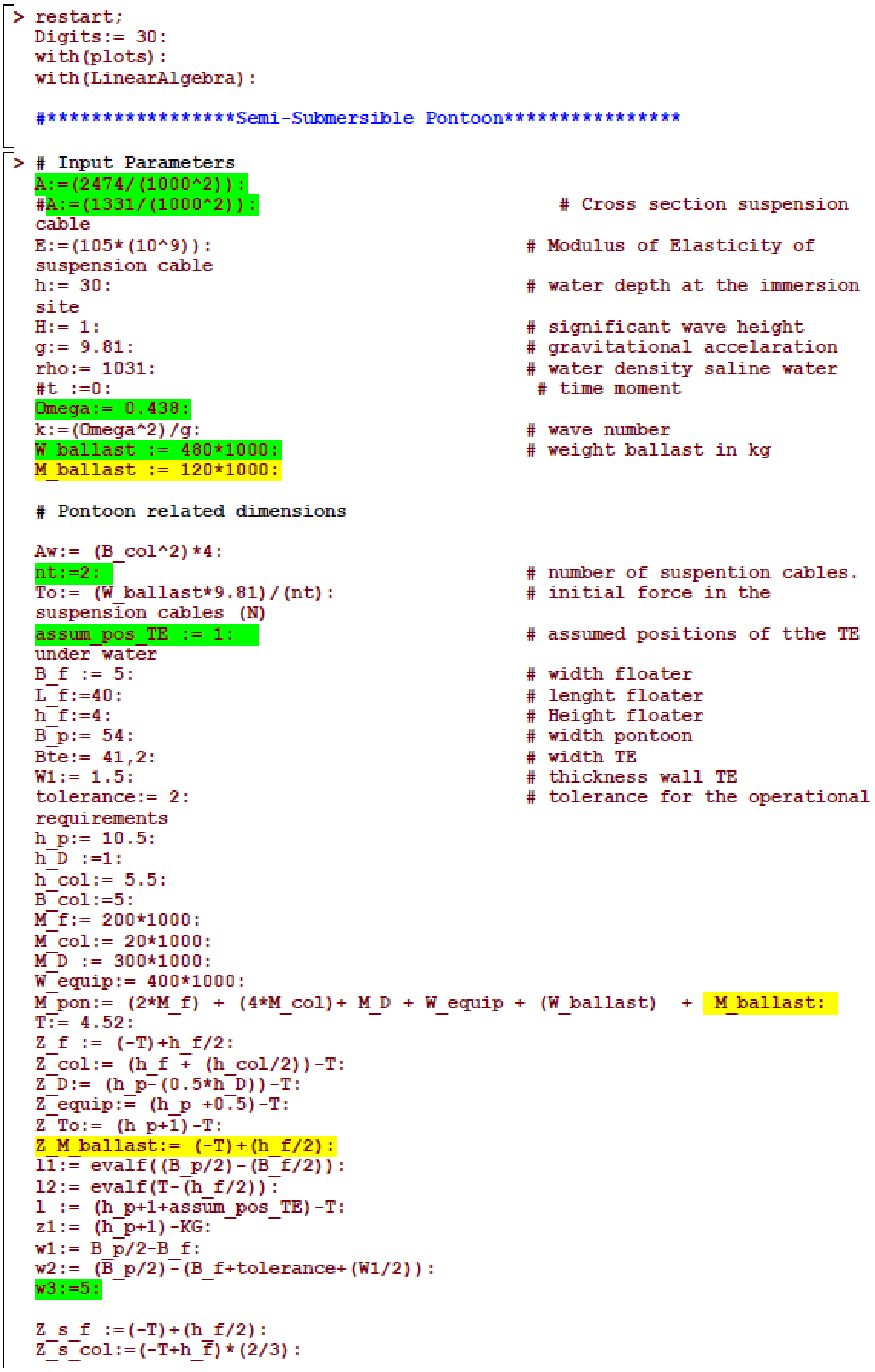


```
  ss3:= [seq([t||i,Uo[i][3]], i = 1 ..ndt)]:
  ss4:= [seq([t||i,Uo[i][4]], i = 1 ..ndt)]:
  ss5:= [seq([t||i,Uo[i][5]], i = 1 ..ndt)]:
  ss6:= [seq([t||i,Uo[i][6]], i = 1 ..ndt)]:

> P1:= plot(ss1,style=line,color=red, labels=["time [s]","[m]"],title="Surge
  motion for Catamaran pontoon ", labeldirections=[horizontal,vertical],font=
  [Calibri,1,10],labelfont=[Calibri,1,10]):
  P2:=plot(ss2,style=line,color=blue, labels=["time [s]","[m]"], title="Sway
  motion for Catamaran pontoon", labeldirections=[horizontal,vertical],font=
  [Calibri,1,10],labelfont=[Calibri,1,10]):
  P3:= plot(ss3,style=line,color=green, labels=["time [s]","[m]"],title="Heave
  Motion for Catamaran pontoon", labeldirections=[horizontal,vertical],font=
  [Calibri,1,10],labelfont=[Calibri,1,10]):
  P4:= plot(ss4,style=line,color=purple, labels=["time [s]","[rad]"],title=
  "Roll Motion for Catamaran pontoon", labeldirections=[horizontal,vertical],
  font=[Calibri,1,10],labelfont=[Calibri,1,10]):
  P5:= plot(ss5,style=line,color=black, labels=["time [s]","[rad]"],title=
  "Pitch Motion for Catamaran pontoon", labeldirections=[horizontal,vertical],
  font=[Calibri,1,10],labelfont=[Calibri,1,10]):
  P6:= plot(ss6, style = line, color = brown, labels=["time [s]","[rad]"],
  title = "Yaw Motion for Catamaran pontoon", labeldirections = [horizontal,
  vertical], font = [Calibri, 1, 10], labelfont = [Calibri, 1, 10]):

  #P1; P2; P3; P4; P5; P6;

> ss21:= [seq([t||i,dT1[i]/(1000)], i = 1 ..ndt)]:
  ss22:= [seq([t||i,dT2[i]/(1000)], i = 1 ..ndt)]:
  ss23:= [seq([t||i,dT3[i]/(1000)], i = 1 ..ndt)]:
  ss24:= [seq([t||i,dT4[i]/(1000)], i = 1 ..ndt)]:
  ss25:= [seq([t||i,dT5[i]/(1000)], i = 1 ..ndt)]:
  ss26:= [seq([t||i,dT6[i]/(1000)], i = 1 ..ndt)]:

> P21:= plot(ss21,style=line,color=red, labels=["time [s]","[kN]"],title=
  "Surge Force for Catamaran", labeldirections=[horizontal,vertical],font=
  [Calibri,1,10],labelfont=[Calibri,1,10]):
  P22:=plot(ss22,style=line,color=blue, labels=["time [s]","[kN]"], title=
  "Sway Force for Catamaran", labeldirections=[horizontal,vertical],font=
  [Calibri,1,10],labelfont=[Calibri,1,10]):
  P23:= plot(ss23,style=line,color=green, labels=["time [s]","[kN]"],title=
  "Heave Force for Catamaran", labeldirections=[horizontal,vertical],font=
  [Calibri,1,10],labelfont=[Calibri,1,10]):
  P24:= plot(ss24,style=line,color=purple, labels=["time [s]","[kN]"],title=
  "Roll Force for Caamaran", labeldirections=[horizontal,vertical],font=
  [Calibri,1,10],labelfont=[Calibri,1,10]):
  P25:= plot(ss25,style=line,color=black, labels=["time [s]","[kN]"],title=
  "Pitch Force for Catamaran", labeldirections=[horizontal,vertical],font=
  [Calibri,1,10],labelfont=[Calibri,1,10]):
  P26:= plot(ss26, style = line, color = brown, labels=["time [s]","[kN]"],
  title = "Yaw Force for Catamaran", labeldirections = [horizontal, vertical],
  font = [Calibri, 1, 10], labelfont = [Calibri, 1, 10]):

> #P21; P22; P23; P24; P25; P26;

> ss51:= [seq([t||i,Upp[i][1]], i = 1 ..ndt)]:
  ss52:= [seq([t||i,Upp[i][2]], i = 1 ..ndt)]:
  ss53:= [seq([t||i,Upp[i][3]], i = 1 ..ndt)]:
  ss54:= [seq([t||i,Upp[i][4]], i = 1 ..ndt)]:
  ss55:= [seq([t||i,Upp[i][5]], i = 1 ..ndt)]:
  ss56:= [seq([t||i,Upp[i][6]], i = 1 ..ndt)]:

> P51:= plot(ss51,style=line,color=red, labels=["time [s]","[m/s^2]"],title=
  "Surge accelaration for Catamaran pontoon ", labeldirections=[horizontal,
  vertical],font=[Calibri,1,10],labelfont=[Calibri,1,10]):
```

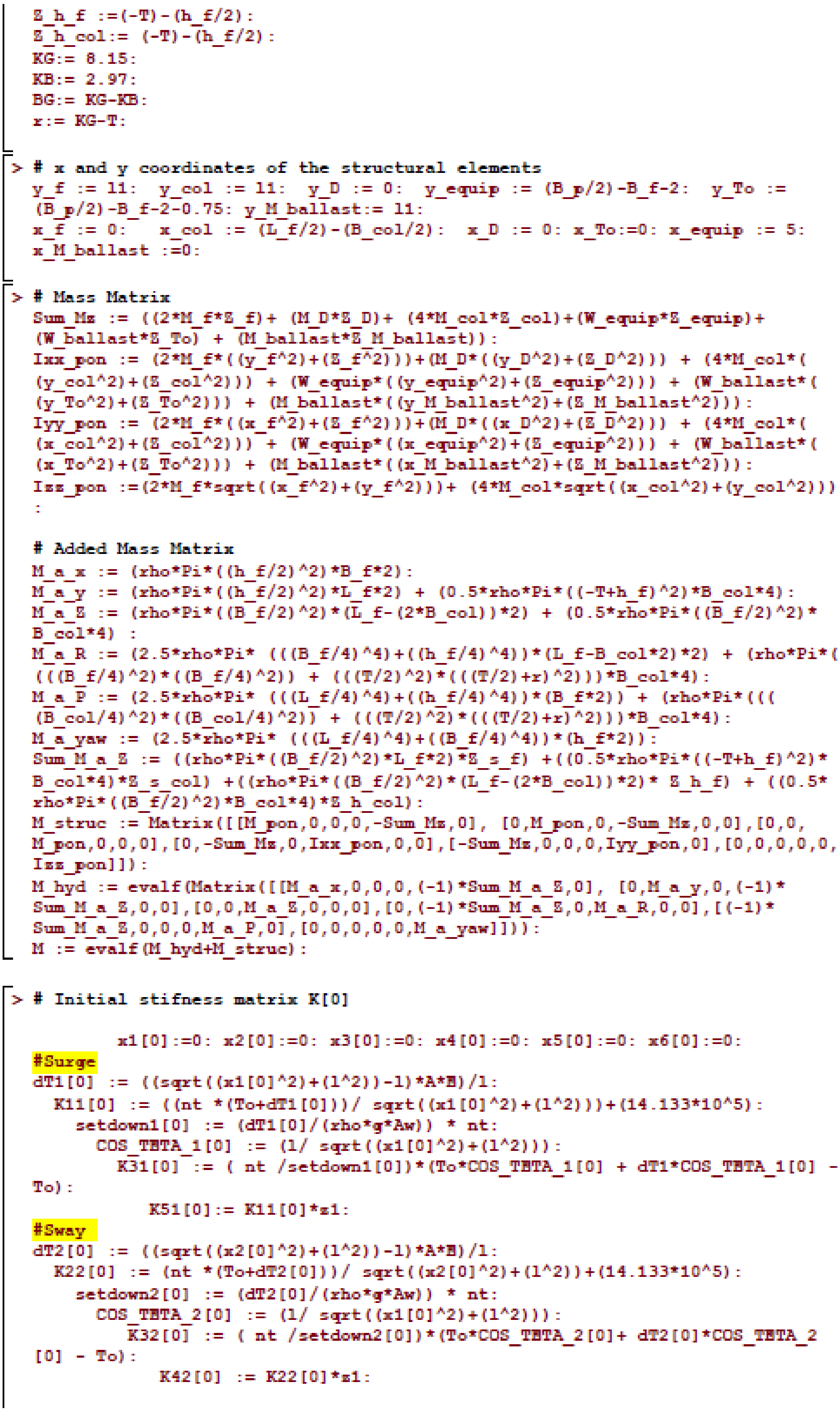


```
P52:=plot(ss52,style=line,color=blue, labels=["time [s]","[m/s^2]"], title=
"Sway accelaration for Catamaran pontoon ", labeldirections=[horizontal,
vertical],font=[Calibri,1,10],labelfont=[Calibri,1,10]):
P53:= plot(ss53,style=line,color=green, labels=["time [s]","[m/s^2]"],title=
"Heave accelaration for Catamaran pontoon ", labeldirections=[horizontal,
vertical],font=[Calibri,1,10],labelfont=[Calibri,1,10]):
P54:= plot(ss54,style=line,color=purple, labels=["time [s]","[rad/s^2]"],
title="Roll accelaration for Catamaran pontoon ", labeldirections=
[horizontal,vertical],font=[Calibri,1,10],labelfont=[Calibri,1,10]):
P55:= plot(ss55,style=line,color=black, labels=["time [s]","[rad/s^2]"],
title="Pitch accelaration for Catamaran pontoon ", labeldirections=
[horizontal,vertical],font=[Calibri,1,10],labelfont=[Calibri,1,10]):
P56:= plot(ss56, style = line, color = brown, labels=["time [rad/s^2]","
[degree]"], title = "Yaw accelaration for Catamaran pontoon ",
labeldirections = [horizontal, vertical], font = [Calibri, 1, 10], labelfont
= [Calibri, 1, 10]):

> #P51; P52; P53; P54; P55; P56;
```

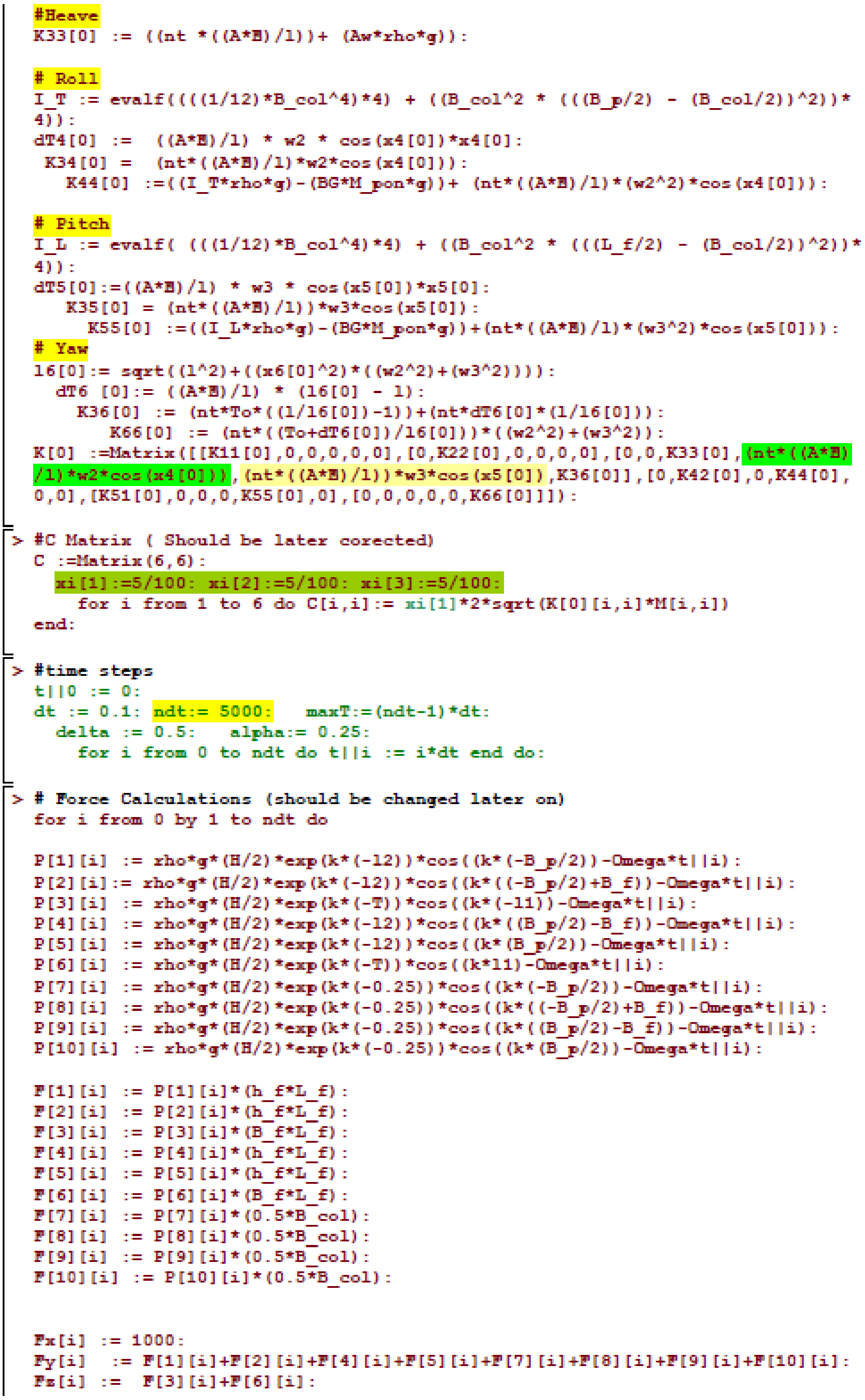
Time domain Calculations Semi-Submersible



