NTNU
Norwegian University of Science and Technology
Department of Marine Technology

MASTER THESIS

Title:	**Delivered:**
SIMULATION OF MOTION OF SES IN WAVES (Simulering av bevegelse av SES i bølger)	9.June 2008 **Availability:** Open
Student: Trygve Espeland Halvorsen	**Number of pages:** 142

Abstract:

A simulation program dealing with seakeeping for SES has been developed. The program uses the Marintek developed simulator VeSim (Vessel Simulator). The forces and motions of a SES are calculated in real-time by using the simulator. Using a visualization program (SimVis), the actual waves and motions of the vessel can be seen. VeSim calculates all hydrodynamic forces and the corresponding motions of the vessel. The SES program contributes with forces due to the air cushion.

The program includes effects of change of air cushion volume, leakages and air flow through fan system. Everything is done in real-time. In addition added viscous resistance is calculated based on wet surface. The seals are modeled as rigid vertical walls. The air cushion pressure is assumed uniform throughout the air cushion. Only the incident wave system is considered, with air cushion pressure giving uniform contribution to wave elevation.

The software has been tested using some simple SES designs, as well as a real vessel. Seakeeping results for these vessels have been compared to linear calculations. The results show good resemblance. Non-linear effects on results have been identified. These include effects from non-vertical side hulls, wet-deck slamming and seal models. The effects from the SES program on seakeeping calculations are as they should be, given the simplifications and modeling.

Calculations of viscous resistance turned out as expected. Results showed that increased wave height gives increased viscous resistance. Also wave periods giving more violent motion give increased viscous resistance. The magnitude of calculated viscous resistance cannot be trusted as the wave elevation is not good enough modeled.

The work with the program is judged a success. A working program has been developed. It captures the most essential effects of a SES. There is however lots of effects that can be added and modeling that can be improved. The hope is therefore that the work shall be continued. With that in mind, this report is thoroughly documented. It covers general seakeeping theory and the special air cushion theory. Effects not included in the program are discussed. The simulator and the program are presented, as well as calculations results. The appendix include a thoroughly walk-through of the SES program.

Keyword:

SES, Hydrodynamic
Seakeeping, Resistance
Time Domain

Advisor:

Professor Bjørnar Pettersen

Address:
NTNU
Department of Marine Technology
N-7491 Trondheim

Location
Marinteknisk Senter
O. Nielsens vei 10

Tel. +47 73 595501
Fax +47 73 595697

NTNU Trondheim
Norges teknisk-naturvitenskapelige universitet
Institutt for marin teknikk

HOVEDOPPGAVE I MARIN HYDRODYNAMIKK

VÅR 2008

FOR

STUD. TECHN. Trygve E. Halvorsen

SIMULERING AV BEVEGELSE AV SES-FARTØY I BØLGER.
(Simulation of motion of SES vehicle in waves)

Bevegelsene av en SES i bølger skal implementeres i en simulator ved Marintek.

Som utgangspunkt gjøres det beregning med en forenklet geometrisk modell av en SES i regulære bølger. Krefter på skroget beregnes i frekvensplanet med Veres, og implementeres til tidsplanet i simulatoren. Modellering av luftpute og skjørt diskuteres, likeså viftekarakteristikker. Tilleggsmotstand i bølger skal også undersøkes.

Visualisering av fartøyet i bølger kan være til hjelp i utviklingen.
Antagelser og forutsetninger som gjøres og som kan ha betydning for resultatene eller skape begrensninger skal dokumenteres.

Kandidaten skal i besvarelsen legge frem sitt personlige bidrag til løsning av de problemer som oppgaven stiller. Påstander og konklusjoner som legges frem, skal underbygges med matematiske utledninger og logiske resonnementer der de forskjellige trinn tydelig fremgår. I besvarelsen skal det klart fremgå hva som er kandidatens eget arbeid, og hva som er tatt fra andre kilder.
Kandidaten skal utnytte de muligheter som finnes til å skaffe seg relevant litteratur for det problemområdet kandidaten skal bearbeide.
Besvarelsen skal være oversiktlig og gi en klar fremstilling av resultater og vurderinger. Det er viktig at teksten er velskrevet og klart redigert med tabeller og figurer. Besvarelsen skal gjøres så kortfattet som mulig, men skrives i klart språk.
Besvarelsen skal inneholde oppgaveteksten, forord, innholdsfortegnelse, sammendrag, hoveddel, konklusjon med anbefalinger for videre arbeide, symbolliste, referanser og eventuelle vedlegg. Alle figurer, tabeller og ligninger skal nummereres.
Det forutsettes at Institutt for marin teknikk, NTNU, fritt kan benytte seg av resultatene i sitt forskningsarbeid, da med referanse til studentens besvarelse.

Besvarelsen leveres 9. juni 2008 i tre eksemplarer.

Faglig veileder: Forsker Dariusz Fathi, Marintek

Bjørnar Pettersen

Bjørnar Pettersen
Professor/faglærer

Preface

This report is the result of my master thesis work spring 2008, at Norwegian University of Science and Technology (NTNU). The topic is seakeeping and added resistance analysis in the time-domain for SES (Surface Effect Ship). The main goal was to develop a time-domain calculation tool utilizing Marintek's vessel simulator (VeSim).

The work was initiated autumn 2007 after suggestion by research manager Dariusz Erik Fathi [FATHI 2008] and senior research engineer Jan Roger Hoff [HOFF 2008] at Marintek, Trondheim. This was my project thesis. It covered seakeeping theory and the foundation of the time-domain air cushion equations used here. In addition, the programming task was initiated (using Matlab).

During autumn 2007 and spring 2008 the project and later master thesis theme was developed and refined in cooperation with Fathi, Hoff and professor Bjørnar Pettersen [PETTERSEN 08] at NTNU. Bjørnar Pettersen was the teacher supervisor both on the project and the master thesis.

Integrating a SES option for the Marintek Vessel Simulator is a vast task, involving much Marintek developed software. The total work for a fully functional integrated calculation tool was much greater than the work load possible in a master thesis. The discussion about what to consider and what to omit has therefore been going on throughout the semester. Options and their decisions are presented where appropriate in this report.

As I had never programmed using Java before (the simulator is programmed using Java), most of the workload has been to learn using Java, do the programming and fighting error messages. Lack of knowledge of how the simulator and the other hydrodynamic software work has also been a progress stopper and workload expander from time to time. Great thanks to Fathi for helping out in times of trouble.

I would also like to acknowledge Bjørnar Pettersen for his always persisting support. Weekendly meetings have been good help. Lieutenant-Commander Christian Wines [WINES 08] has been a great support, supplying vessel and seakeeping data and much good advice.

Professor Sverre Steen [STEEN 08] and professor Odd M. Faltinsen [FALTINSEN 08] have been helping out with advice on progress. From Marintek, besides Hoff, also Master of Science Lasse Bjermeland [BJERMELAND 08] and Master of Science Edvard Ringen [RINGEN 08] have been helping out with the various hydrodynamic software. Finally D.Eng Nere Skomedal [SKOMEDAL 08], head of research and development at UMOE Mandal, has contributed with vessel data. Great thanks to them all!