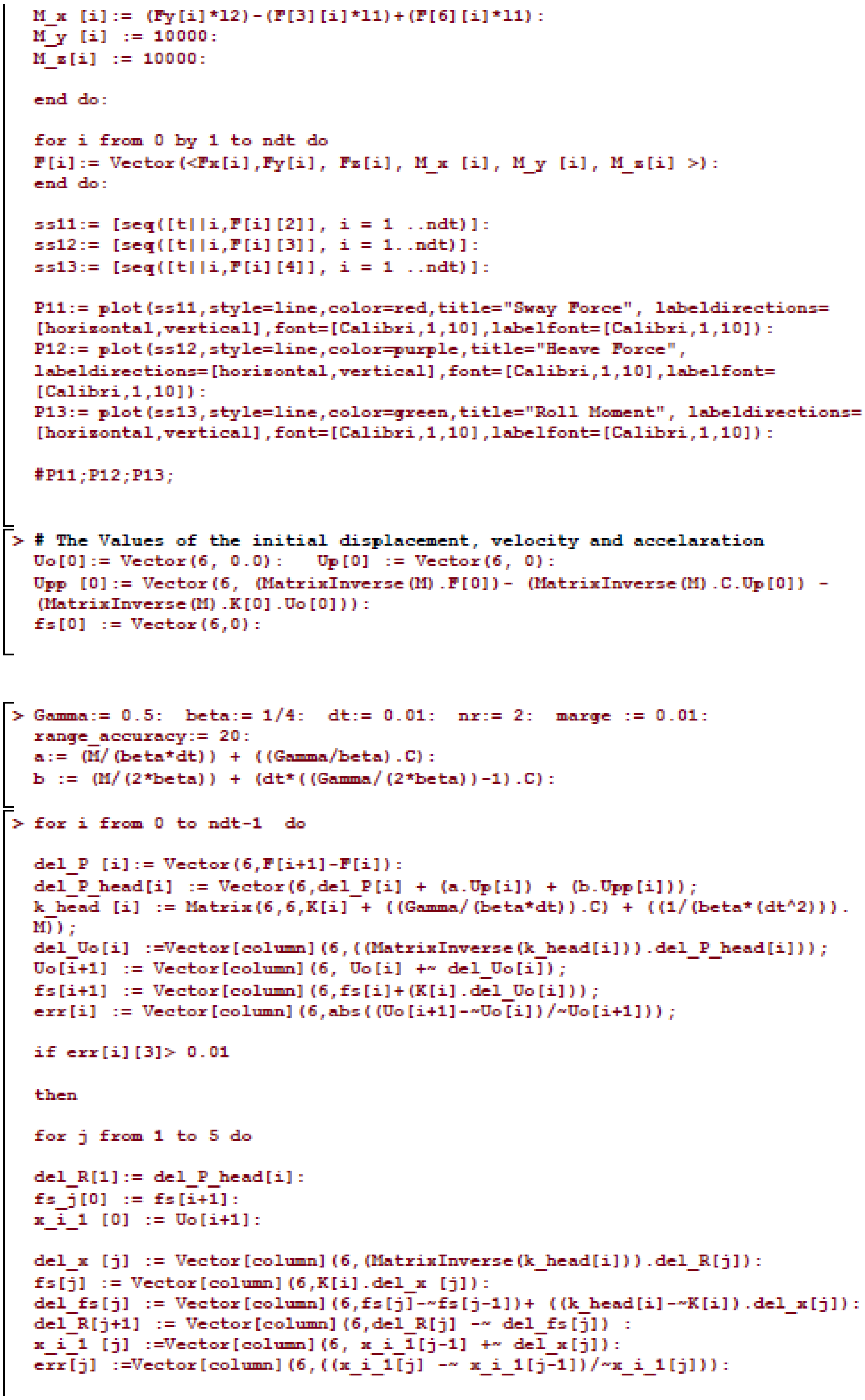
```
> restart;
  Digits:= 30:
  with(plots):
  with(LinearAlgebra):

  #*****************Semi-Submersible Pontoon****************

> # Input Parameters
  A:=(2474/(1000^2)):
  #A:=(1331/(1000^2)):                          # Cross section suspension
  cable
  E:=(105*(10^9)):                          # Modulus of Elasticity of
  suspension cable
  h:= 30:                                   # water depth at the immersion
  site
  H:= 1:                                    # significant wave height
  g:= 9.81:                                 # gravitational accelaration
  rho:= 1031:                               # water density saline water
  #t :=0:                                    # time moment
  Omega:= 0.438:
  k:=(Omega^2)/g:                           # wave number
  W_ballast := 480*1000:                    # weight ballast in kg
  M_ballast := 120*1000:

  # Pontoon related dimensions

  Aw:= (B_col^2)*4:
  nt:=2:                                    # number of suspention cables.
  To:= (W_ballast*9.81)/(nt):               # initial force in the
  suspension cables (N)
  assum_pos_TE := 1:                        # assumed positions of tthe TE
  under water
  B_f := 5:                                 # width floater
  L_f:=40:                                  # lenght floater
  h_f:=4:                                   # Height floater
  B_p:= 54:                                 # width pontoon
  Bte:= 41,2:                               # width TE
  W1:= 1.5:                                 # thickness wall TE
  tolerance:= 2:                            # tolerance for the operational
  requirements
  h_p:= 10.5:
  h_D :=1:
  h_col:= 5.5:
  B_col:=5:
  M_f:= 200*1000:
  M_col:= 20*1000:
  M_D := 300*1000:
  W_equip:= 400*1000:
  M_pon:= (2*M_f) + (4*M_col)+ M_D + W_equip + (W_ballast)  +  M_ballast:
  T:= 4.52:
  Z_f := (-T)+h_f/2:
  Z_col:= (h_f + (h_col/2))-T:
  Z_D:= (h_p-(0.5*h_D))-T:
  Z_equip:= (h_p +0.5)-T:
  Z_To:= (h_p+1)-T:
  Z_M_ballast:= (-T)+(h_f/2):
  l1:= evalf((B_p/2)-(B_f/2)):
  l2:= evalf(T-(h_f/2)):
  l := (h_p+1+assum_pos_TE)-T:
  z1:= (h_p+1)-KG:
  w1:= B_p/2-B_f:
  w2:= (B_p/2)-(B_f+tolerance+(W1/2)):
  w3:=5:

  Z_s_f :=(-T)+(h_f/2):
  Z_s_col:=(-T+h_f)*(2/3):
```



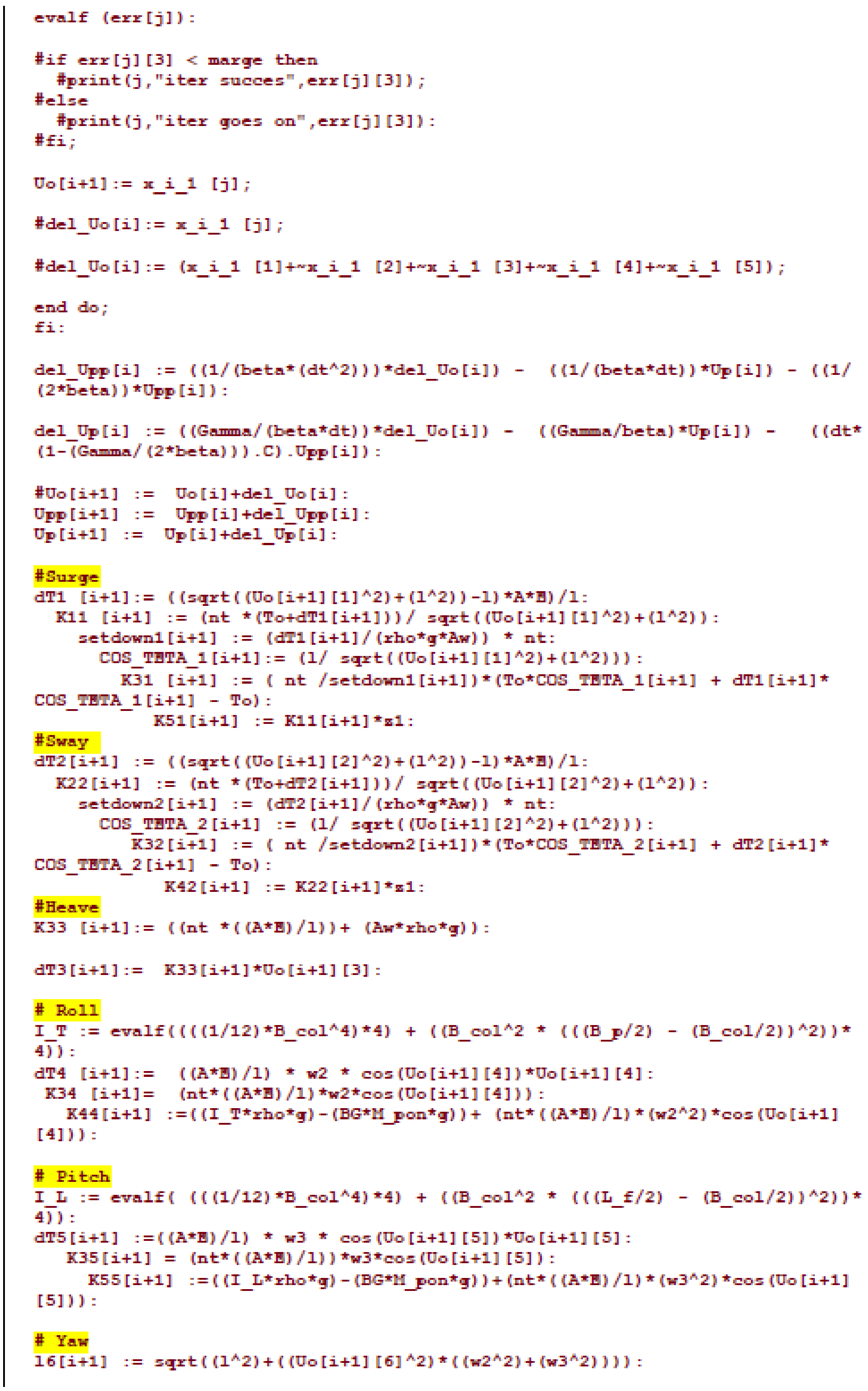
```
  Z_h_f :=(-T)-(h_f/2):
  Z_h_col:= (-T)-(h_f/2):
  KG:= 8.15:
  KB:= 2.97:
  BG:= KG-KB:
  r:= KG-T:

> # x and y coordinates of the structural elements
  y_f := 11:  y_col := 11:  y_D := 0:  y_equip := (B_p/2)-B_f-2:  y_To :=
  (B_p/2)-B_f-2-0.75: y_M_ballast:= 11:
  x_f := 0:   x_col := (L_f/2)-(B_col/2):  x_D := 0: x_To:=0: x_equip := 5:
  x_M_ballast :=0:

> # Mass Matrix
  Sum_Mz := ((2*M_f*Z_f)+ (M_D*Z_D)+ (4*M_col*Z_col)+(W_equip*Z_equip)+
  (W_ballast*Z_To) + (M_ballast*Z_M_ballast)):
  Ixx_pon := (2*M_f*((y_f^2)+(Z_f^2)))+(M_D*((y_D^2)+(Z_D^2))) + (4*M_col*(
  (y_col^2)+(Z_col^2))) + (W_equip*((y_equip^2)+(Z_equip^2))) + (W_ballast*(
  (y_To^2)+(Z_To^2))) + (M_ballast*((y_M_ballast^2)+(Z_M_ballast^2))):
  Iyy_pon := (2*M_f*((x_f^2)+(Z_f^2)))+(M_D*((x_D^2)+(Z_D^2))) + (4*M_col*(
  (x_col^2)+(Z_col^2))) + (W_equip*((x_equip^2)+(Z_equip^2))) + (W_ballast*(
  (x_To^2)+(Z_To^2))) + (M_ballast*((x_M_ballast^2)+(Z_M_ballast^2))):
  Izz_pon :=(2*M_f*sqrt((x_f^2)+(y_f^2)))+ (4*M_col*sqrt((x_col^2)+(y_col^2)))
  :

  # Added Mass Matrix
  M_a_x := (rho*Pi*((h_f/2)^2)*B_f*2):
  M_a_y := (rho*Pi*((h_f/2)^2)*L_f*2) + (0.5*rho*Pi*((-T+h_f)^2)*B_col*4):
  M_a_Z := (rho*Pi*((B_f/2)^2)*(L_f-(2*B_col))*2) + (0.5*rho*Pi*((B_f/2)^2)*
  B_col*4) :
  M_a_R := (2.5*rho*Pi* (((B_f/4)^4)+((h_f/4)^4))*(L_f-B_col*2)*2) + (rho*Pi*(
  (((B_f/4)^2)*((B_f/4)^2)) + (((T/2)^2)*(((T/2)+r)^2)))*B_col*4):
  M_a_P := (2.5*rho*Pi* (((L_f/4)^4)+((h_f/4)^4))*(B_f*2)) + (rho*Pi*(((
  (B_col/4)^2)*((B_col/4)^2)) + (((T/2)^2)*(((T/2)+r)^2)))*B_col*4):
  M_a_yaw := (2.5*rho*Pi* (((L_f/4)^4)+((B_f/4)^4))*(h_f*2)):
  Sum_M_a_Z := ((rho*Pi*((B_f/2)^2)*L_f*2)*Z_s_f) +((0.5*rho*Pi*((-T+h_f)^2)*
  B_col*4)*Z_s_col) +((rho*Pi*((B_f/2)^2)*(L_f-(2*B_col))*2)* Z_h_f) + ((0.5*
  rho*Pi*((B_f/2)^2)*B_col*4)*Z_h_col):
  M_struc := Matrix([[M_pon,0,0,0,-Sum_Mz,0], [0,M_pon,0,-Sum_Mz,0,0],[0,0,
  M_pon,0,0,0],[0,-Sum_Mz,0,Ixx_pon,0,0],[-Sum_Mz,0,0,0,Iyy_pon,0],[0,0,0,0,0,
  Izz_pon]]):
  M_hyd := evalf(Matrix([[M_a_x,0,0,0,(-1)*Sum_M_a_Z,0], [0,M_a_y,0,(-1)*
  Sum_M_a_Z,0,0],[0,0,M_a_Z,0,0,0],[0,(-1)*Sum_M_a_Z,0,M_a_R,0,0],[(-1)*
  Sum_M_a_Z,0,0,0,M_a_P,0],[0,0,0,0,0,M_a_yaw]])):
  M := evalf(M_hyd+M_struc):

> # Initial stifness matrix K[0]

          x1[0]:=0: x2[0]:=0: x3[0]:=0: x4[0]:=0: x5[0]:=0: x6[0]:=0:
  #Surge
  dT1[0] := ((sqrt((x1[0]^2)+(1^2))-1)*A*E)/1:
    K11[0] := ((nt *(To+dT1[0]))/ sqrt((x1[0]^2)+(1^2)))+(14.133*10^5):
      setdown1[0] := (dT1[0]/(rho*g*Aw)) * nt:
        COS_TETA_1[0] := (1/ sqrt((x1[0]^2)+(1^2))):
          K31[0] := ( nt /setdown1[0])*(To*COS_TETA_1[0] + dT1*COS_TETA_1[0] -
  To):
             K51[0]:= K11[0]*z1:
  #Sway
  dT2[0] := ((sqrt((x2[0]^2)+(1^2))-1)*A*E)/1:
    K22[0] := (nt *(To+dT2[0]))/ sqrt((x2[0]^2)+(1^2))+(14.133*10^5):
      setdown2[0] := (dT2[0]/(rho*g*Aw)) * nt:
        COS_TETA_2[0] := (1/ sqrt((x1[0]^2)+(1^2))):
           K32[0] := ( nt /setdown2[0])*(To*COS_TETA_2[0]+ dT2[0]*COS_TETA_2
  [0] - To):
              K42[0] := K22[0]*z1:
```



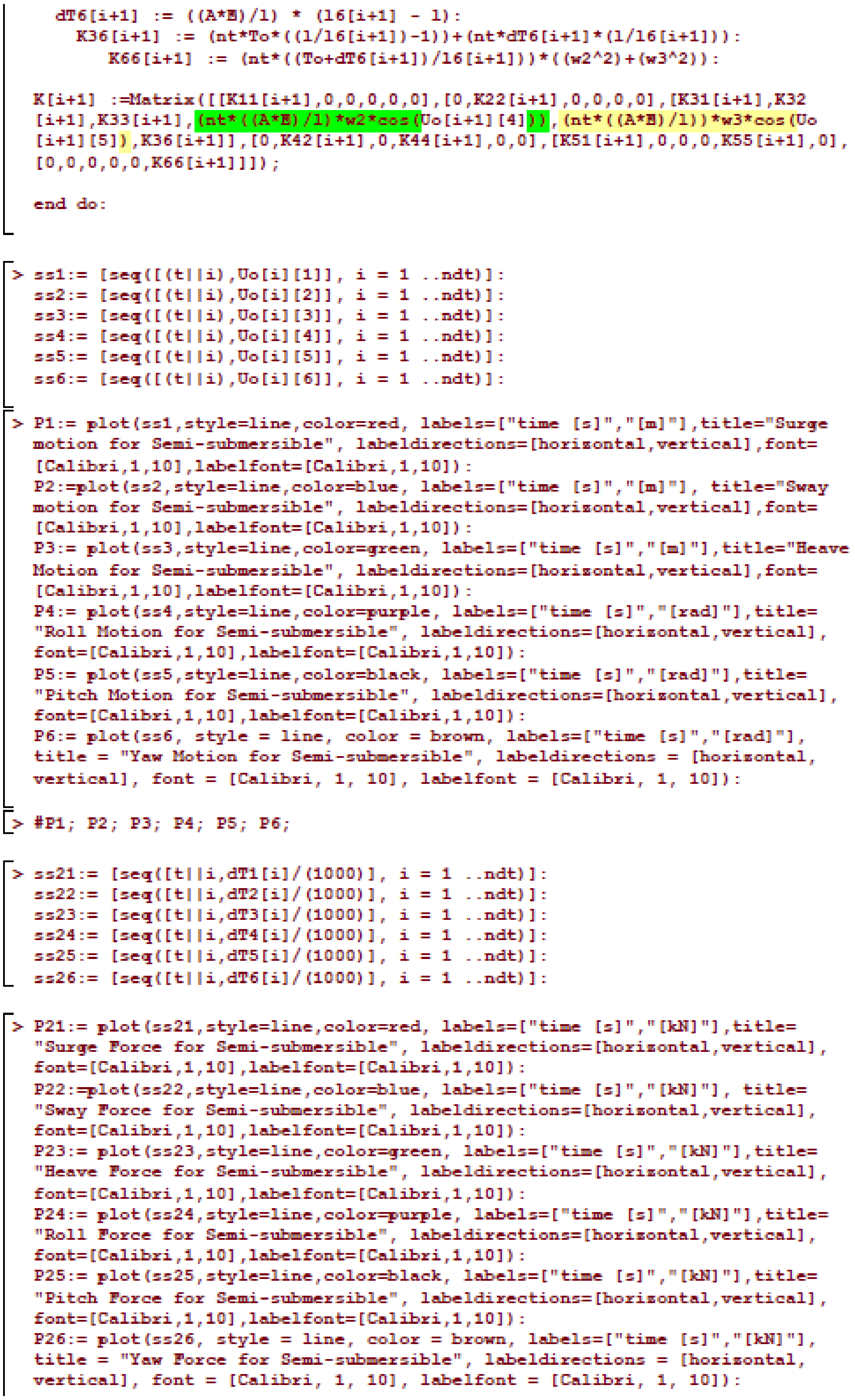
```
#Heave
K33[0] := ((nt *((A*B)/l))+ (Aw*rho*g)):

# Roll
I_T := evalf((((1/12)*B_col^4)*4) + ((B_col^2 * (((B_p/2) - (B_col/2))^2))*
4)):
dT4[0] :=  ((A*B)/l) * w2 * cos(x4[0])*x4[0]:
 K34[0] =  (nt*((A*B)/l)*w2*cos(x4[0])):
   K44[0] :=((I_T*rho*g)-(BG*M_pon*g))+ (nt*((A*B)/l)*(w2^2)*cos(x4[0])):

# Pitch
I_L := evalf( (((1/12)*B_col^4)*4) + ((B_col^2 * (((L_f/2) - (B_col/2))^2))*
4)):
dT5[0]:=((A*B)/l) * w3 * cos(x5[0])*x5[0]:
   K35[0] = (nt*((A*B)/l))*w3*cos(x5[0]):
     K55[0] :=((I_L*rho*g)-(BG*M_pon*g))+(nt*((A*B)/l)*(w3^2)*cos(x5[0])):
# Yaw
l6[0]:= sqrt((l^2)+((x6[0]^2)*((w2^2)+(w3^2)))):
  dT6 [0]:= ((A*B)/l) * (l6[0] - l):
    K36[0] := (nt*To*((1/l6[0])-1))+(nt*dT6[0]*(1/l6[0])):
       K66[0] := (nt*((To+dT6[0])/l6[0]))*((w2^2)+(w3^2)):
K[0] :=Matrix([[K11[0],0,0,0,0,0],[0,K22[0],0,0,0,0],[0,0,K33[0],(nt*((A*B)
/l)*w2*cos(x4[0])),(nt*((A*B)/l))*w3*cos(x5[0]),K36[0]],[0,K42[0],0,K44[0],
0,0],[K51[0],0,0,0,K55[0],0],[0,0,0,0,0,K66[0]]]):

> #C Matrix ( Should be later corected)
C :=Matrix(6,6):
  xi[1]:=5/100: xi[2]:=5/100: xi[3]:=5/100:
    for i from 1 to 6 do C[i,i]:= xi[1]*2*sqrt(K[0][i,i]*M[i,i])
end:

> #time steps
t||0 := 0:
dt := 0.1: ndt:= 5000:   maxT:=(ndt-1)*dt:
  delta := 0.5:   alpha:= 0.25:
    for i from 0 to ndt do t||i := i*dt end do:

> # Force Calculations (should be changed later on)
for i from 0 by 1 to ndt do

P[1][i] := rho*g*(H/2)*exp(k*(-12))*cos((k*(-B_p/2))-Omega*t||i):
P[2][i]:= rho*g*(H/2)*exp(k*(-12))*cos((k*((-B_p/2)+B_f))-Omega*t||i):
P[3][i] := rho*g*(H/2)*exp(k*(-T))*cos((k*(-l1))-Omega*t||i):
P[4][i] := rho*g*(H/2)*exp(k*(-12))*cos((k*((B_p/2)-B_f))-Omega*t||i):
P[5][i] := rho*g*(H/2)*exp(k*(-12))*cos((k*(B_p/2))-Omega*t||i):
P[6][i] := rho*g*(H/2)*exp(k*(-T))*cos((k*l1)-Omega*t||i):
P[7][i] := rho*g*(H/2)*exp(k*(-0.25))*cos((k*(-B_p/2))-Omega*t||i):
P[8][i] := rho*g*(H/2)*exp(k*(-0.25))*cos((k*((-B_p/2)+B_f))-Omega*t||i):
P[9][i] := rho*g*(H/2)*exp(k*(-0.25))*cos((k*((B_p/2)-B_f))-Omega*t||i):
P[10][i] := rho*g*(H/2)*exp(k*(-0.25))*cos((k*(B_p/2))-Omega*t||i):

F[1][i] := P[1][i]*(h_f*L_f):
F[2][i] := P[2][i]*(h_f*L_f):
F[3][i] := P[3][i]*(B_f*L_f):
F[4][i] := P[4][i]*(h_f*L_f):
F[5][i] := P[5][i]*(h_f*L_f):
F[6][i] := P[6][i]*(B_f*L_f):
F[7][i] := P[7][i]*(0.5*B_col):
F[8][i] := P[8][i]*(0.5*B_col):
F[9][i] := P[9][i]*(0.5*B_col):
F[10][i] := P[10][i]*(0.5*B_col):


Fx[i] := 1000:
Fy[i]  := F[1][i]+F[2][i]+F[4][i]+F[5][i]+F[7][i]+F[8][i]+F[9][i]+F[10][i]:
Fz[i] :=  F[3][i]+F[6][i]:
```



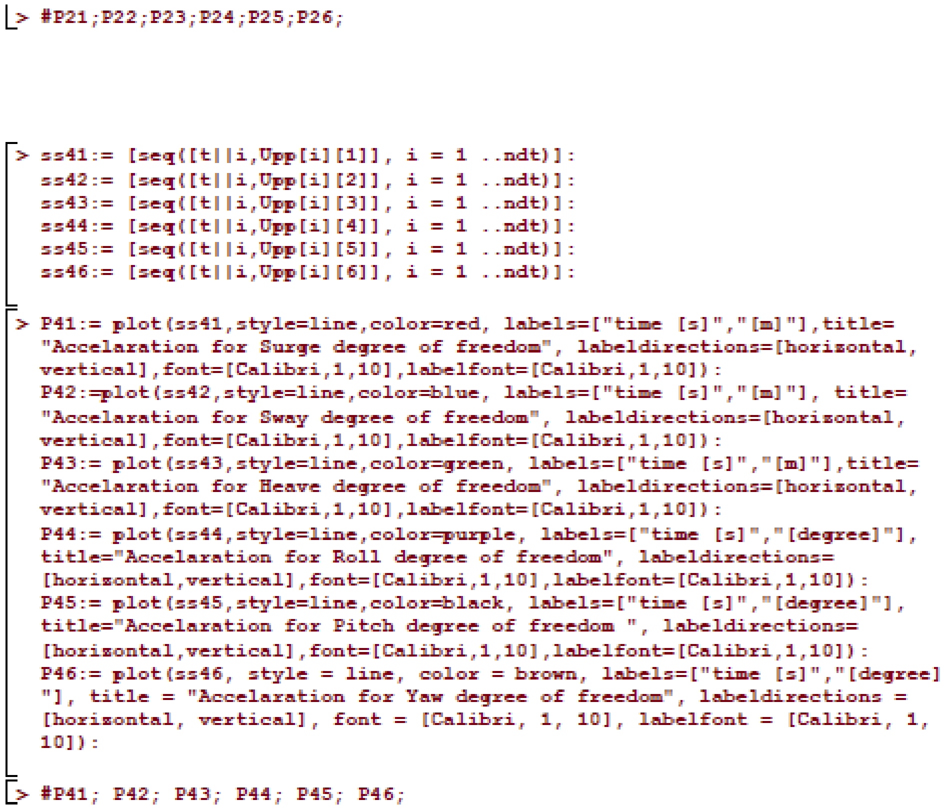
```
M_x [i]:= (Fy[i]*12)-(F[3][i]*11)+(F[6][i]*11):
M_y [i] := 10000:
M_z[i] := 10000:

end do:

for i from 0 by 1 to ndt do
F[i]:= Vector(<Fx[i],Fy[i], Fz[i], M_x [i], M_y [i], M_z[i] >):
end do:

ss11:= [seq([t||i,F[i][2]], i = 1 ..ndt)]:
ss12:= [seq([t||i,F[i][3]], i = 1..ndt)]:
ss13:= [seq([t||i,F[i][4]], i = 1 ..ndt)]:

P11:= plot(ss11,style=line,color=red,title="Sway Force", labeldirections=
[horizontal,vertical],font=[Calibri,1,10],labelfont=[Calibri,1,10]):
P12:= plot(ss12,style=line,color=purple,title="Heave Force",
labeldirections=[horizontal,vertical],font=[Calibri,1,10],labelfont=
[Calibri,1,10]):
P13:= plot(ss13,style=line,color=green,title="Roll Moment", labeldirections=
[horizontal,vertical],font=[Calibri,1,10],labelfont=[Calibri,1,10]):

#P11;P12;P13;

> # The Values of the initial displacement, velocity and accelaration
Uo[0]:= Vector(6, 0.0):   Up[0] := Vector(6, 0):
Upp [0]:= Vector(6, (MatrixInverse(M).F[0])- (MatrixInverse(M).C.Up[0]) -
(MatrixInverse(M).K[0].Uo[0])):
fs[0] := Vector(6,0):


> Gamma:= 0.5:  beta:= 1/4:  dt:= 0.01:  nr:= 2:  marge := 0.01:
range_accuracy:= 20:
a:= (M/(beta*dt)) + ((Gamma/beta).C):
b := (M/(2*beta)) + (dt*((Gamma/(2*beta))-1).C):
> for i from 0 to ndt-1  do

del_P [i]:= Vector(6,F[i+1]-F[i]):
del_P_head[i] := Vector(6,del_P[i] + (a.Up[i]) + (b.Upp[i]));
k_head [i] := Matrix(6,6,K[i] + ((Gamma/(beta*dt)).C) + ((1/(beta*(dt^2))).
M));
del_Uo[i] :=Vector[column](6,((MatrixInverse(k_head[i])).del_P_head[i]));
Uo[i+1] := Vector[column](6, Uo[i] +~ del_Uo[i]);
fs[i+1] := Vector[column](6,fs[i]+(K[i].del_Uo[i]));
err[i] := Vector[column](6,abs((Uo[i+1]-~Uo[i])/~Uo[i+1]));

if err[i][3]> 0.01

then

for j from 1 to 5 do

del_R[1]:= del_P_head[i]:
fs_j[0] := fs[i+1]:
x_i_1 [0] := Uo[i+1]:

del_x [j] := Vector[column](6,(MatrixInverse(k_head[i])).del_R[j]):
fs[j] := Vector[column](6,K[i].del_x [j]):
del_fs[j] := Vector[column](6,fs[j]-~fs[j-1])+ ((k_head[i]-~K[i]).del_x[j]):
del_R[j+1] := Vector[column](6,del_R[j] -~ del_fs[j]) :
x_i_1 [j] :=Vector[column](6, x_i_1[j-1] +~ del_x[j]):
err[j] :=Vector[column](6,((x_i_1[j] -~ x_i_1[j-1])/~x_i_1[j])):
```



```
evalf (err[j]):

#if err[j][3] < marge then
  #print(j,"iter succes",err[j][3]);
#else
  #print(j,"iter goes on",err[j][3]):
#fi;

Uo[i+1]:= x_i_1 [j];

#del_Uo[i]:= x_i_1 [j];

#del_Uo[i]:= (x_i_1 [1]+~x_i_1 [2]+~x_i_1 [3]+~x_i_1 [4]+~x_i_1 [5]);

end do;
fi:

del_Upp[i] := ((1/(beta*(dt^2)))*del_Uo[i]) -  ((1/(beta*dt))*Up[i]) - ((1/
(2*beta))*Upp[i]):

del_Up[i] := ((Gamma/(beta*dt))*del_Uo[i]) -  ((Gamma/beta)*Up[i]) -   ((dt*
(1-(Gamma/(2*beta))).C).Upp[i]):

#Uo[i+1] :=  Uo[i]+del_Uo[i]:
Upp[i+1] :=  Upp[i]+del_Upp[i]:
Up[i+1] :=  Up[i]+del_Up[i]:

#Surge
dT1 [i+1]:= ((sqrt((Uo[i+1][1]^2)+(l^2))-l)*A*E)/l:
  K11 [i+1] := (nt *(To+dT1[i+1]))/ sqrt((Uo[i+1][1]^2)+(l^2)):
    setdown1[i+1] := (dT1[i+1]/(rho*g*Aw)) * nt:
      COS_TETA_1[i+1]:= (l/ sqrt((Uo[i+1][1]^2)+(l^2))):
        K31 [i+1] := ( nt /setdown1[i+1])*(To*COS_TETA_1[i+1] + dT1[i+1]*
COS_TETA_1[i+1] - To):
           K51[i+1] := K11[i+1]*z1:
#Sway
dT2[i+1] := ((sqrt((Uo[i+1][2]^2)+(l^2))-l)*A*E)/l:
  K22[i+1] := (nt *(To+dT2[i+1]))/ sqrt((Uo[i+1][2]^2)+(l^2)):
    setdown2[i+1] := (dT2[i+1]/(rho*g*Aw)) * nt:
      COS_TETA_2[i+1] := (l/ sqrt((Uo[i+1][2]^2)+(l^2))):
         K32[i+1] := ( nt /setdown2[i+1])*(To*COS_TETA_2[i+1] + dT2[i+1]*
COS_TETA_2[i+1] - To):
            K42[i+1] := K22[i+1]*z1:
#Heave
K33 [i+1]:= ((nt *((A*E)/l))+ (Aw*rho*g)):

dT3[i+1]:=  K33[i+1]*Uo[i+1][3]:

# Roll
I_T := evalf((((1/12)*B_col^4)*4) + ((B_col^2 * (((B_p/2) - (B_col/2))^2))*
4)):
dT4 [i+1]:=  ((A*E)/l) * w2 * cos(Uo[i+1][4])*Uo[i+1][4]:
 K34 [i+1]=  (nt*((A*E)/l)*w2*cos(Uo[i+1][4])):
   K44[i+1] :=((I_T*rho*g)-(BG*M_pon*g))+ (nt*((A*E)/l)*(w2^2)*cos(Uo[i+1]
[4])):

# Pitch
I_L := evalf( (((1/12)*B_col^4)*4) + ((B_col^2 * (((L_f/2) - (B_col/2))^2))*
4)):
dT5[i+1] :=((A*E)/l) * w3 * cos(Uo[i+1][5])*Uo[i+1][5]:
   K35[i+1] = (nt*((A*E)/l))*w3*cos(Uo[i+1][5]):
     K55[i+1] :=((I_L*rho*g)-(BG*M_pon*g))+(nt*((A*E)/l)*(w3^2)*cos(Uo[i+1]
[5])):

# Yaw
l6[i+1] := sqrt((l^2)+((Uo[i+1][6]^2)*((w2^2)+(w3^2)))):
```



```
    dT6[i+1] := ((A*E)/l) * (l6[i+1] - l):
      K36[i+1] := (nt*To*((l/l6[i+1])-1))+(nt*dT6[i+1]*(l/l6[i+1])):
         K66[i+1] := (nt*((To+dT6[i+1])/l6[i+1]))*((w2^2)+(w3^2)):

  K[i+1] :=Matrix([[K11[i+1],0,0,0,0,0],[0,K22[i+1],0,0,0,0],[K31[i+1],K32
  [i+1],K33[i+1],(nt*((A*E)/l)*w2*cos(Uo[i+1][4])),(nt*((A*E)/l))*w3*cos(Uo
  [i+1][5]),K36[i+1]],[0,K42[i+1],0,K44[i+1],0,0],[K51[i+1],0,0,0,K55[i+1],0],
  [0,0,0,0,0,K66[i+1]]]);

  end do:

> ss1:= [seq([(t||i),Uo[i][1]], i = 1 ..ndt)]:
  ss2:= [seq([(t||i),Uo[i][2]], i = 1 ..ndt)]:
  ss3:= [seq([(t||i),Uo[i][3]], i = 1 ..ndt)]:
  ss4:= [seq([(t||i),Uo[i][4]], i = 1 ..ndt)]:
  ss5:= [seq([(t||i),Uo[i][5]], i = 1 ..ndt)]:
  ss6:= [seq([(t||i),Uo[i][6]], i = 1 ..ndt)]:

> P1:= plot(ss1,style=line,color=red, labels=["time [s]","[m]"],title="Surge
  motion for Semi-submersible", labeldirections=[horizontal,vertical],font=
  [Calibri,1,10],labelfont=[Calibri,1,10]):
  P2:=plot(ss2,style=line,color=blue, labels=["time [s]","[m]"], title="Sway
  motion for Semi-submersible", labeldirections=[horizontal,vertical],font=
  [Calibri,1,10],labelfont=[Calibri,1,10]):
  P3:= plot(ss3,style=line,color=green, labels=["time [s]","[m]"],title="Heave
  Motion for Semi-submersible", labeldirections=[horizontal,vertical],font=
  [Calibri,1,10],labelfont=[Calibri,1,10]):
  P4:= plot(ss4,style=line,color=purple, labels=["time [s]","[rad]"],title=
  "Roll Motion for Semi-submersible", labeldirections=[horizontal,vertical],
  font=[Calibri,1,10],labelfont=[Calibri,1,10]):
  P5:= plot(ss5,style=line,color=black, labels=["time [s]","[rad]"],title=
  "Pitch Motion for Semi-submersible", labeldirections=[horizontal,vertical],
  font=[Calibri,1,10],labelfont=[Calibri,1,10]):
  P6:= plot(ss6, style = line, color = brown, labels=["time [s]","[rad]"],
  title = "Yaw Motion for Semi-submersible", labeldirections = [horizontal,
  vertical], font = [Calibri, 1, 10], labelfont = [Calibri, 1, 10]):

> #P1; P2; P3; P4; P5; P6;

> ss21:= [seq([t||i,dT1[i]/(1000)], i = 1 ..ndt)]:
  ss22:= [seq([t||i,dT2[i]/(1000)], i = 1 ..ndt)]:
  ss23:= [seq([t||i,dT3[i]/(1000)], i = 1 ..ndt)]:
  ss24:= [seq([t||i,dT4[i]/(1000)], i = 1 ..ndt)]:
  ss25:= [seq([t||i,dT5[i]/(1000)], i = 1 ..ndt)]:
  ss26:= [seq([t||i,dT6[i]/(1000)], i = 1 ..ndt)]:

> P21:= plot(ss21,style=line,color=red, labels=["time [s]","[kN]"],title=
  "Surge Force for Semi-submersible", labeldirections=[horizontal,vertical],
  font=[Calibri,1,10],labelfont=[Calibri,1,10]):
  P22:=plot(ss22,style=line,color=blue, labels=["time [s]","[kN]"], title=
  "Sway Force for Semi-submersible", labeldirections=[horizontal,vertical],
  font=[Calibri,1,10],labelfont=[Calibri,1,10]):
  P23:= plot(ss23,style=line,color=green, labels=["time [s]","[kN]"],title=
  "Heave Force for Semi-submersible", labeldirections=[horizontal,vertical],
  font=[Calibri,1,10],labelfont=[Calibri,1,10]):
  P24:= plot(ss24,style=line,color=purple, labels=["time [s]","[kN]"],title=
  "Roll Force for Semi-submersible", labeldirections=[horizontal,vertical],
  font=[Calibri,1,10],labelfont=[Calibri,1,10]):
  P25:= plot(ss25,style=line,color=black, labels=["time [s]","[kN]"],title=
  "Pitch Force for Semi-submersible", labeldirections=[horizontal,vertical],
  font=[Calibri,1,10],labelfont=[Calibri,1,10]):
  P26:= plot(ss26, style = line, color = brown, labels=["time [s]","[kN]"],
  title = "Yaw Force for Semi-submersible", labeldirections = [horizontal,
  vertical], font = [Calibri, 1, 10], labelfont = [Calibri, 1, 10]):
```



```
> #P21;P22;P23;P24;P25;P26;
```