Trondheim 09.06.2008

Trygve Espeland Halvorsen

Contents

Nomenclature

A	Integration variable	[1/s]
A_j	Frequency dependent amplitude, node functions	[-]
A_b	Air cushion area	[m^2]
A_C	Air cushion area	[m^2]
A_{kj}	Added mass in k-direction due to acceleration in j-direction	Various
A_L	Leakage area	[m^2]
A_{LDyn}	Dynamic leakage area	[m^2]
A_{LStat}	Static leakage area	[m^2]
B	Breadth	[m]
	Integration variable	[1/s]
B_c	Breadth of air cushion	[m]
B_d	Breadth of demi hull	[m]
B_{kj}	Hydrodynamic damping in k-direction due to velocity in j-direction	Various
C	Constant	Various
C_F	Frictional resistance coefficient	[-]
C_V	Viscous resistance coefficient	[-]
c	Velocity of sound in air	[m/s]
c_n	Correction for orifice	[-]
dA	Finite area element	[m^2]
D_r	Moulded depth	[m]
dQ/dp	Change of air flow with respect to air cushion pressure	[m^3/sPa]
dS	Finite surface element	[m^2]
dV	Finite volume element	[m^3]
F	Force	[N]
F_C	Air cushion force	[N]
F_k	Force in k direction	[N]
$F^{air\_cushion}$	Air cushion force	[N]
F^D	Damping force	[N]
F^{Diff}	Diffraction force	[N]
F_{Dyn}	Dynamic force	[N]
F^{exc}	Excitation force	[N]
F^{ext}	External force	[N]
F^{FK}	Froude Kriloff force	[N]
F^M	Mass force	[N]
F_n	Froude number	[-]
F^R	Restoring force	[N]
F_{Stat}	Static force	[N]
f_n	Natural frequency	[Hz]
g	Acceleration of gravity	[m/s^2]
H	Response amplitude operator	Various
	Hull roughness factor	[µm]
h	Height difference between water level inside and outside air cushion	[m]
	Water depth	[m]
h_b	Height difference between mean water surface and roof inside air cushion	[m]
h_c	Height difference between base line and roof inside air cushion	[m]
$h_{sea\ bottom}$	Depth between sea bed and mean free surface	[m]
i	Imaginary unit	[-]
	Various numbering	[-]

i	Vector denotation, x-direction	[-]
j	Various numbering	[-]
j	Vector denotation, y-direction	[-]
k	Wave number	[1/m]
	Various numbering	[-]
	Ride control parameter	[-]
k	Vector denotation, z-direction	[-]
L_c	Length of air cushion	[m]
L_{CG}	Longitudinal center of gravity, relative aft perpendicular	[m]
L_{oa}	Length over all	[m]
L_{pp}	Length between perpendiculars	[m]
M	Mass	[kg]
	Moment	[Nm]
M_C	Air cushion moment	[Nm]
M_{ik}	Coefficient on place (i,k) in the generalized mass matrix	[-]
m	Value used to calculate radiation potentials	[-]
n	Normal vector	[-]
n	Various numbering	[-]
p	Pressure	[Pa]
	Integration variable	[-]
p_0	Constant part of overpressure inside air cushion	[Pa]
p_a	Atmospheric pressure	[Pa]
p_b	Lob bag overpressure	[Pa]
p_c	Air cushion pressure	[Pa]
p_{sp}	Spatially varying over pressure	[Pa]
p_u	Uniform over pressure	[Pa]
Q	Volume flow	[m^3/s]
Q_0	Static volume flow	[m^3/s]
Q_{in}	Volume flow into air cushion	[m^3/s]
Q_{inDyn}	Dynamic volume flow into air cushion	[m^3/s]
Q_{out}	Volume flow out from air cushion	[m^3/s]
r	Distance from center of gravity	[m]
r	Node function	[-]
r44	Radius of gyration, roll	[m]
r55	Radius of gyration, pitch	[m]
r66	Radius of gyration, yaw	[m]
R_A	Air resistance	[N]
R_{kj}	Restoring force in k-direction due to position in j-direction	Various
R_n	Reynold’s number	[-]
R_P	Pressure related resistance	[N]
R_T	Total vessel resistance	[N]
R_V	Viscous resistance	[N]
S	Wave spectrum	[m^2/s]
	Wet surface	[m^2]
S-	Control surface	[-]
S+	Control surface	[-]
So	Control suface, body surface	[-]
S_0	Static wet surface	[m^2]
Sb	Control suface, seabed	[-]
Sf	Control suface, free surface	[-]

s	Position vector	[m]
T	Draught	[m]
	Wave period	[s]
t	Time	[s]
u	Fluid velocity	[m/s]
U	Ship velocity	[m/s]
V_{CG}	Vertical center of gravity, relative base line	[m]
x	Coordinate in x-direction	[m]
y	Coordinate in y-direction	[m]
z	Coordinate in z-direction	[m]

Greek symbols

ε	Phase angle	[rad]
β	Angle between wave and vessel heading	[°]
δ	Ratio viscous added resistance / total viscous resistance	[-]
γ	Specific heat of air	[-]
ζ	Wave height	[m]
ζ_a	Wave height amplitude	[m]
$\zeta_{incident}$	Incident wave height amplitude	[m]
η_1	Sway. Rigid vessel motion	[m/s]
η_2	Surge. Rigid vessel motion	[m/s]
η_3	Heave. Rigid vessel motion	[m/s]
η_4	Roll. Rigid vessel motion	[rad/s]
η_5	Pitch. Rigid vessel motion	[rad/s]
η_6	Yaw. Rigid vessel motion	[rad/s]
η_7	Dynamic over pressure coefficient	[-]
μ	Dynamic over pressure coefficient	[-]
μ_{sp}	Dynamic spatially varying overpressure coefficient	[-]
μ_u	Dynamic uniform varying overpressure coefficient	[-]
ρ	Mass density of seawater	[kg/m^3]
ρ_{air}	Mass density of air	[kg/m^3]
ρ_c	Mass density of air inside air cushion	[kg/m^3]
ρ_{c_0}	Static mass density of air inside air cushion	[kg/m^3]
$\boldsymbol{\omega}$	Rotation vector	[rad]
ω	Frequency	[rad/s]
ω_0	Wave frequency	[rad/s]
ω_e	Frequency of encounter	[rad/s]
ω_n	Natural frequency	[rad/s]
Ω	Volume of air cushion	[m^3]
Ω_0	Static volume of air cushion	[m^3]
Φ	Total velocity potential	[m^2/s]
φ	Velocity potential	[m^2/s]
φ_D	Diffraction potential	[m^2/s]
φ_I	Incident wave potential	[m^2/s]
φ_j, φ_k	Radiation potentials, j and k={1,…,7}	[m^2/s]
φ_s	Steady perturbation potential	[m^2/s]
φ_u	Unsteady potential	[m^2/s]
θ	Wave heading	[°]
λ	Wave length	[m]

υ	Kinematic viscosity	[m^2/s]

Mathematical operators and other symbols

∇	Differential operator
Δ	Small increment between two successive numbers
$\times$	Vector product
$\frac{d}{d}$	Derivative
$\frac{\partial}{\partial}$	Partial derivative
$\frac{D}{Dt}$	Substantial derivative
$\iint$	Integrating in 2 dimensions
$\iiint$	Integrating in 3 dimensions
$\lvert\ \rvert$	Absolute number
$\dot{}$	First derivative with respect to time
$\ddot{}$	Second derivative with respect to time
∞	Infinity

Abbreviations

AP	Aft Perpendicular
BL	Base Line
CG	Center of gravity
CL	Center Line
CSI	Common Simulation Interface
DOF	Degree Of Freedom
FK	Froude Kriloff
FP	Fore perpendicular
IMO	International Maritime Organization
JDK	Java Development Kit
NED	North East Down
NMRIWF	Non-linear Modification of Restoring and Incident Wave Forces
NTNU	Norwegian University of Science and Technology
RAO	Response Amplitude Operator
SES	Surface Effect Ship
VERES	Vessel Response program
VeSim	Vessel Simulator
SimVis	Simulation Visualization

Introduction

The seakeeping properties (how the ship behaves in the sea) are crucial for a ship, and are very dependent on the shape of the hull. It is easier and a lot cheaper to change the ship design early in the design process. Thus the seakeeping properties should be decided as fast as possible to avoid late design changes of the hull. The objectives of seakeeping tests can also be e.g. to determine operational limits, perform capsize and safety studies or document maneuvering criteria.