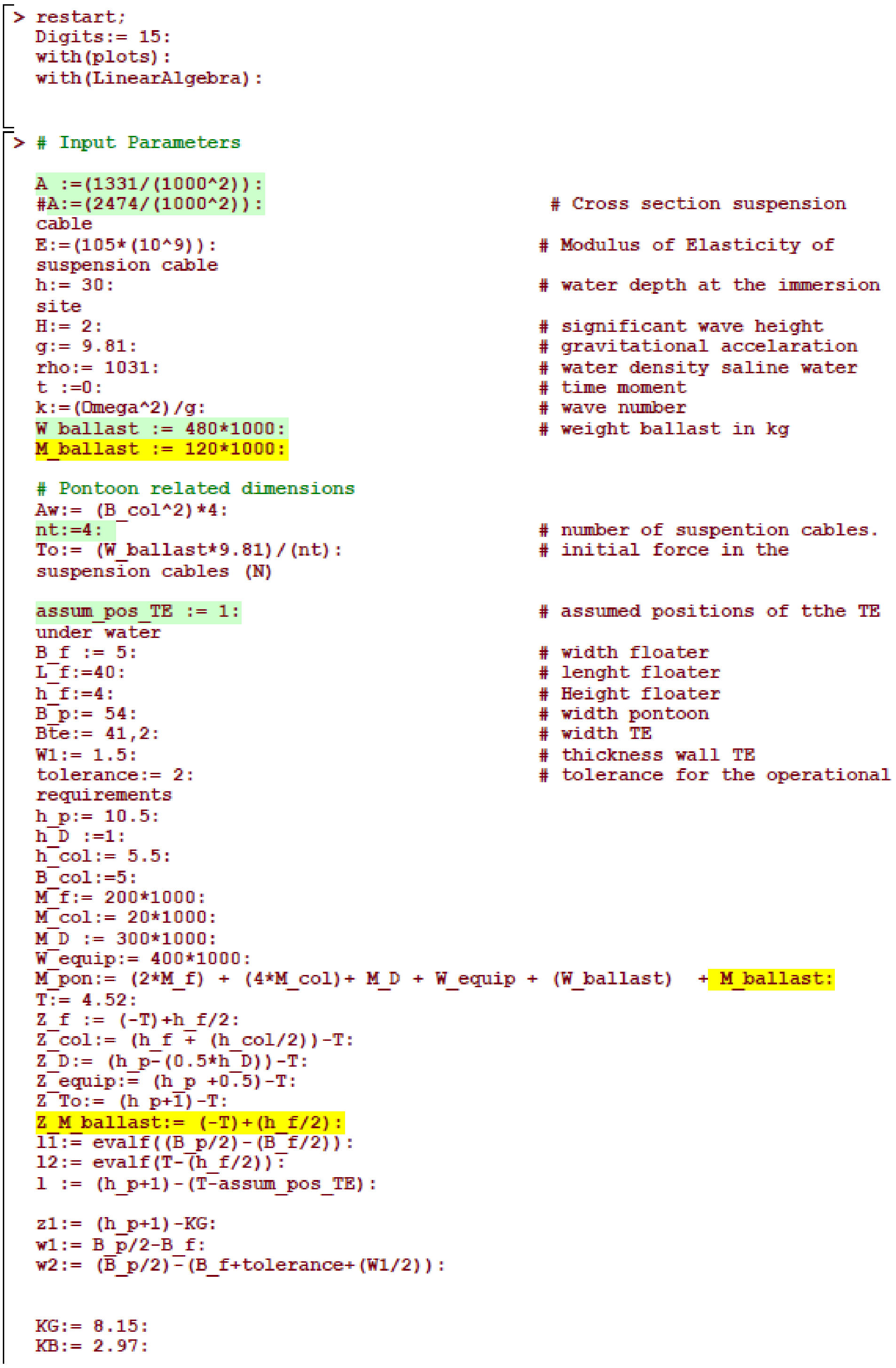
```
> ss41:= [seq([t||i,Upp[i][1]], i = 1 ..ndt)]:
  ss42:= [seq([t||i,Upp[i][2]], i = 1 ..ndt)]:
  ss43:= [seq([t||i,Upp[i][3]], i = 1 ..ndt)]:
  ss44:= [seq([t||i,Upp[i][4]], i = 1 ..ndt)]:
  ss45:= [seq([t||i,Upp[i][5]], i = 1 ..ndt)]:
  ss46:= [seq([t||i,Upp[i][6]], i = 1 ..ndt)]:
```

```
> P41:= plot(ss41,style=line,color=red, labels=["time [s]","[m]"],title=
  "Accelaration for Surge degree of freedom", labeldirections=[horizontal,
  vertical],font=[Calibri,1,10],labelfont=[Calibri,1,10]):
  P42:=plot(ss42,style=line,color=blue, labels=["time [s]","[m]"], title=
  "Accelaration for Sway degree of freedom", labeldirections=[horizontal,
  vertical],font=[Calibri,1,10],labelfont=[Calibri,1,10]):
  P43:= plot(ss43,style=line,color=green, labels=["time [s]","[m]"],title=
  "Accelaration for Heave degree of freedom", labeldirections=[horizontal,
  vertical],font=[Calibri,1,10],labelfont=[Calibri,1,10]):
  P44:= plot(ss44,style=line,color=purple, labels=["time [s]","[degree]"],
  title="Accelaration for Roll degree of freedom", labeldirections=
  [horizontal,vertical],font=[Calibri,1,10],labelfont=[Calibri,1,10]):
  P45:= plot(ss45,style=line,color=black, labels=["time [s]","[degree]"],
  title="Accelaration for Pitch degree of freedom ", labeldirections=
  [horizontal,vertical],font=[Calibri,1,10],labelfont=[Calibri,1,10]):
  P46:= plot(ss46, style = line, color = brown, labels=["time [s]","[degree]
  "], title = "Accelaration for Yaw degree of freedom", labeldirections =
  [horizontal, vertical], font = [Calibri, 1, 10], labelfont = [Calibri, 1,
  10]):
```

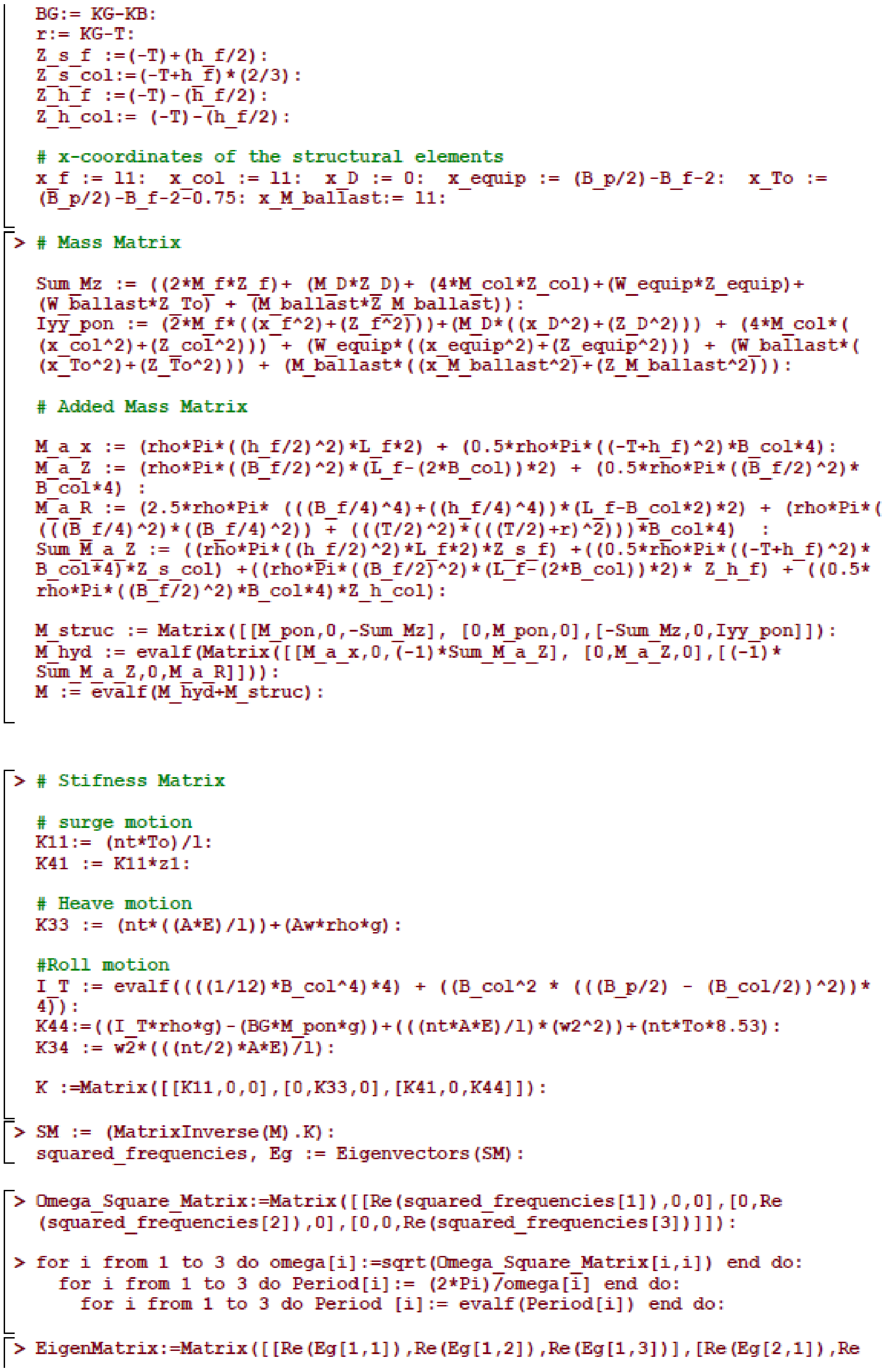
```
> #P41; P42; P43; P44; P45; P46;
```