The cheapest way to assess seakeeping properties is numerical calculations. It is more cost-effective and faster than model testing, but model testing is still more reliable.

At present Marintek only offers the possibility of dealing with seakeeping for SES in the frequency-plane. A former analysis [WINES et.al. 07] indicates that pitch motions for a SES are exaggerated in linear theory, maybe with as much as 20-40%. When operating in the frequency plane many effects are missed out. Marintek are developing a time-domain analysis tool called VeSim (Vessel Simulator), but this does not support SES. By developing a SES addition to this simulator, the author hopes to improve the quality of seakeeping analysis for SES.

Resistance calculations are also important when designing ships. A special effect with SES is that it experiences a larger involuntary speed loss in waves than a similarly sized catamaran. This effect is explored in this report.

A java program dealing with seakeeping and added resistance in waves for SES has been developed. The theory behind this program and VeSim is presented in this report. As a complete program with all the necessary effects include far more work than possible in a master thesis, the methods and principles are thoroughly documented (for instance an overwhelming amount of figures are used). This was done in order for other people to continue the work.

The program and this report are written entirely by Trygve Espeland Halvorsen. Dariusz Fathi [FATHI 08] has been helping out a lot with the transition between the SES program and VeSim. The derivation from boundary conditions for air cushion pressure to equations that can be used in the time-domain has been made by the author. Also all principles for how volumes, leakages, forces etc. are defined and used in the java code have been developed by the author. The different simple vessel designs, and all calculations and analysis have been made by the author.

First a short introduction to the SES concept is given. Then general seakeeping theory is presented. This is theory that is applied in VeSim. Next the different effects with a SES are discussed. Air cushion equations are derived. Added resistance in waves is discussed before VeSim is presented. Then the SES program is displayed before seakeeping and resistance calculations are performed. The goal of these is to test that the program gives the output it should do. The appendix includes a thoroughly walkthrough of the different part of the SES program.

It is throughout the report assumed that the reader has some basic knowledge about hydrodynamics and mathematics.

The SES concept

The Surface Effect Ship (SES) is a ship where a large part of the lift force is supplied by one or several air cushions. This in contrast to a regular displacement ship where all lift is due to buoyancy and hydrodynamic forces. The lift force supplied by the air cushion(s) typically carries 80% of the weight of the SES, while the rest is carried by buoyancy from the side hulls [FALTINSEN 05, chapter 5.1]. The side hulls are also often referred to as demi hulls.

The SES concept developed by UMOE Mandal is showed in the illustration below and on the next page. The vessel has one air cushion that is enclosed by two rigid side hulls and flexible seal systems in bow and aft. The seal system is in aft a flexible bag consisting of several loops (See Figure 2 and Figure 3). The pressure inside the aft bag is larger than inside the air cushion (lob bag overpressure p_b). The front seal (skirt) is a finger seal, which consists of a row of vertical loops of flexible material. See Figure 1 and Figure 2.

Figure 1: The Skjold class Fast Attack Craft, developed by UMOE Mandal. Pictures from [UMOE 08]

The pressure inside the air cushion (air cushion pressure consist of overpressure p_c plus atmospheric pressure p_a) is larger than the pressure outside (atmospheric pressure, p_a). The resulting net force lifts the SES up. The excess pressure (p_0) also causes the water lever inside the air cushion to be lower than the water level outside (see Figure 4). The excess pressure is provided by an air fan system, which may use about 20-30% of total installed power

[MINSAAS 06, p.182]. On cushion is the notation for an SES with the air cushion active. Off cushion means that the cushion is not active.

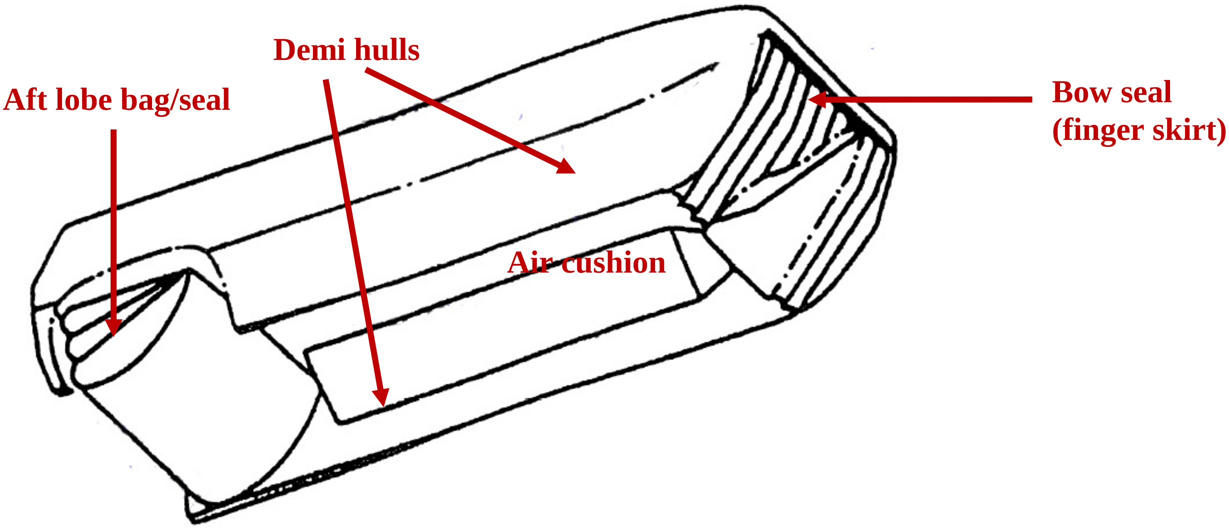


Figure 2: Illustration of the SES concept. View from below [STEEN 07 TMR]

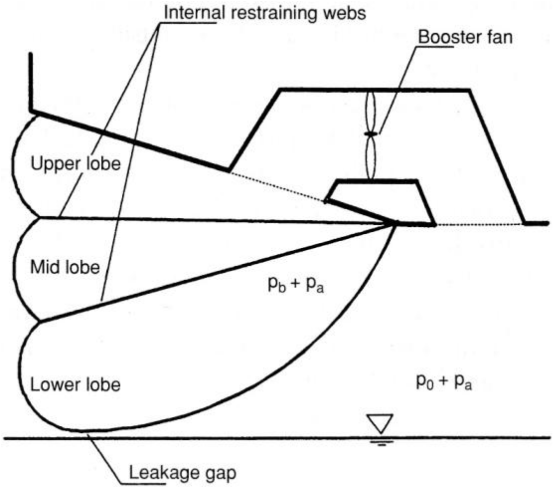


Figure 3: Aft seal system consisting of a flexible bag [FALTINSEN 05, figure 5.10]

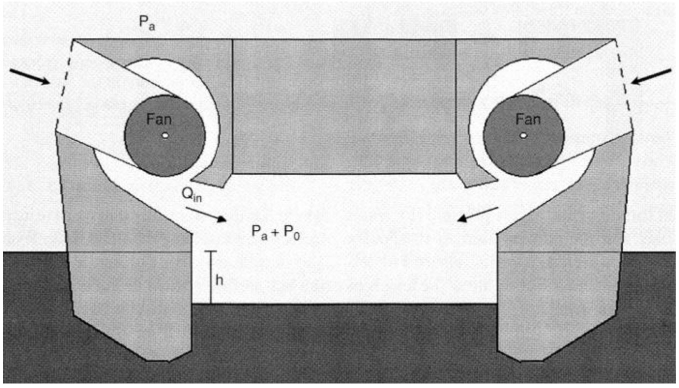


Figure 4: Cross section of a SES. Overpressure inside air cushion provided by lift fans [FALTINSEN 05, figure 5.4]

The SES concept offers several benefits:

- Large and wide deck for operations

- Low resistance at high ship speed (the viscous resistance is small due to small wetted surface). In practice this means higher speed with less installed power
- Better seakeeping capabilities than a similarly sized catamaran in moderate sea states
- Low draught on cushion
- Less sensitive to underwater explosives

Drawbacks for a SES are:

- Wear of seals
- Ventilation and cavitation of waterjet system
- Slamming in wet deck
- Speed loss in waves (due to ventilation of air cushion)
- Cobblestone oscillations
- More costly and complicated than conventional fast ships

For a more extensive introduction to the SES concept, the reader is referred to e.g. [FALTINSEN 05].

1 Seakeeping theory

Numerical seakeeping consists basically of three steps:

- Establish equations of motions
- Establish boundary conditions
- Solve equations of motions by using boundary conditions and suitable analysis tools

The equations of motions come from equilibrium of forces. The boundary conditions are how physically conditions are implemented. How the system is modeled, and how the equations are modified and solved, heavily depend on what kind of theory is used. There are also several different ways of describing the ship (the geometry) and environment.

The theory used throughout this project is based on potential theory, which is a very common assumption in seakeeping theory. The theory presented is not very detailed, and are concentrated around the special adaptation to air cushion crafts. For more in depth theory, [FALTINSEN 90] and [FALTINSEN 05] may be consulted.

Cartesian coordinates x, y and z are used throughout the text. The terms used, both here and in the next chapter, are mostly those of [FALTINSEN 05 chapter 3 and 5].

1.1 Main assumptions

The basic assumptions in potential theory are

- Irrotational motions
- Incompressible fluid
- Inviscid fluid

The following assumptions are also applied

- The wave elevation is considered independent of the air cushion pressure when calculating hydrodynamic forces
- Waves are harmonic and sinusoidal
- Only rigid body motions (no hydroelastic effects)
- Strip theory is applicable

Main assumptions in linear theory

- Linear relation between response and incident wave amplitude
- The superposition principle is used to derive the loads and motions in a sea state. The loads are calculated for each wave frequency and then added together
- Small translations and waves
- The ship oscillate harmonically with frequency equal to frequency of encounter
- No transient effects due to initial conditions

1.2 Seakeeping characteristics of a SES

The seakeeping characteristics of an SES are dominated by the air cushion. It is even argued that the hydrodynamic forces on the hull can be disregarded in seakeeping analysis [FALTINSEN 05 p.151]. The justification for this is that typically about 80% of the weight of the SES is carried by the air cushion, and that this is proportional to the relative importance between hydrodynamic and air cushion forces.

One of the unique characteristics with an SES is the difference in water levels inside and outside air cushion. When there are no waves or motion, the pressure in the water will vary linearly. This is illustrated in Figure 5.

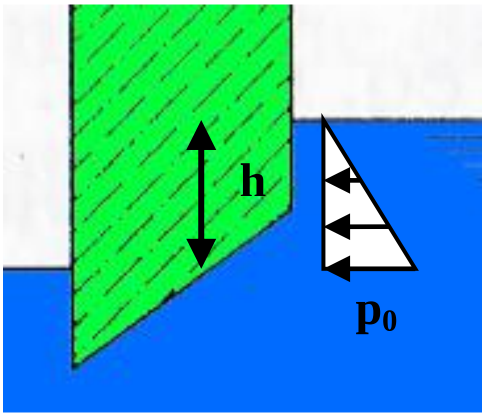


Figure 5: Static conditions, resulting in static overpressure and static difference in water levels inside and outside air cushion

The static difference in water levels, h, can be calculated from Bernoulli's equation. The result is:

$$h = \frac{p_0}{\rho g} \tag{1.1}$$

Where ρ denotes density of sea water and g is acceleration of gravity. Typically h can be in the range of 0.5 to 1 [m].

The motions of the SES will induce pressure oscillations in the air cushion. These pressure oscillations are coupled to the rigid body motions of the SES. This because the air cushion pressure will contribute to the dynamic forces on the vessel. This implies that that the air cushion pressure must be treated as an additional variable in the equations of motions for the SES. To solve these equations it is then needed an additional equation where the air cushion pressure is included. This equation is the continuity equation for the air in the air cushion.

1.3 Description of the environment

The undisturbed wave elevation can be described in numerous ways. A common way of doing it is the linear way, where the wave elevation is assumed to consist of many harmonic sinusoidal waves. The total wave elevation is the sum of all wave components. Each component has different wave frequency. The wave elevation can then be described as:

$$\zeta = \sum_{j=1}^{N} \zeta_{a\_j} \sin\left(\omega_j t - k_j x + \varepsilon_j\right) - \frac{p_c(t)}{\rho g} \tag{1.2}$$

Where ζ denotes the incident wave elevation and ζ_{a_j}, ω_j, k_j and ε_j are respectively the wave height amplitude, wave frequency, wave number and random phase angle of the j-th wave component. t denotes time and x is coordinate in x-direction. $p_c/\rho g$ is contribution from the air cushion pressure p_c. This term is zero outside the air cushion.

The wave amplitude can be described by a wave spectrum S(ω):

$$\zeta_{a\_j} = \sqrt{2S(\omega_j)\Delta\omega} \tag{1.3}$$

Where Δω is a constant frequency increment between two successive frequencies. The wave spectrum can be seen upon as wave energy density, distributing wave energy over the different wave frequencies.

If the wave spectrum is given as S(ω) (no wave heading), the waves are called long-crested. That means that they only move in one direction, making the wave elevation a function of position in x-direction and time only (and not position in y-direction). If the waves are moving in several directions, they are called short-crested. The wave spectrum are then a function of both wave frequency and wave heading, S = S(ω,θ). The wave amplitude will now be:

$$\zeta_{a\_j} = \sqrt{2S(\omega_j,\theta_k)\Delta\omega_j\Delta\theta_k} \tag{1.4}$$

Where $\Delta\theta_k$ is the angular increment between two successive angles. The wave elevation can then be expressed as:

$$\zeta = \sum_{j=1}^{N}\sum_{k=1}^{K}\zeta_{a\_jk}\sin\left(\omega_j t - k_j\left(xcos(\theta_k) + ysin(\theta_k)\right) + \varepsilon_{jk}\right) - \frac{p_c(t)}{\rho g} \tag{1.5}$$

When the ship is moving, it “sees” a different frequency than the wave frequency. This frequency is called the frequency of encounter and is given by

$$\omega_e = \omega_0 + kUcos\beta \tag{1.6}$$

Where ω_0 is the incident wave frequency, U is constant ship velocity and β is wave heading relative vessel. Value 0 is head sea.

One of the basic assumptions about the wave conditions is that the motions are irrotational. Mathematically this is described by

$$\boldsymbol{\nabla} \times \boldsymbol{u} = 0 \tag{1.7}$$

Where **u** is the fluid velocity and ∇ is the differential operator and × is vector product. It follows from this that the fluid velocity can be described by a velocity potential Φ as

$$u = \nabla\Phi = i\frac{\partial\Phi}{\partial x} + j\frac{\partial\Phi}{\partial x} + k\frac{\partial\Phi}{\partial x} \tag{1.8}$$

Here **i**, **j** and **k** are vector denotation for x-, y- and z-direction.