7.6 CALCULATION FILES RAO'S PONTOONS

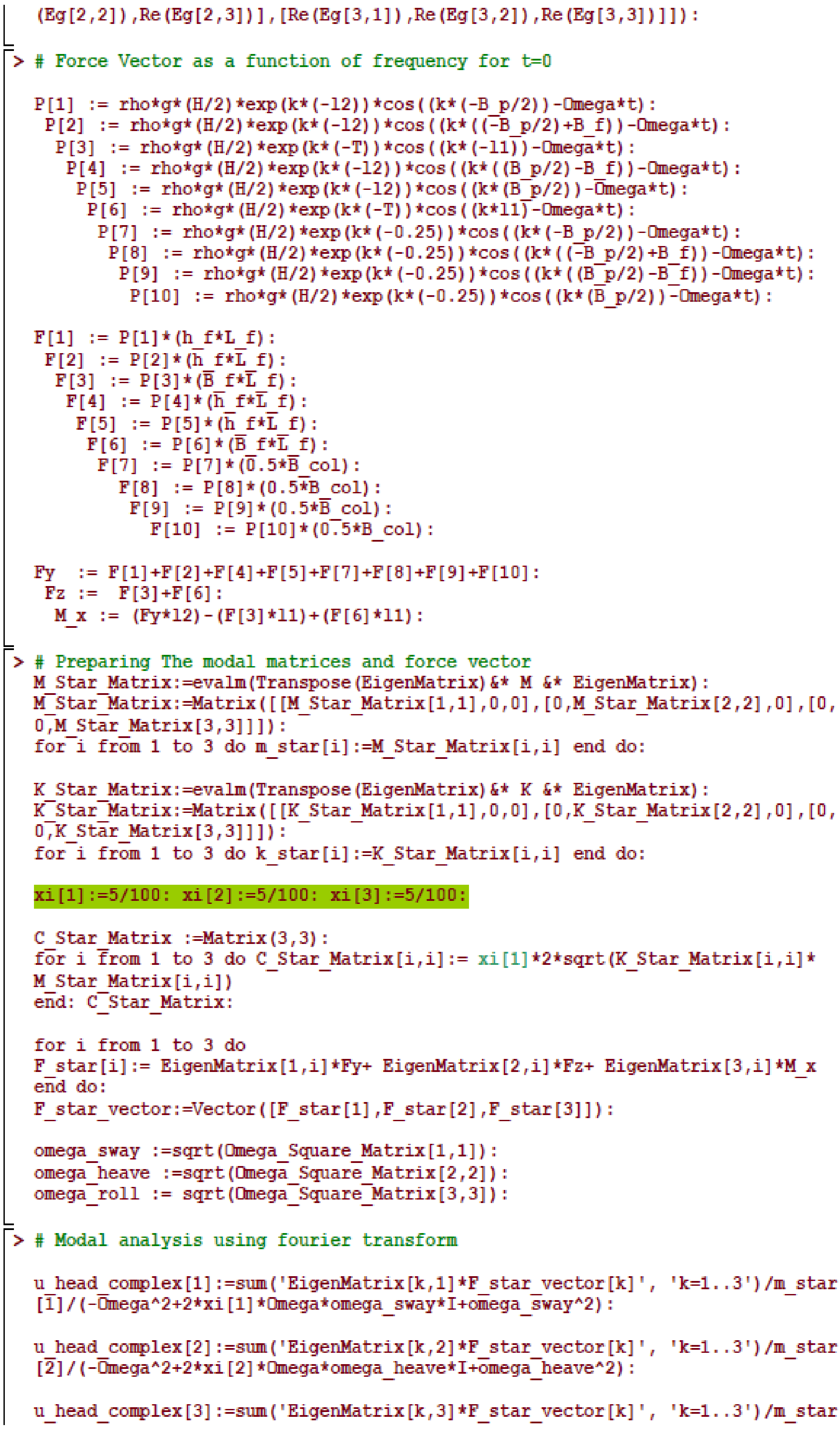
Calculation file RAO's Semi-submersible pontoon



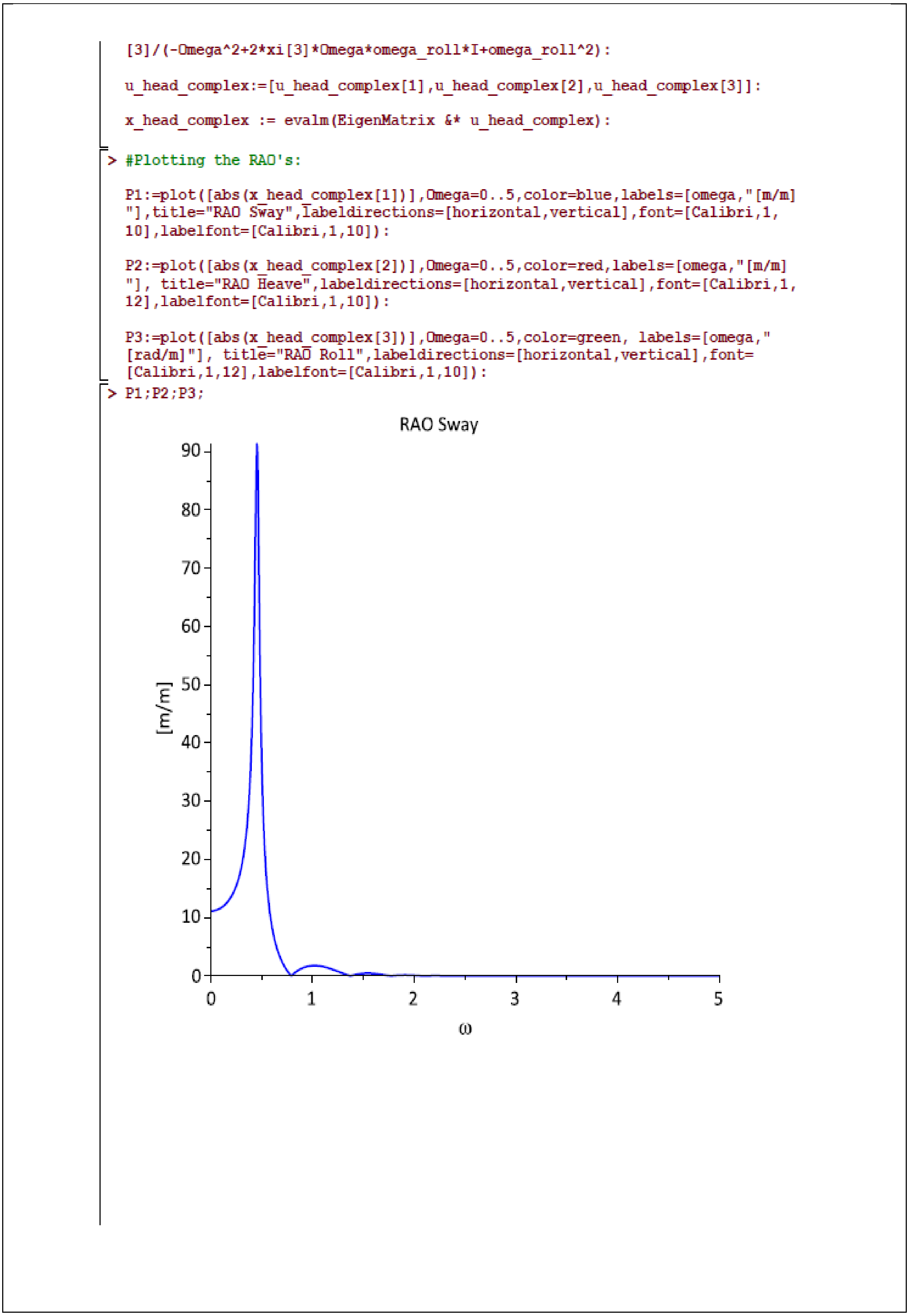
```
> restart;
  Digits:= 15:
  with(plots):
  with(LinearAlgebra):

> # Input Parameters

  A :=(1331/(1000^2)):
  #A:=(2474/(1000^2)):                          # Cross section suspension
  cable
  E:=(105*(10^9)):                             # Modulus of Elasticity of
  suspension cable
  h:= 30:                                      # water depth at the immersion
  site
  H:= 2:                                       # significant wave height
  g:= 9.81:                                    # gravitational accelaration
  rho:= 1031:                                  # water density saline water
  t :=0:                                       # time moment
  k:=(Omega^2)/g:                              # wave number
  W_ballast := 480*1000:                       # weight ballast in kg
  M_ballast := 120*1000:

  # Pontoon related dimensions
  Aw:= (B_col^2)*4:
  nt:=4:                                       # number of suspention cables.
  To:= (W_ballast*9.81)/(nt):                  # initial force in the
  suspension cables (N)

  assum_pos_TE := 1:                           # assumed positions of tthe TE
  under water
  B_f := 5:                                    # width floater
  L_f:=40:                                     # lenght floater
  h_f:=4:                                      # Height floater
  B_p:= 54:                                    # width pontoon
  Bte:= 41,2:                                  # width TE
  W1:= 1.5:                                    # thickness wall TE
  tolerance:= 2:                               # tolerance for the operational
  requirements
  h_p:= 10.5:
  h_D :=1:
  h_col:= 5.5:
  B_col:=5:
  M_f:= 200*1000:
  M_col:= 20*1000:
  M_D := 300*1000:
  W_equip:= 400*1000:
  M_pon:= (2*M_f) + (4*M_col)+ M_D + W_equip + (W_ballast)  + M_ballast:
  T:= 4.52:
  Z_f := (-T)+h_f/2:
  Z_col:= (h_f + (h_col/2))-T:
  Z_D:= (h_p-(0.5*h_D))-T:
  Z_equip:= (h_p +0.5)-T:
  Z_To:= (h_p+1)-T:
  Z_M_ballast:= (-T)+(h_f/2):
  l1:= evalf((B_p/2)-(B_f/2)):
  l2:= evalf(T-(h_f/2)):
  l := (h_p+1)-(T-assum_pos_TE):

  z1:= (h_p+1)-KG:
  w1:= B_p/2-B_f:
  w2:= (B_p/2)-(B_f+tolerance+(W1/2)):


  KG:= 8.15:
  KB:= 2.97:
```



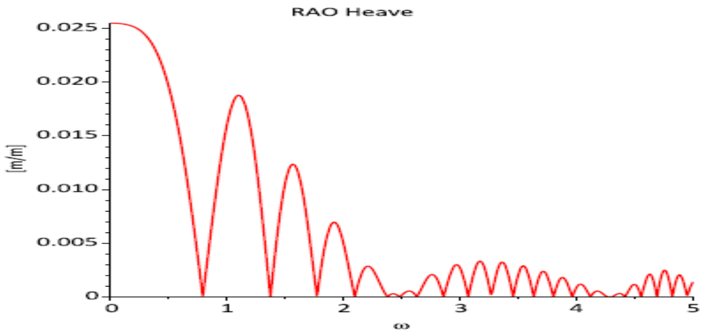
```
  BG:= KG-KB:
  r:= KG-T:
  Z_s_f :=(-T)+(h_f/2):
  Z_s_col:=(-T+h_f)*(2/3):
  Z_h_f :=(-T)-(h_f/2):
  Z_h_col:= (-T)-(h_f/2):

  # x-coordinates of the structural elements
  x_f := 11:  x_col := 11:  x_D := 0:  x_equip := (B_p/2)-B_f-2:  x_To :=
  (B_p/2)-B_f-2-0.75: x_M_ballast:= 11:

> # Mass Matrix

  Sum_Mz := ((2*M_f*Z_f)+ (M_D*Z_D)+ (4*M_col*Z_col)+(W_equip*Z_equip)+
  (W_ballast*Z_To) + (M_ballast*Z_M_ballast)):
  Iyy_pon := (2*M_f*((x_f^2)+(Z_f^2)))+(M_D*((x_D^2)+(Z_D^2))) + (4*M_col*(
  (x_col^2)+(Z_col^2))) + (W_equip*((x_equip^2)+(Z_equip^2))) + (W_ballast*(
  (x_To^2)+(Z_To^2))) + (M_ballast*((x_M_ballast^2)+(Z_M_ballast^2))):

  # Added Mass Matrix

  M_a_x := (rho*Pi*((h_f/2)^2)*L_f*2) + (0.5*rho*Pi*((-T+h_f)^2)*B_col*4):
  M_a_Z := (rho*Pi*((B_f/2)^2)*(L_f-(2*B_col))*2) + (0.5*rho*Pi*((B_f/2)^2)*
  B_col*4) :
  M_a_R := (2.5*rho*Pi* (((B_f/4)^4)+((h_f/4)^4))*(L_f-B_col*2)*2) + (rho*Pi*(
  (((B_f/4)^2)*((B_f/4)^2)) + (((T/2)^2)*(((T/2)+r)^2)))*B_col*4)   :
  Sum_M_a_Z := ((rho*Pi*((h_f/2)^2)*L_f*2)*Z_s_f) +((0.5*rho*Pi*((-T+h_f)^2)*
  B_col*4)*Z_s_col) +((rho*Pi*((B_f/2)^2)*(L_f-(2*B_col))*2)* Z_h_f) + ((0.5*
  rho*Pi*((B_f/2)^2)*B_col*4)*Z_h_col):

  M_struc := Matrix([[M_pon,0,-Sum_Mz], [0,M_pon,0],[-Sum_Mz,0,Iyy_pon]]):
  M_hyd := evalf(Matrix([[M_a_x,0,(-1)*Sum_M_a_Z], [0,M_a_Z,0],[(-1)*
  Sum_M_a_Z,0,M_a_R]])):
  M := evalf(M_hyd+M_struc):


> # Stifness Matrix

  # surge motion
  K11:= (nt*To)/l:
  K41 := K11*z1:

  # Heave motion
  K33 := (nt*((A*E)/l))+(Aw*rho*g):

  #Roll motion
  I_T := evalf((((1/12)*B_col^4)*4) + ((B_col^2 * (((B_p/2) - (B_col/2))^2))*
  4)):
  K44:=((I_T*rho*g)-(BG*M_pon*g))+(((nt*A*E)/l)*(w2^2))+(nt*To*8.53):
  K34 := w2*(((nt/2)*A*E)/l):

  K :=Matrix([[K11,0,0],[0,K33,0],[K41,0,K44]]):
> SM := (MatrixInverse(M).K):
  squared_frequencies, Eg := Eigenvectors(SM):

> Omega_Square_Matrix:=Matrix([[Re(squared_frequencies[1]),0,0],[0,Re
  (squared_frequencies[2]),0],[0,0,Re(squared_frequencies[3])]]):

> for i from 1 to 3 do omega[i]:=sqrt(Omega_Square_Matrix[i,i]) end do:
    for i from 1 to 3 do Period[i]:= (2*Pi)/omega[i] end do:
      for i from 1 to 3 do Period [i]:= evalf(Period[i]) end do:

> EigenMatrix:=Matrix([[Re(Eg[1,1]),Re(Eg[1,2]),Re(Eg[1,3])],[Re(Eg[2,1]),Re
```

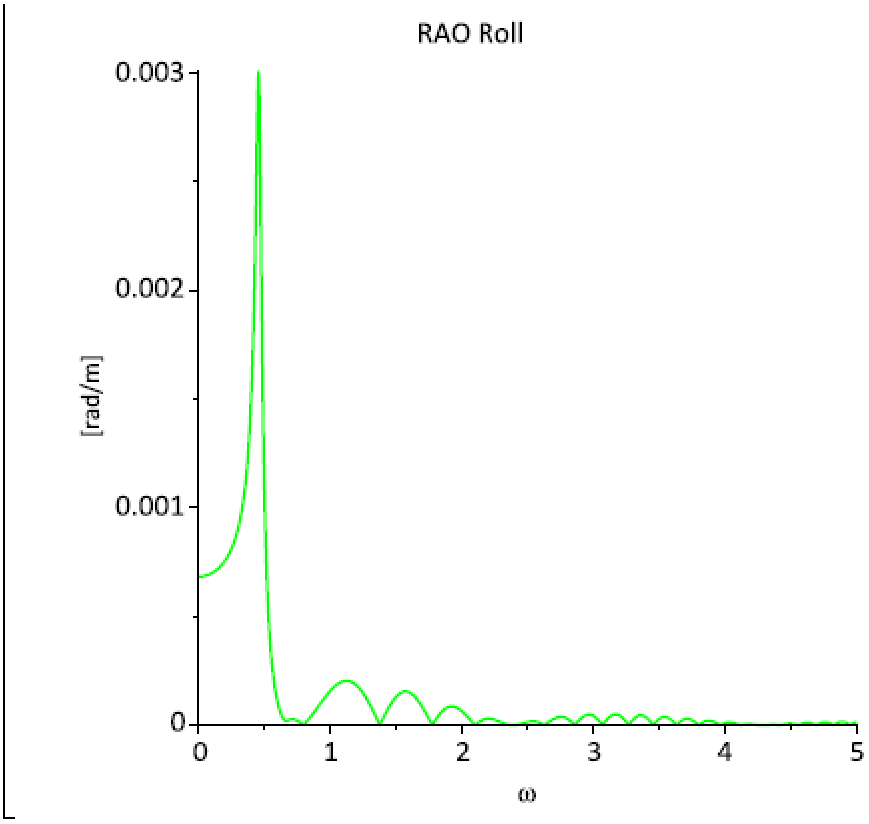


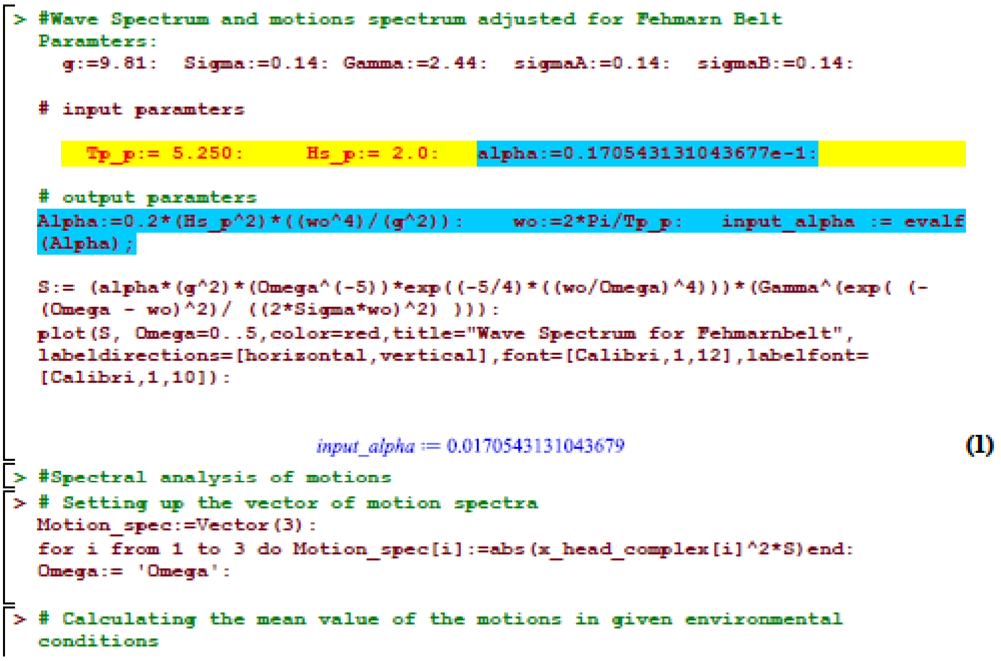
```
 (Eg[2,2]),Re(Eg[2,3])],[Re(Eg[3,1]),Re(Eg[3,2]),Re(Eg[3,3])]]):

> # Force Vector as a function of frequency for t=0

 P[1]  := rho*g*(H/2)*exp(k*(-l2))*cos((k*(-B_p/2))-Omega*t):
  P[2]  := rho*g*(H/2)*exp(k*(-l2))*cos((k*((-B_p/2)+B_f))-Omega*t):
   P[3]  := rho*g*(H/2)*exp(k*(-T))*cos((k*(-l1))-Omega*t):
    P[4]  := rho*g*(H/2)*exp(k*(-l2))*cos((k*((B_p/2)-B_f))-Omega*t):
     P[5]  := rho*g*(H/2)*exp(k*(-l2))*cos((k*(B_p/2))-Omega*t):
      P[6]  := rho*g*(H/2)*exp(k*(-T))*cos((k*l1)-Omega*t):
       P[7]  := rho*g*(H/2)*exp(k*(-0.25))*cos((k*(-B_p/2))-Omega*t):
        P[8]  := rho*g*(H/2)*exp(k*(-0.25))*cos((k*((-B_p/2)+B_f))-Omega*t):
         P[9]  := rho*g*(H/2)*exp(k*(-0.25))*cos((k*((B_p/2)-B_f))-Omega*t):
          P[10]  := rho*g*(H/2)*exp(k*(-0.25))*cos((k*(B_p/2))-Omega*t):

 F[1]  := P[1]*(h_f*L_f):
  F[2]  := P[2]*(h_f*L_f):
   F[3]  := P[3]*(B_f*L_f):
    F[4]  := P[4]*(h_f*L_f):
     F[5]  := P[5]*(h_f*L_f):
      F[6]  := P[6]*(B_f*L_f):
       F[7]  := P[7]*(0.5*B_col):
         F[8]  := P[8]*(0.5*B_col):
          F[9]  := P[9]*(0.5*B_col):
            F[10]  := P[10]*(0.5*B_col):

 Fy  := F[1]+F[2]+F[4]+F[5]+F[7]+F[8]+F[9]+F[10]:
  Fz :=  F[3]+F[6]:
   M_x := (Fy*l2)-(F[3]*l1)+(F[6]*l1):

> # Preparing The modal matrices and force vector
 M_Star_Matrix:=evalm(Transpose(EigenMatrix)&* M &* EigenMatrix):
 M_Star_Matrix:=Matrix([[M_Star_Matrix[1,1],0,0],[0,M_Star_Matrix[2,2],0],[0,
 0,M_Star_Matrix[3,3]]]):
 for i from 1 to 3 do m_star[i]:=M_Star_Matrix[i,i] end do:

 K_Star_Matrix:=evalm(Transpose(EigenMatrix)&* K &* EigenMatrix):
 K_Star_Matrix:=Matrix([[K_Star_Matrix[1,1],0,0],[0,K_Star_Matrix[2,2],0],[0,
 0,K_Star_Matrix[3,3]]]):
 for i from 1 to 3 do k_star[i]:=K_Star_Matrix[i,i] end do:

 xi[1]:=5/100: xi[2]:=5/100: xi[3]:=5/100:

 C_Star_Matrix :=Matrix(3,3):
 for i from 1 to 3 do C_Star_Matrix[i,i]:= xi[1]*2*sqrt(K_Star_Matrix[i,i]*
 M_Star_Matrix[i,i])
 end: C_Star_Matrix:

 for i from 1 to 3 do
 F_star[i]:= EigenMatrix[1,i]*Fy+ EigenMatrix[2,i]*Fz+ EigenMatrix[3,i]*M_x
 end do:
 F_star_vector:=Vector([F_star[1],F_star[2],F_star[3]]):

 omega_sway :=sqrt(Omega_Square_Matrix[1,1]):
 omega_heave :=sqrt(Omega_Square_Matrix[2,2]):
 omega_roll := sqrt(Omega_Square_Matrix[3,3]):

> # Modal analysis using fourier transform

 u_head_complex[1]:=sum('EigenMatrix[k,1]*F_star_vector[k]', 'k=1..3')/m_star
 [1]/(-Omega^2+2*xi[1]*Omega*omega_sway*I+omega_sway^2):

 u_head_complex[2]:=sum('EigenMatrix[k,2]*F_star_vector[k]', 'k=1..3')/m_star
 [2]/(-Omega^2+2*xi[2]*Omega*omega_heave*I+omega_heave^2):

 u_head_complex[3]:=sum('EigenMatrix[k,3]*F_star_vector[k]', 'k=1..3')/m_star
```



```
  [3]/(-Omega^2+2*xi[3]*Omega*omega_roll*I+omega_roll^2):

  u_head_complex:=[u_head_complex[1],u_head_complex[2],u_head_complex[3]]:

  x_head_complex := evalm(EigenMatrix &* u_head_complex):

> #Plotting the RAO's:

  P1:=plot([abs(x_head_complex[1])],Omega=0..5,color=blue,labels=[omega,"[m/m]
  "],title="RAO Sway",labeldirections=[horizontal,vertical],font=[Calibri,1,
  10],labelfont=[Calibri,1,10]):

  P2:=plot([abs(x_head_complex[2])],Omega=0..5,color=red,labels=[omega,"[m/m]
  "], title="RAO Heave",labeldirections=[horizontal,vertical],font=[Calibri,1,
  12],labelfont=[Calibri,1,10]):

  P3:=plot([abs(x_head_complex[3])],Omega=0..5,color=green, labels=[omega,"
  [rad/m]"], title="RAO Roll",labeldirections=[horizontal,vertical],font=
  [Calibri,1,12],labelfont=[Calibri,1,10]):
> P1;P2;P3;
```


RAO Heave
0.025
0.020
0.015
0.010
0.005
0
[m/m]
0
1
2
3
4
5
ω

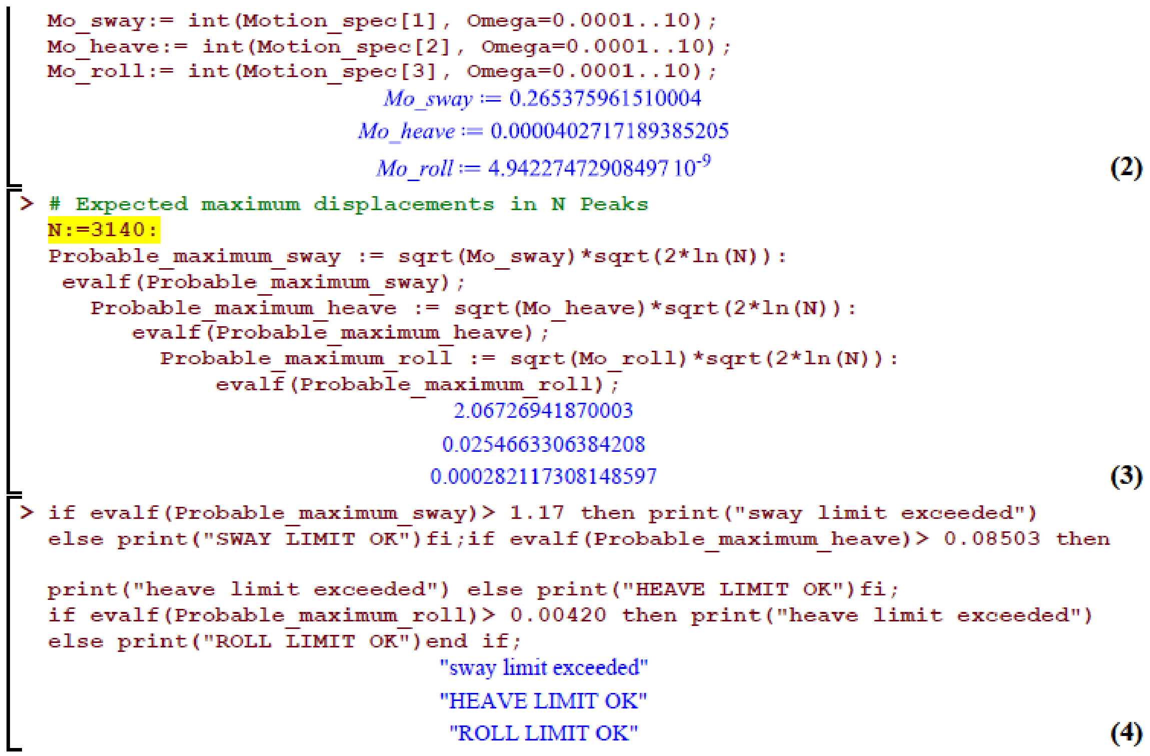




```
> #Wave Spectrum and motions spectrum adjusted for Fehmarn Belt
  Paramters:
    g:=9.81:  Sigma:=0.14: Gamma:=2.44:  sigmaA:=0.14:  sigmaB:=0.14:

  # input paramters

     Tp_p:= 5.250:      Hs_p:= 2.0:    alpha:=0.170543131043677e-1:

  # output paramters
  Alpha:=0.2*(Hs_p^2)*((wo^4)/(g^2)):     wo:=2*Pi/Tp_p:   input_alpha := evalf
  (Alpha);

  S:= (alpha*(g^2)*(Omega^(-5))*exp((-5/4)*((wo/Omega)^4)))*(Gamma^(exp( (-
  (Omega - wo)^2)/ ((2*Sigma*wo)^2) ))):
  plot(S, Omega=0..5,color=red,title="Wave Spectrum for Fehmarnbelt",
  labeldirections=[horizontal,vertical],font=[Calibri,1,12],labelfont=
  [Calibri,1,10]):
```

$$input\_alpha := 0.01705431310436779 \qquad (1)$$

```
> #Spectral analysis of motions
> # Setting up the vector of motion spectra
  Motion_spec:=Vector(3):
  for i from 1 to 3 do Motion_spec[i]:=abs(x_head_complex[i]^2*S)end:
  Omega:= 'Omega':

> # Calculating the mean value of the motions in given environmental
  conditions
```



```
Mo_sway:= int(Motion_spec[1], Omega=0.0001..10);
Mo_heave:= int(Motion_spec[2], Omega=0.0001..10);
Mo_roll:= int(Motion_spec[3], Omega=0.0001..10);
```

$$Mo\_sway := 0.265375961510004$$

$$Mo\_heave := 0.000040271718938 5205$$

$$Mo\_roll := 4.94227472908497\ 10^{-9} \qquad (2)$$

```
> # Expected maximum displacements in N Peaks
N:=3140:
Probable_maximum_sway := sqrt(Mo_sway)*sqrt(2*ln(N)):
 evalf(Probable_maximum_sway);
   Probable_maximum_heave := sqrt(Mo_heave)*sqrt(2*ln(N)):
      evalf(Probable_maximum_heave);
        Probable_maximum_roll := sqrt(Mo_roll)*sqrt(2*ln(N)):
            evalf(Probable_maximum_roll);
```

2.06726941870003

0.0254663306384208

0.000282117308148597 (3)

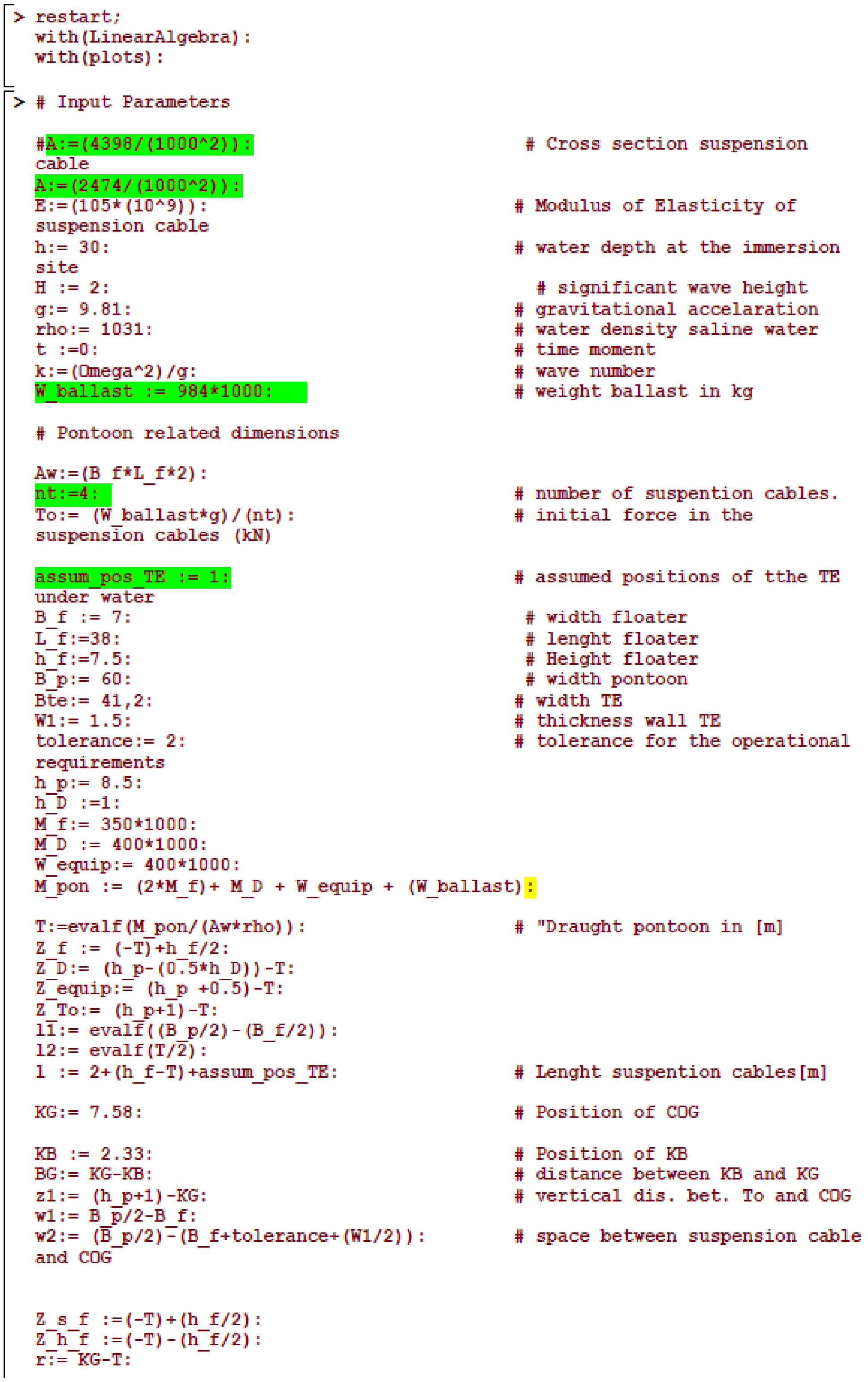
```
> if evalf(Probable_maximum_sway)> 1.17 then print("sway limit exceeded")
else print("SWAY LIMIT OK")fi;if evalf(Probable_maximum_heave)> 0.08503 then

print("heave limit exceeded") else print("HEAVE LIMIT OK")fi;
if evalf(Probable_maximum_roll)> 0.00420 then print("heave limit exceeded")
else print("ROLL LIMIT OK")end if;
```

"sway limit exceeded"