If the vessel is travelling with constant velocity along a straight course, the velocity potential ($\Phi = \Phi(x, y, z, t)$) can be expressed as (in an axis-system fixed to the vessel) (note: some literature uses a negative sign on Ux. This has to do with definition of axis system):

$$\Phi = Ux + \varphi \tag{1.9}$$

φ can further be divided into steady disturbance potential and unsteady oscillatory potential, $\varphi = \varphi_s + \varphi_u$.

Now assuming incompressible fluid ($\nabla \cdot u = 0$), the Laplace equation must be satisfied:

$$\nabla\mathbf{u} = \nabla^2\Phi = \frac{\partial^2\Phi}{\partial \mathrm{x}^2} + \frac{\partial^2\Phi}{\partial \mathrm{y}^2} + \frac{\partial^2\Phi}{\partial \mathrm{z}^2} = 0 \tag{1.10}$$

Using potential theory, the fluid pressure can be described by Bernoulli's equation

$$p + \rho\frac{\partial\Phi}{\partial t} + \frac{\rho}{2}|\nabla\Phi|^2 + \rho g z = C \tag{1.11}$$

Here p denotes pressure in the fluid, ρ is density of the fluid, g is acceleration of gravity and C is a constant. The z-axis is vertical and positive upwards. The constant can be evaluated far away from the ship where the ship causes zero disturbance of the water.

1.4 Equations of motions

The equations of motions is deduced from Newton's second law, which states that mass times acceleration equals sum of all forces:

$$\sum F = ma \tag{1.12}$$

In seakeeping theory, the forces are usually divided into whether they are in phase with acceleration, velocity or position. Forces not in phase with this are called external forces. Following this, forces in phase with acceleration are called mass forces, damping forces are forces in phase with velocity and restoring forces are in phase with position. This can be written:

$$F^M + F^D + F^R = F^{ext} \tag{1.13}$$

Here F^M is mass forces (includes mass times acceleration and hydrodynamic mass), F^D is damping forces, F^R is restoring forces and F^{ext} is other external forces including air cushion forces (other external forces are e.g. wind, current and excitation forces). External forces are forces working from the surroundings (fluid/air cushion/wind) and on to the vessel.

1.4.1 Description of motions

Hydroelastic effects are neglected. Then only rigid body modes remain. There are three translation modes and three rotations, denoted η_k where k = {1,…,6}. The modes are defined different in VERES (Vessel Response program) and in VeSim (Vessel Simulator, see chapter 4). See Figure 6 and Figure 7.

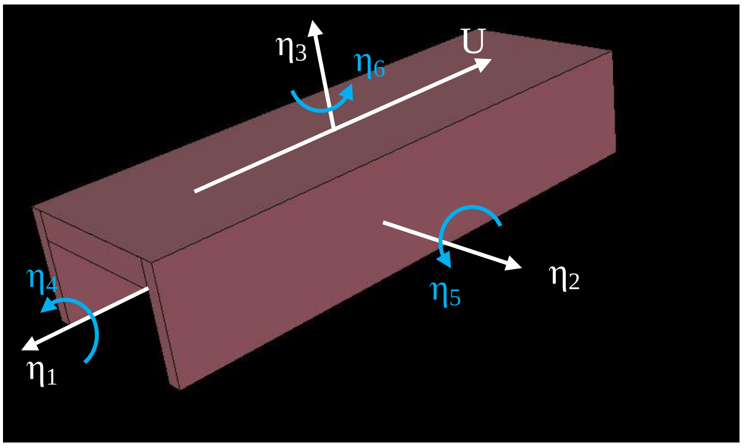


Figure 6: Sign conventions for translatory and rotational displacements according to VERES. Center of axis system is center of gravity

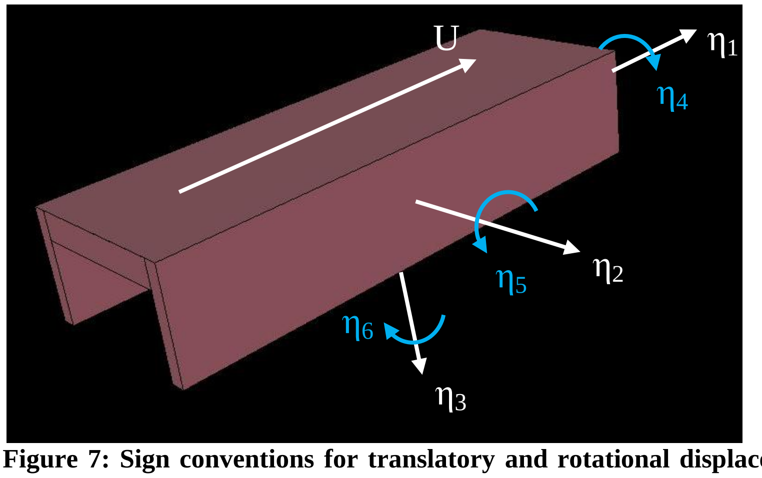


Figure 7: Sign conventions for translatory and rotational displacements according to VeSim. Center of axis system is center of gravity

When the body modes are known, the position, velocity and acceleration of any single point can be written as

$$\boldsymbol{s}(x, y, z, t) = \eta_1 \boldsymbol{i} + \eta_2 \boldsymbol{j} + \eta_3 \boldsymbol{k} + \boldsymbol{\omega} \times \boldsymbol{r} \tag{1.14}$$

where

$$\boldsymbol{\omega} = \eta_4 \boldsymbol{i} + \eta_5 \boldsymbol{j} + \eta_6 \boldsymbol{k} \quad \text{and} \quad \boldsymbol{r} = x\boldsymbol{i} + y\boldsymbol{j} + z\boldsymbol{k} \tag{1.15}$$

Differentiating the rigid body modes with respect to time gives velocity and differentiating them twice gives accelerations.

1.4.2 Force components in regular waves

The different force contributors need to be find. In potential theory, this problem is divided into two:

A. The ship is restrained from oscillation and there are incident waves. Forces are denoted excitation forces (F^{exc}), which consist of Froude Kriloff forces (F^{FK}) and diffraction forces (F^{Diff}). Froude Kriloff forces come from the pressure induced by the undisturbed wave field. The vessel's presence changes the pressure field, and this contributor is the diffraction forces. These forces belong to F^{ext} in equation 2.13.

B. The vessel oscillates in calm water, thus creating radiating waves. There are no incident waves. Forces are restoring forces and hydrodynamic mass (added mass) and damping

In addition to hydrodynamic damping, there is also viscous damping. When using potential theory as here this contribution is disregarded. As it is more important for single hull vessel (provides roll damping), this assumption is good.

The forces described so far are parts of the unsteady fluid potential. This can be expressed as [VERES SES, p.26-27]

$$\Phi(x,y,z,t) = Ux + \varphi_s(x,y,z) + \varphi_u(x,y,z,t) \tag{1.16}$$

Where φ_s is the steady perturbation potential and φ_u is the unsteady potential. The unsteady potential can further be split down to (when linearized):

$$\varphi_u = \left[\zeta_a(\varphi_I + \varphi_D) + \sum_{k=1}^{7} \eta_{k,a}\varphi_k\right] e^{i\omega t} \tag{1.17}$$

Where φ_I is incident wave potential, φ_D is wave diffraction potential, φ_k (k={1,…,6}) are radiation potentials and φ_k (k = 7) is air cushion potential.

The radiation and diffraction potentials are assumed independent of the air cushion (when the height difference in water levels is ignored) and hence identical to those obtained for a regular catamaran. The air cushion potential is due the air cushion deflecting the free surface inside the air cushion. This potential is not included in any of the theories presented here.

The forces and moments are decomposed in forces in x-, y- and z-direction and moments around the same axis. For excitation forces, which is for zero body motion, we will then have six Froude-Kriloff (from incident wave potential φ_I) forces and moments and six diffraction (from diffraction potential φ_D) forces and moments.

The radiation forces are induced by motion, thus there is a coupling between the different motions and forces. The radiation forces and moments are hydrodynamic mass and damping. Formally these are written:

$$F_k = -A_{kj}\frac{\partial^2 \eta_j}{\partial t^2} - B_{kj}\frac{\partial \eta_j}{\partial t} \tag{1.18}$$

Where F_k is radiation force (or moment) in k-direction, A_{kj} and B_{kj} is added mass and damping in k-direction due to respectively acceleration and damping in j-direction.

Same as with radiation forces, the mass times acceleration is dependent on the different motions. A pitch motion will for instance induce acceleration in z-direction. Hence a generalized mass matrix is build. The same analogy goes for the restoring force, which is formally defined as:

$$F_k = -R_{kj}\eta_j \tag{1.19}$$

Here F_k denotes restoring force in k-direction and R_{kj} is restoring force in k-direction due to position in j-direction.

The equations of motions can now be described as:

$$\sum_{k=1}^{7}[(M_{ik}+A_{ik})\ddot{\eta}_k+B_{ik}\dot{\eta}_k+R_{ik}\eta_k]=F_i^{exc}+F_i^{air\_cushion} \quad \forall\, i=\{1,\dots,7\} \tag{1.20}$$

Here M_{ik} is coefficient from the generalized mass matrix. k and i denotes the different rigid body modes. The dots mean time differentiation. One dot gives velocity and two dots give acceleration. F_i^{exc} equals 0 for i = 7, and $F_i^{air\_cushion}$ equals 0 for $i \neq 7$. $M_{i7} = 0$. Compared with equation 1.13, M and A equal F^M, B equals F^D and R equals F^R. F^{exc} and $F^{air\_cushion}$ equals F^{ext}.

1.4.3 Boundary conditions

The potentials describing the wave conditions must fulfill the following boundary conditions (in addition to Laplace equation):

1) Free surface conditions
   a) Kinematic: A fluid particle on the water surface stays on the water surface. The exact condition (not linearized) reads:

$$\frac{D}{Dt}(z-\zeta(x,y,t))=0 \; on \; z=0 \tag{1.21}$$

   Where D/Dt denotes the substantial derivative
   b) Dynamic: The dynamic pressure on the free surface equals atmospheric pressure outside air cushion and air cushion pressure inside air cushion. Outside the air cushion the equation becomes:

$$\frac{Dp}{Dt}=-\rho\frac{D}{Dt}\left(\frac{\partial\Phi}{\partial t}+\frac{1}{2}|\nabla\Phi|^2+gz\right)=0 \; on \; z=\zeta(x,y,t) \tag{1.22}$$

   Inside the air cushion, the air cushion pressure needs to be accounted for. This gives:

$$\frac{D}{Dt}[p-p_c]=-\rho\frac{D}{Dt}\left(\frac{\partial\Phi}{\partial t}+\frac{1}{2}|\nabla\Phi|^2+gz\right)=0 \; on \; z=\zeta(x,y,t) \tag{1.23}$$

   Here p_c denotes the pressure inside the air cushion.

   The kinematic and dynamic free surface conditions may be linearized and combined. See for instance [VERES SES, p.26]
2) Radiation condition, the waves generated by the body radiates away from the body
3) Body boundary conditions
   a) Radiation: The velocity of the water particle adjacent to the body is equal to the body's velocity

$$\boldsymbol{u} \cdot \boldsymbol{n} = \dot{\boldsymbol{s}} \cdot \boldsymbol{n} \text{ on submerged part of hull} \quad (1.24)$$

Where n is the normal vector to the hull surface. With the radiations potentials defined the condition can be shown to be [HYDRO VESIM p.23]:

$$\frac{\partial \varphi_k}{\partial n} = i\omega n_k + U m_k \;\; \forall\, k = \{1, \ldots, 6\} \quad (1.25)$$

Where n is the normal vector on the hull surface and s is the motion of the body at that position. k denotes rigid body mode number. m_k is defined as:

$$(m_1, m_2, m_3) = \boldsymbol{m} = (n\nabla)\nabla\left(-x + \frac{1}{U}\varphi_S\right) \quad (1.26)$$

$$(m_4, m_5, m_6) = \boldsymbol{r} \times \boldsymbol{m} - \nabla\left(-x + \frac{1}{U}\varphi_S\right) \quad (1.27)$$

b) Diffraction and incident waves: The water particles does not penetrate the body

$$\boldsymbol{u} \cdot \boldsymbol{n} = 0 \text{ on submerged part of hull} \quad (1.28)$$

Or with the potentials defined:

$$\frac{\partial \varphi_I}{\partial n} + \frac{\partial \varphi_D}{\partial n} = 0 \quad (1.29)$$

c) Steady perturbation potential: The water particles does not penetrate the body

$$\frac{\partial}{\partial n}[Ux + \varphi_S] = 0 \quad (1.30)$$

4) Sea bottom condition. No fluid particle penetrated the sea bottom. This condition is equal for all potentials. For finite water depth it reads:

$$\frac{\partial \Phi}{\partial z} = 0 \; on \; z = -h_{sea\ bottom} \quad (1.31)$$

Where $h_{sea\ bottom}$ denotes water depth. When the depth is infinite, the condition reads:

$$|\nabla \Phi| \rightarrow 0 \; when \; z \rightarrow -\infty \quad (1.32)$$

1.4.4 Strip theory

The different forces are here found by means of strip theory. In strip theory the underwater part of the ship is divided into a number of 2 dimensional strips. The hydrodynamic forces are calculated separately in means of 2 dimensional coefficients for each strip. Total forces are obtained by integrating the coefficients over the length of the ship. As the forces are evaluated separately at each cross section, no 3 dimensional flow effects are accounted for.

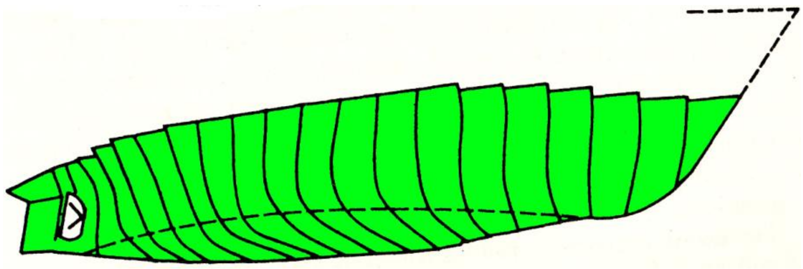


Figure 8: Strip theory, only submerged part of hull is considered [FALTINSEN 90, figure 3.7]. The figure is modified by the author

Assumptions in strip theory

The following assumptions are applied in strip theory

- High wave frequency (implies that difficulties can be encountered for quartering and following waves)
- The flow varies much more in the cross-sectional plane than in the longitudinal (the ship generates waves only in transverse direction)
- Large length to beam ratio (slender vessel)
- Moderate Froude number (Froude number = $F_n = U/(L_{pp}*g)^{0.5}$) (care should be shown in applying the theory for Froude number larger than about 0.4 [FALTINSEN 90 p.57]). No interaction between the strips are accounted for
- Linearity between response and incident wave amplitude (strip theory is questionable to apply in high sea states)

The following procedures for solving the hydrodynamic forces are from the VERES theory manual [FATHI & HOFF 04].

The boundary value problem is solved for each cross sectional strip of the body in the fluid domain. The diffraction and radiation potential are solved by matching the near-field solution and a far-field solution. The near-field solution is obtained by a numerical boundary-element formulation, while the far-field solution is obtained by an asymptotic solution.