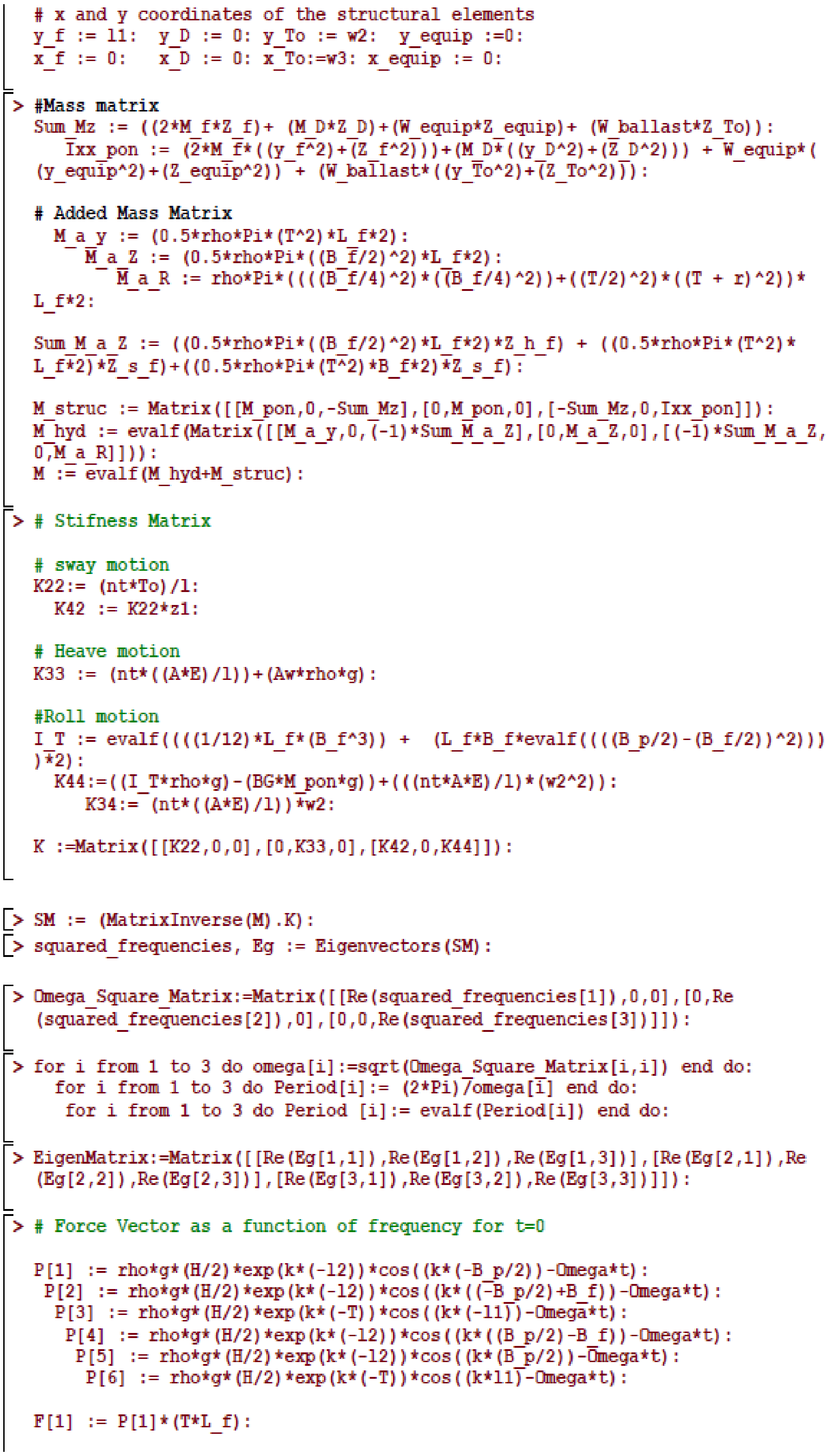
"HEAVE LIMIT OK"

"ROLL LIMIT OK" (4)

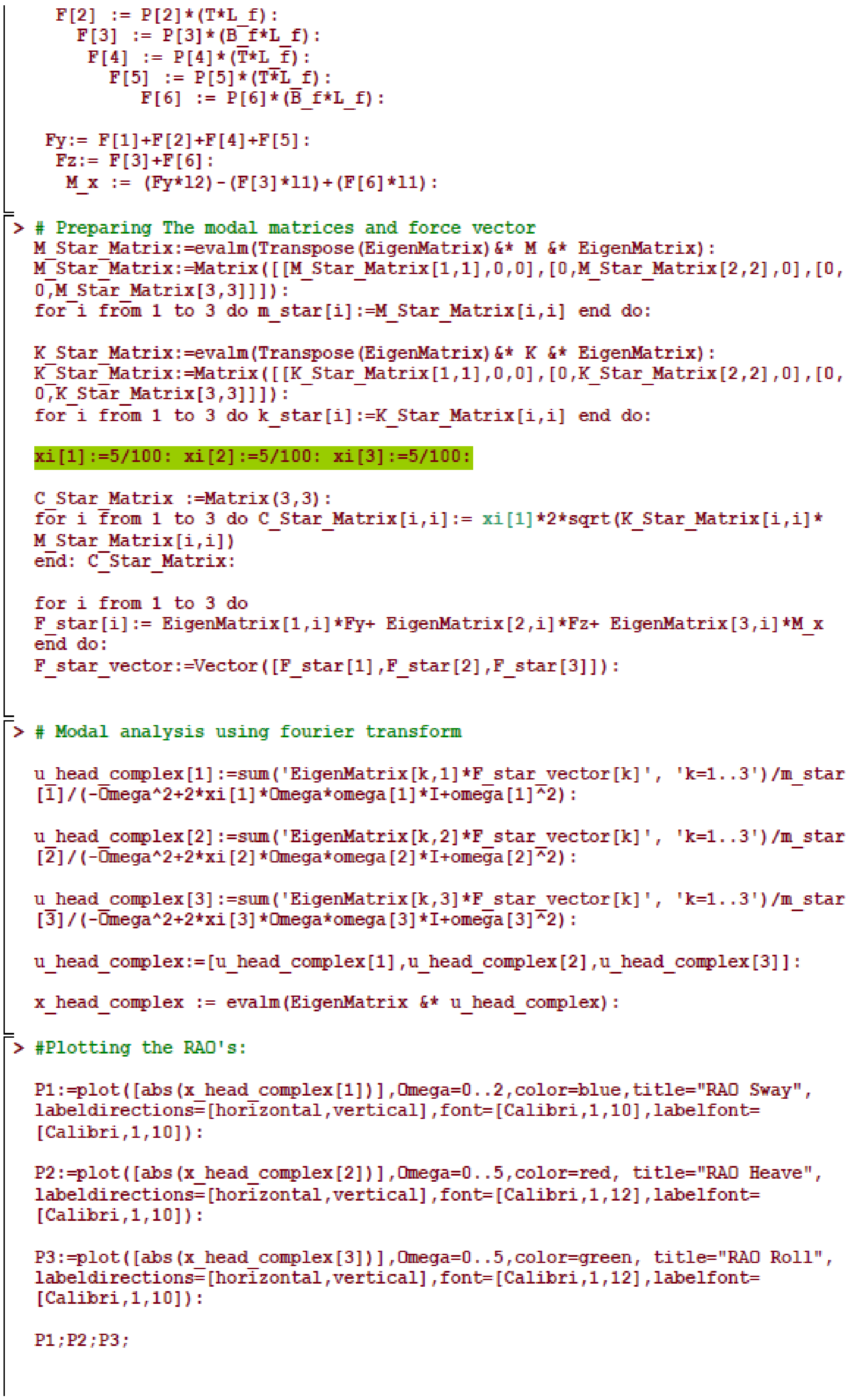
Calculation files RAO's Catamaran pontoon



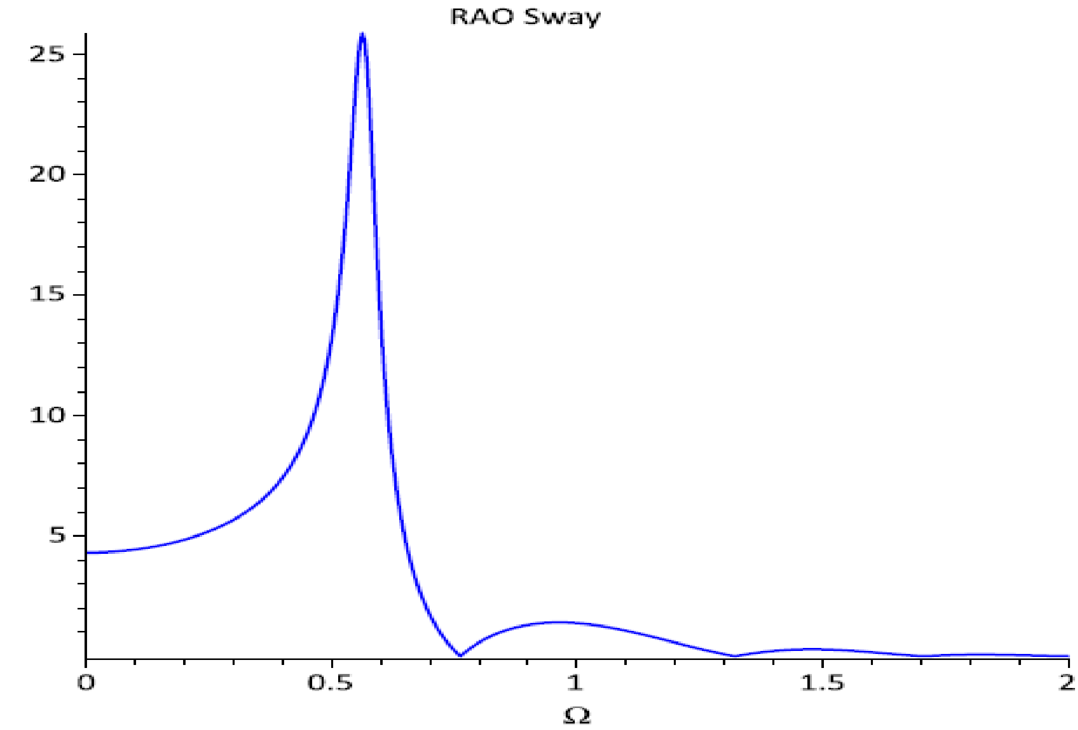
```
> restart;
  with(LinearAlgebra):
  with(plots):

> # Input Parameters

  #A:=(4398/(1000^2)):                          # Cross section suspension
  cable
  A:=(2474/(1000^2)):
  E:=(105*(10^9)):                             # Modulus of Elasticity of
  suspension cable
  h:= 30:                                      # water depth at the immersion
  site
  H := 2:                                        # significant wave height
  g:= 9.81:                                    # gravitational accelaration
  rho:= 1031:                                  # water density saline water
  t :=0:                                       # time moment
  k:=(Omega^2)/g:                              # wave number
  W_ballast := 984*1000:                       # weight ballast in kg

  # Pontoon related dimensions

  Aw:=(B_f*L_f*2):
  nt:=4:                                       # number of suspention cables.
  To:= (W_ballast*g)/(nt):                     # initial force in the
  suspension cables (kN)

  assum_pos_TE := 1:                           # assumed positions of tthe TE
  under water
  B_f := 7:                                     # width floater
  L_f:=38:                                      # lenght floater
  h_f:=7.5:                                     # Height floater
  B_p:= 60:                                     # width pontoon
  Bte:= 41,2:                                  # width TE
  W1:= 1.5:                                    # thickness wall TE
  tolerance:= 2:                               # tolerance for the operational
  requirements
  h_p:= 8.5:
  h_D :=1:
  M_f:= 350*1000:
  M_D := 400*1000:
  W_equip:= 400*1000:
  M_pon := (2*M_f)+ M_D + W_equip + (W_ballast):

  T:=evalf(M_pon/(Aw*rho)):                    # "Draught pontoon in [m]
  Z_f := (-T)+h_f/2:
  Z_D:= (h_p-(0.5*h_D))-T:
  Z_equip:= (h_p +0.5)-T:
  Z_To:= (h_p+1)-T:
  l1:= evalf((B_p/2)-(B_f/2)):
  l2:= evalf(T/2):
  l := 2+(h_f-T)+assum_pos_TE:                 # Lenght suspention cables[m]

  KG:= 7.58:                                   # Position of COG

  KB := 2.33:                                  # Position of KB
  BG:= KG-KB:                                  # distance between KB and KG
  z1:= (h_p+1)-KG:                             # vertical dis. bet. To and COG
  w1:= B_p/2-B_f:
  w2:= (B_p/2)-(B_f+tolerance+(W1/2)):         # space between suspension cable
  and COG


  Z_s_f :=(-T)+(h_f/2):
  Z_h_f :=(-T)-(h_f/2):
  r:= KG-T:
```