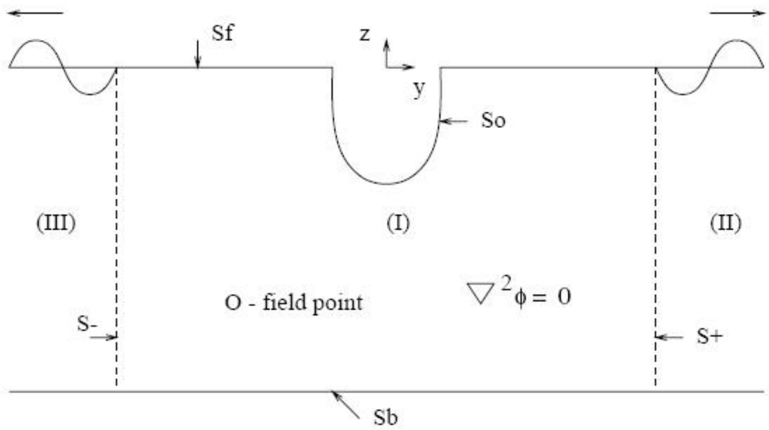


Figure 9: Fluid domain and control surfaces according to strip theory [FATHI & HOFF 04]

In Figure 9 So is the body surface. S- and S+ are the control surfaces far away from the body. Sb is seabed and Sf is the free surface.

1.5 Solving equations of motions

The equations are solved linearly in VERES. VeSim is used for the time-domain analysis.

1.5.1 Linear solution

When working in the frequency plane, it is assumed that the motions can be described by adding up harmonic motions with different frequencies. For each frequency, this can be described as

$$\sum_{k=1}^{7}[(M_{ik}+A_{ik})\ddot{\eta}_k(\omega)+B_{ik}\dot{\eta}_k(\omega)+R_{ik}\eta_k(\omega)] = F_i^{exc}(\omega)+F_i^{air\_cushion}(\omega) \quad \forall\, i=\{1,\dots,7\} \tag{1.33}$$

The responses of the vessel and the air cushion pressure can be described on complex form as

$$\eta_j=\eta_{ja}e^{-i\omega_e t}\ \forall\, j=\{1,\dots,7\} \tag{1.34}$$

The excitation forces and air cushion forces are described as

$$F_i^{exc}=F_{ia}^{exc}e^{-i\omega_e t}\ \forall\, i=\{1,\dots,7\} \tag{1.35}$$

Inserting these expressions into the equation of motions:

$$\sum_{j=1}^{7}\left[-\omega_e^2\left(M_{ij}+A_{ij}\right)+i\omega_e B_{ij}+R_{ij}\right]\eta_{ja}=F_{ia}^{exc}+F_{ia}^{air\_cushion} \quad \forall\, i=\{1,\dots,7\} \tag{1.36}$$

When all force coefficients are known, the equation system is solved with regards to the body modes. The wave amplitude is separated from the other terms, so that the body modes can be described:

$$\eta_j(\omega_e)=H_j(\omega_e)\zeta(\omega_e) \tag{1.37}$$

Where H(ω) is called a response amplitude operator. The equations of motions are solved for all relevant encounter frequencies, and a plot as shown in Figure 10 is obtained.

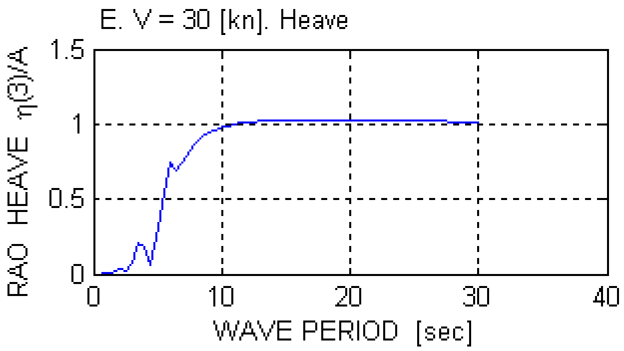


Figure 10: Example of a RAO function. For each wave (with accompanying wave period) the response is found by multiplying the RAO value with the wave amplitude (A in the figure)

1.5.2 Time-domain solution

The equations of motions are in time-domain theory solved for every time step. The theory behind the calculations are quite extensive, thus only a simple overview is presented here (as explained by Research Manager Dariusz Erik Fathi at MARINTEK [FATHI 07]). The reader thirsting for more knowledge is encouraged to consult the VESIM's documentation [HYDRO VESIM 04].

Using the mass matrix as before, the equations of motion can be written as:

$$\boldsymbol{M}\ddot{\eta}_j = \sum F_j \ \ \forall\, j = \{1, \dots, 6\} \tag{1.38}$$

Where F_j denotes the different force contributors in j-direction. The equation system is solved for the different accelerations as:

$$\ddot{\eta}_j = \boldsymbol{M}^{-1} \sum F_j \ \ \forall\, j = \{1, \dots, 6\} \tag{1.39}$$

This equation is solved for each time step (every 0.05 [s] is the default time step for this). Velocity is found by integrating the acceleration once and position/rotations by integrating twice.

As the different force components are dependent on the vessel motions, these are evaluated for the previous time step. This can be written as:

$$\begin{gathered} \ddot{\eta}_{j_n} = \ddot{\eta}_{j_n}(F_{j_n}) \ \ , \ \ F_{j_n} = F_{j_n}(\ddot{\eta}_{i_n-1}, \dot{\eta}_{i_n-1}, \eta_{i_n-1}) \\ j = \{1, \dots, 6\} \ and \ i = \{1, \dots, 7\} \end{gathered} \tag{1.40}$$

Where n denotes the time step. j denotes rigid body modes, and i denotes the 7 different variables, that is 6 rigid body modes and air cushion pressure.

The velocities are found by integrating the accelerations once and the positions by integrating twice.

It is important that the time steps are sufficiently small. If too large time step, the motions used when deciding the forces are no longer valid, and the new calculated motions may be significantly off from what they should have been.

1.5.3 The different forces' influence on motion

A typical equation of motion, at steady state, linear theory, has the following setup (as shown in equation 1.36):

$$[-\omega^2 F^M + i\omega F^D + F^R]\eta_a = F^{ext} \tag{1.41}$$

Where the wave frequency is defined as

$$\omega = \frac{2\pi}{T} \tag{1.42}$$

From the two equations, it is seen that at for low wave period, ω is large and thus mass forces are the largest contributor in deciding motion. At large wave periods, ω is small and thus restoring forces are most important. This means that for large periods, which mean large waves, the restoring forces will make the vessel follow the waves. If for instance heave motion is divided with wave amplitude, the result will be 1.

At intermediate wave periods, damping will be important to decide response peaks.

The regions where the different forces dominate are shown in Figure 11.