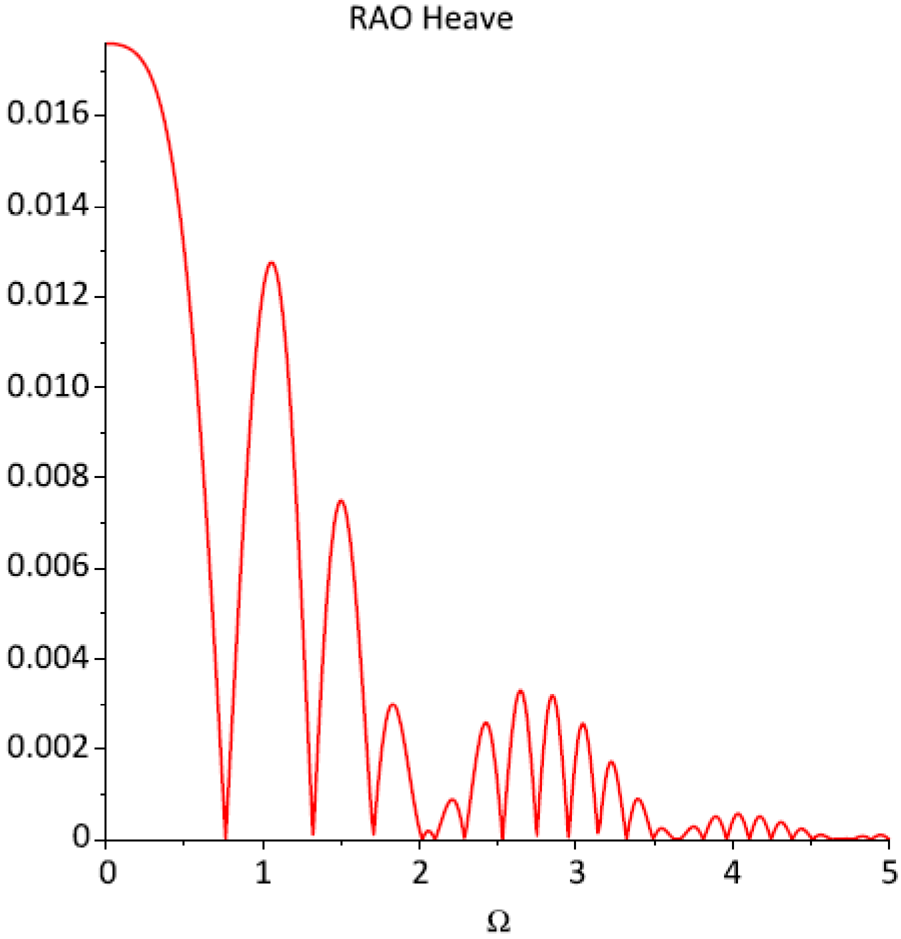


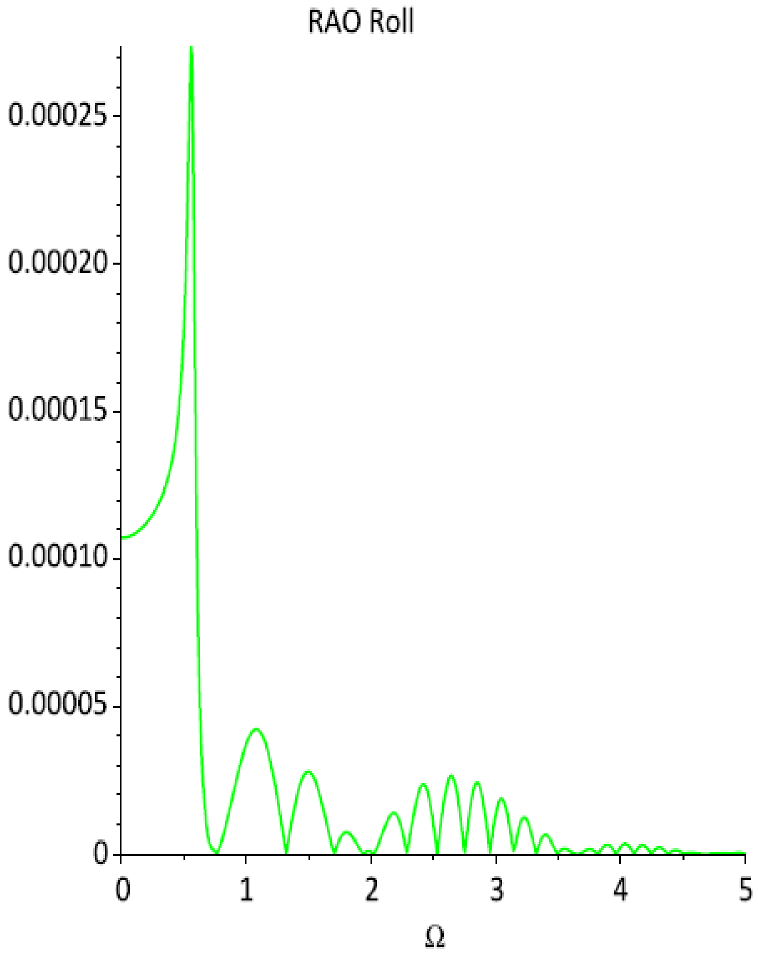
```
  # x and y coordinates of the structural elements
  y_f := l1:  y_D := 0: y_To := w2:  y_equip :=0:
  x_f := 0:   x_D := 0: x_To:=w3: x_equip := 0:

> #Mass matrix
  Sum_Mz := ((2*M_f*Z_f)+ (M_D*Z_D)+(W_equip*Z_equip)+ (W_ballast*Z_To)):
     Ixx_pon := (2*M_f*((y_f^2)+(Z_f^2)))+(M_D*((y_D^2)+(Z_D^2))) + W_equip*(
  (y_equip^2)+(Z_equip^2)) + (W_ballast*((y_To^2)+(Z_To^2))):

  # Added Mass Matrix
    M_a_y := (0.5*rho*Pi*(T^2)*L_f*2):
       M_a_Z := (0.5*rho*Pi*((B_f/2)^2)*L_f*2):
          M_a_R := rho*Pi*((((B_f/4)^2)*((B_f/4)^2))+((T/2)^2)*((T + r)^2))*
  L_f*2:

  Sum_M_a_Z := ((0.5*rho*Pi*((B_f/2)^2)*L_f*2)*Z_h_f) + ((0.5*rho*Pi*(T^2)*
  L_f*2)*Z_s_f)+((0.5*rho*Pi*(T^2)*B_f*2)*Z_s_f):

  M_struc := Matrix([[M_pon,0,-Sum_Mz],[0,M_pon,0],[-Sum_Mz,0,Ixx_pon]]):
  M_hyd := evalf(Matrix([[M_a_y,0,(-1)*Sum_M_a_Z],[0,M_a_Z,0],[(-1)*Sum_M_a_Z,
  0,M_a_R]])):
  M := evalf(M_hyd+M_struc):

> # Stifness Matrix

  # sway motion
  K22:= (nt*To)/l:
    K42 := K22*z1:

  # Heave motion
  K33 := (nt*((A*E)/l))+(Aw*rho*g):

  #Roll motion
  I_T := evalf((((1/12)*L_f*(B_f^3)) +  (L_f*B_f*evalf((((B_p/2)-(B_f/2))^2)))
  )*2):
    K44:=((I_T*rho*g)-(BG*M_pon*g))+(((nt*A*E)/l)*(w2^2)):
       K34:= (nt*((A*E)/l))*w2:

  K :=Matrix([[K22,0,0],[0,K33,0],[K42,0,K44]]):


> SM := (MatrixInverse(M).K):
> squared_frequencies, Eg := Eigenvectors(SM):

> Omega_Square_Matrix:=Matrix([[Re(squared_frequencies[1]),0,0],[0,Re
  (squared_frequencies[2]),0],[0,0,Re(squared_frequencies[3])]]):

> for i from 1 to 3 do omega[i]:=sqrt(Omega_Square_Matrix[i,i]) end do:
   for i from 1 to 3 do Period[i]:= (2*Pi)/omega[i] end do:
     for i from 1 to 3 do Period [i]:= evalf(Period[i]) end do:

> EigenMatrix:=Matrix([[Re(Eg[1,1]),Re(Eg[1,2]),Re(Eg[1,3])],[Re(Eg[2,1]),Re
  (Eg[2,2]),Re(Eg[2,3])],[Re(Eg[3,1]),Re(Eg[3,2]),Re(Eg[3,3])]]):

> # Force Vector as a function of frequency for t=0

  P[1] := rho*g*(H/2)*exp(k*(-l2))*cos((k*(-B_p/2))-Omega*t):
   P[2] := rho*g*(H/2)*exp(k*(-l2))*cos((k*((-B_p/2)+B_f))-Omega*t):
    P[3] := rho*g*(H/2)*exp(k*(-T))*cos((k*(-l1))-Omega*t):
     P[4] := rho*g*(H/2)*exp(k*(-l2))*cos((k*((B_p/2)-B_f))-Omega*t):
      P[5] := rho*g*(H/2)*exp(k*(-l2))*cos((k*(B_p/2))-Omega*t):
       P[6] := rho*g*(H/2)*exp(k*(-T))*cos((k*l1)-Omega*t):

  F[1] := P[1]*(T*L_f):
```



```
  F[2]  := P[2]*(T*L_f):
    F[3]  := P[3]*(B_f*L_f):
     F[4]  := P[4]*(T*L_f):
       F[5]  := P[5]*(T*L_f):
          F[6]  := P[6]*(B_f*L_f):

 Fy:= F[1]+F[2]+F[4]+F[5]:
  Fz:= F[3]+F[6]:
   M_x := (Fy*l2)-(F[3]*l1)+(F[6]*l1):

> # Preparing The modal matrices and force vector
  M_Star_Matrix:=evalm(Transpose(EigenMatrix)&* M &* EigenMatrix):
  M_Star_Matrix:=Matrix([[M_Star_Matrix[1,1],0,0],[0,M_Star_Matrix[2,2],0],[0,
  0,M_Star_Matrix[3,3]]]):
  for i from 1 to 3 do m_star[i]:=M_Star_Matrix[i,i] end do:

  K_Star_Matrix:=evalm(Transpose(EigenMatrix)&* K &* EigenMatrix):
  K_Star_Matrix:=Matrix([[K_Star_Matrix[1,1],0,0],[0,K_Star_Matrix[2,2],0],[0,
  0,K_Star_Matrix[3,3]]]):
  for i from 1 to 3 do k_star[i]:=K_Star_Matrix[i,i] end do:

  xi[1]:=5/100: xi[2]:=5/100: xi[3]:=5/100:

  C_Star_Matrix :=Matrix(3,3):
  for i from 1 to 3 do C_Star_Matrix[i,i]:= xi[1]*2*sqrt(K_Star_Matrix[i,i]*
  M_Star_Matrix[i,i])
  end: C_Star_Matrix:

  for i from 1 to 3 do
  F_star[i]:= EigenMatrix[1,i]*Fy+ EigenMatrix[2,i]*Fz+ EigenMatrix[3,i]*M_x
  end do:
  F_star_vector:=Vector([F_star[1],F_star[2],F_star[3]]):


> # Modal analysis using fourier transform

  u_head_complex[1]:=sum('EigenMatrix[k,1]*F_star_vector[k]', 'k=1..3')/m_star
  [1]/(-Omega^2+2*xi[1]*Omega*omega[1]*I+omega[1]^2):

  u_head_complex[2]:=sum('EigenMatrix[k,2]*F_star_vector[k]', 'k=1..3')/m_star
  [2]/(-Omega^2+2*xi[2]*Omega*omega[2]*I+omega[2]^2):

  u_head_complex[3]:=sum('EigenMatrix[k,3]*F_star_vector[k]', 'k=1..3')/m_star
  [3]/(-Omega^2+2*xi[3]*Omega*omega[3]*I+omega[3]^2):

  u_head_complex:=[u_head_complex[1],u_head_complex[2],u_head_complex[3]]:

  x_head_complex := evalm(EigenMatrix &* u_head_complex):

> #Plotting the RAO's:

  P1:=plot([abs(x_head_complex[1])],Omega=0..2,color=blue,title="RAO Sway",
  labeldirections=[horizontal,vertical],font=[Calibri,1,10],labelfont=
  [Calibri,1,10]):

  P2:=plot([abs(x_head_complex[2])],Omega=0..5,color=red, title="RAO Heave",
  labeldirections=[horizontal,vertical],font=[Calibri,1,12],labelfont=
  [Calibri,1,10]):

  P3:=plot([abs(x_head_complex[3])],Omega=0..5,color=green, title="RAO Roll",
  labeldirections=[horizontal,vertical],font=[Calibri,1,12],labelfont=
  [Calibri,1,10]):

  P1;P2;P3;
```


RAO Sway
25
20
15
10
5
0
0.5
1
1.5
2
Ω

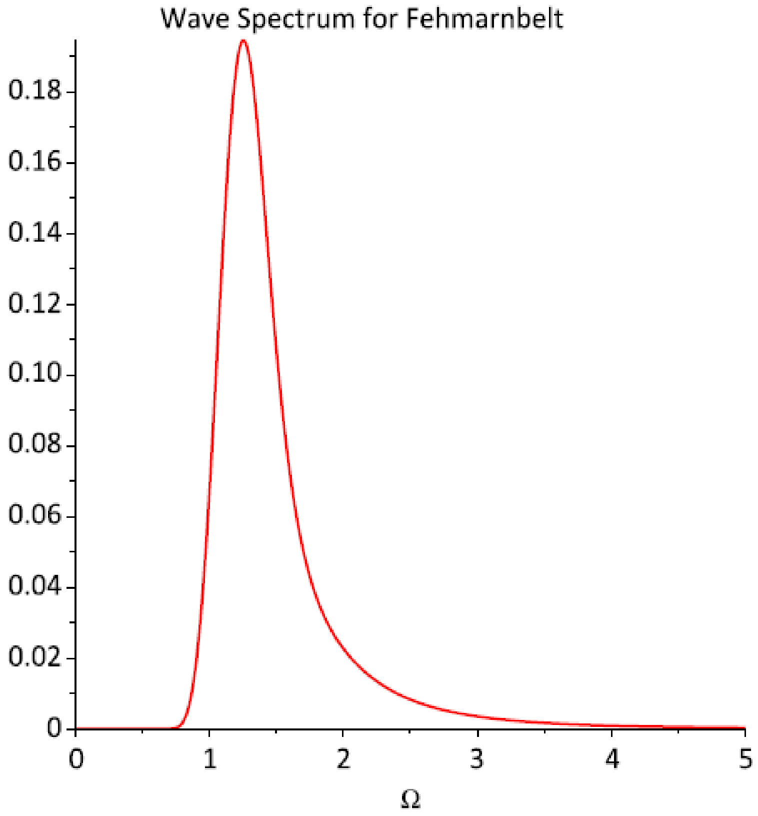

RAO Heave
0.016
0.014
0.012
0.010
0.008
0.006
0.004
0.002
0
0
1
2
3
4
5
Ω



```
> #Wave Spectrum and motions spectrum adjusted for Fehmarn Belt

> #Paramters:
   g:=9.81:   Sigma:=0.14: Gamma:=2.44:   sigmaA:=0.14:   sigmaB:=0.14:

  # input paramters

     Tp_p:= 5.0:      Hs_p:= 1.6:    alpha:=0.009061511828:

  # output paramters
  Alpha:=0.2*(Hs_p^2)*((wo^4)/(g^2)):    wo:=2*Pi/Tp_p:   input_alpha := evalf
  (Alpha);
```

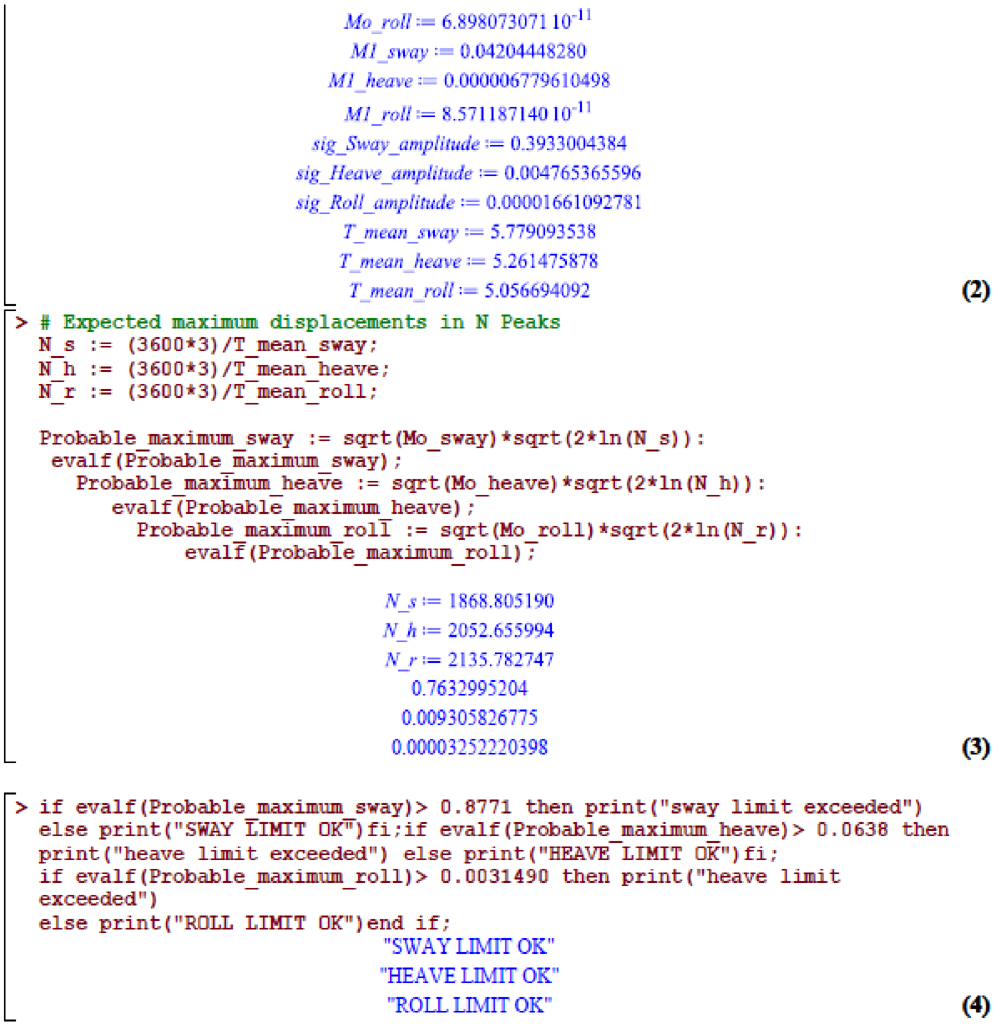
$$input\_alpha := 0.01326695949 \tag{1}$$

```
> S:= (alpha*(g^2)*(Omega^(-5))*exp((-5/4)*((wo/Omega)^4)))*(Gamma^(exp( (-
  (Omega - wo)^2)/ ((2*Sigma*wo)^2) ))):
  plot(S, Omega=0..5,color=red,title="Wave Spectrum for Fehmarnbelt",
  labeldirections=[horizontal,vertical],font=[Calibri,1,12],labelfont=
  [Calibri,1,10]);
```



```
> #Spectral analysis of motions
> # Setting up the vector of motion spectra
  Motion_spec:=Vector(3):
  for i from 1 to 3 do Motion_spec[i]:=abs(x_head_complex[i]^2*S)end:
  Omega:= 'Omega':

> # Calculating the mean value of the motions in given environmental
  conditions

  Mo_sway:= int(Motion_spec[1], Omega=0.0001..10);
  Mo_heave:= int(Motion_spec[2], Omega=0.0001..10);
  Mo_roll:= int(Motion_spec[3], Omega=0.0001..10);

  M1_sway:= int((Motion_spec[1]*Omega), Omega=0.0001..10);
  M1_heave:= int((Motion_spec[2]*Omega), Omega=0.0001..10);
  M1_roll:= int((Motion_spec[3]*Omega), Omega=0.0001..10);

  sig_Sway_amplitude := 2*sqrt(Mo_sway);
  sig_Heave_amplitude := 2*sqrt(Mo_heave);
  sig_Roll_amplitude := 2*sqrt(Mo_roll);

  T_mean_sway:= evalf((2*Pi)*(Mo_sway/M1_sway));
  T_mean_heave:= evalf((2*Pi)*(Mo_heave/M1_heave));
  T_mean_roll:= evalf((2*Pi)*(Mo_roll/M1_roll));
```

$$Mo\_sway := 0.03867130873$$
$$Mo\_heave := 0.000005677177315$$



```
                    Mo_roll := 6.898073071 10^-11
                    M1_sway := 0.04204448280
                    M1_heave := 0.000006779610498
                    M1_roll := 8.571187140 10^-11
                    sig_Sway_amplitude := 0.3933004384
                    sig_Heave_amplitude := 0.004765365596
                    sig_Roll_amplitude := 0.00001661092781
                    T_mean_sway := 5.779093538
                    T_mean_heave := 5.261475878
                    T_mean_roll := 5.056694092                    (2)
> # Expected maximum displacements in N Peaks
  N_s := (3600*3)/T_mean_sway;
  N_h := (3600*3)/T_mean_heave;
  N_r := (3600*3)/T_mean_roll;

  Probable_maximum_sway := sqrt(Mo_sway)*sqrt(2*ln(N_s)):
   evalf(Probable_maximum_sway);
     Probable_maximum_heave := sqrt(Mo_heave)*sqrt(2*ln(N_h)):
        evalf(Probable_maximum_heave);
          Probable_maximum_roll := sqrt(Mo_roll)*sqrt(2*ln(N_r)):
              evalf(Probable_maximum_roll);

                    N_s := 1868.805190
                    N_h := 2052.655994
                    N_r := 2135.782747
                    0.7632995204
                    0.009305826775
                    0.00003252220398                          (3)

> if evalf(Probable_maximum_sway)> 0.8771 then print("sway limit exceeded")
  else print("SWAY LIMIT OK")fi;if evalf(Probable_maximum_heave)> 0.0638 then
  print("heave limit exceeded") else print("HEAVE LIMIT OK")fi;
  if evalf(Probable_maximum_roll)> 0.0031490 then print("heave limit
  exceeded")
  else print("ROLL LIMIT OK")end if;
                    "SWAY LIMIT OK"
                    "HEAVE LIMIT OK"
                    "ROLL LIMIT OK"                           (4)
```