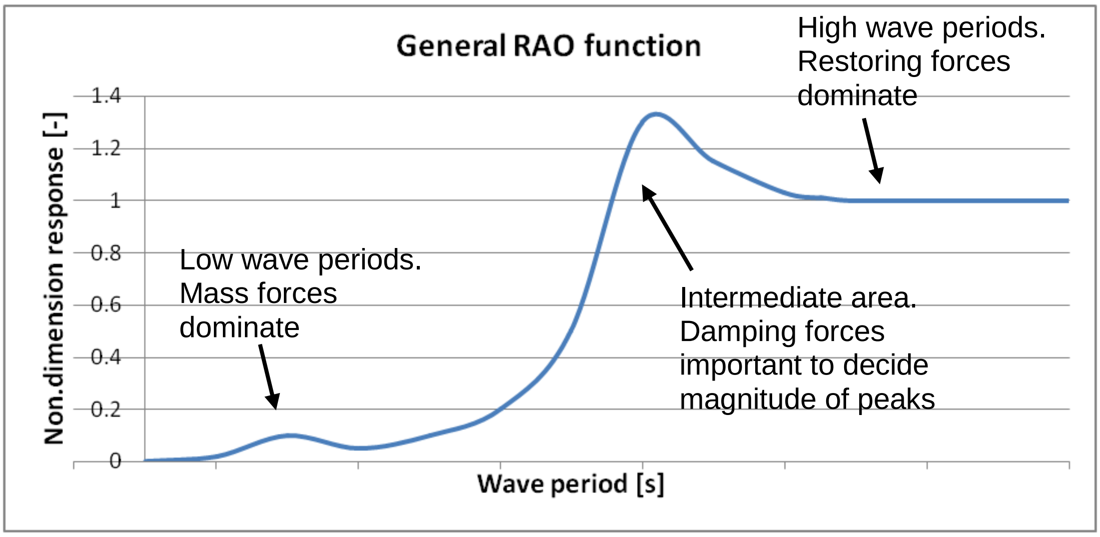


Figure 11: General RAO function. Shows in which regions the different force types, mass, damping and restoring, dominate

Reviewing the different forces, mass forces are mass of vessel and hydrodynamic mass. Damping is viscous damping (not accounted for here) and wave making. Restoring forces are buoyancy. External forces are e.g. waves, current and wind.

2 Air cushion theory

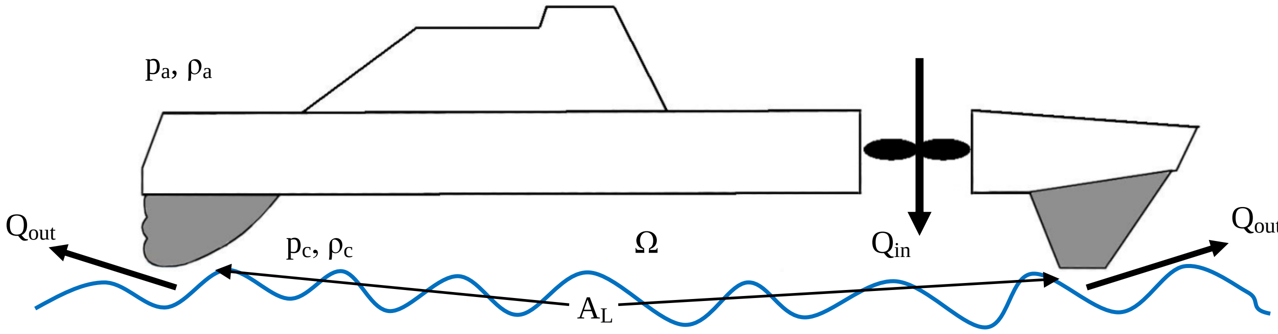


Figure 12: Expressions used in the air cushion equations

The air cushion pressure induces forces on the vessel. The pressure inside the air cushion differs from atmospheric pressure, thus resulting in a force due to the difference in pressure. The goal is to find the pressure. When the pressure is found, forces are calculated by integrating the pressure over hull inside the air cushion. In a seakeeping respect, it is the heave and pitch induced motions (due to the air cushion) that are most important.

The air cushion pressure can be simplified to be uniform in the air cushion. The linear theory used by VERES makes this assumption. The heave motion can be relatively good captured by this assumption (computations with VERES show good agreement with heave motion in low sea states, according to [WINES 08]). However, this method misses the effect spatial varying air cushion pressure has on pitch motion. So the calculated pitch motions show not very good agreement. This report tries to deal with both uniform and spatially varying air cushion pressure. Pressure variations in y- and z-direction are not considered. This is justified by the large length-to-beam and -height ratio of the air cushion. The overpressure in the air cushion, p_c, can then be given by:

$$p_c(t) = p_u(t) + p_{sp}(x,t) \tag{3.1}$$

Where p_u is uniform overpressure and p_{sp} is spatially varying overpressure. It is often convenient to express the dynamic air cushion pressure with non-dimensional coefficients. The constant overpressure is then denoted p_0.

Uniform dynamic air cushion pressure represented by μ_u:

$$\mu_u(t) = \frac{p_u(t) - p_0}{p_0} \tag{3.2}$$

$$p_u(t) = p_0\big(1 + \mu_u(t)\big) \tag{3.3}$$

Dynamic spatially varying air cushion pressure represented by μ_{st}:

$$\mu_{st}(x,t) = \frac{p_{sp}(x,t)}{p_0} \tag{3.4}$$

$$p_{sp}(t) = p_0\mu_{sp}(x,t) \tag{3.5}$$

2.1 Wave elevation

To calculate the enclosed air cushion volume, the wave elevation inside the air cushion is needed. This wave elevation can be divided into four components:

- Incoming incident waves
- Diffraction waves due interaction between waves and the hull of the vessel
- Radiation waves, that is waves induced by vessel motion
- Waves induced by the air cushion pressure

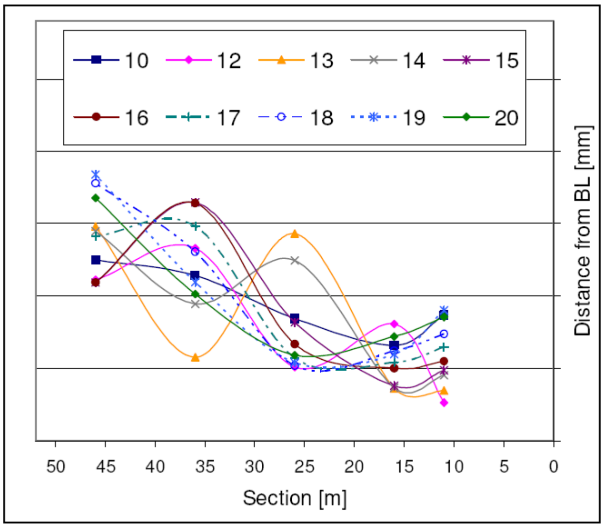


Figure 13: Wave pattern along the side hulls of a SES. Model testing. Variation of vessel speed in calm water [WINES et.al. 07]

The resulting surface will also be influenced by the vessel's speed and geometry. The first component is considered the most important. Diffraction and radiation waves are omitted while considered to lose the "time used vs. gain" competition. Instead of induced waves, the contribution from the air cushion pressure is simplified to a rectangular box with height equal to p0/ρg. As can be seen from Figure 13 (where incoming incident waves are not included), the total wave elevation is quite complex. But it is also seemingly quite static.

According to experiments, the diffraction waves are very important for the water surface [WINES 08]. However, Steen [STEEN 93] refers to work reporting good agreement between computations and experiments, without taking the diffraction effect into account. The reason is probably that for a given speed, the wave elevation is rather stationary, and therefore does not give large contribution to the seakeeping behavior. It is therefore considered ok to omit this, while other effects are more important to include.

2.2 Cobblestone oscillations

Cobblestone oscillations are high frequency resonance oscillations, and are one of the special attribute of the SES. They are caused by rapid changes in the air cushion pressure, and typically appear at small sea states. As the frequency is high the motion is not large, and cobblestone oscillations are therefore typically a comfort problem. The cobblestone oscillations can be divided into two types:

- Uniform pressure resonance, mainly resulting in heave motion
- Acoustic wave resonance, mainly resulting in pitch motion

Uniform pressure resonance is due to homogenous pressure changes, where the pressure changes in the entire air cushion. It is excited by incident waves, which alters the enclosed air cushion volume. The compressibility of the air in the cushion acts like a spring in a mass-spring system.

Acoustic wave resonance is caused by acoustic standing waves inside the air cushion. The interaction with the flexible bag-system in stern is here important [FALTINSEN 05, p.154], see chapter 2.3. The principle of standing waves is illustrated in Figure 14.

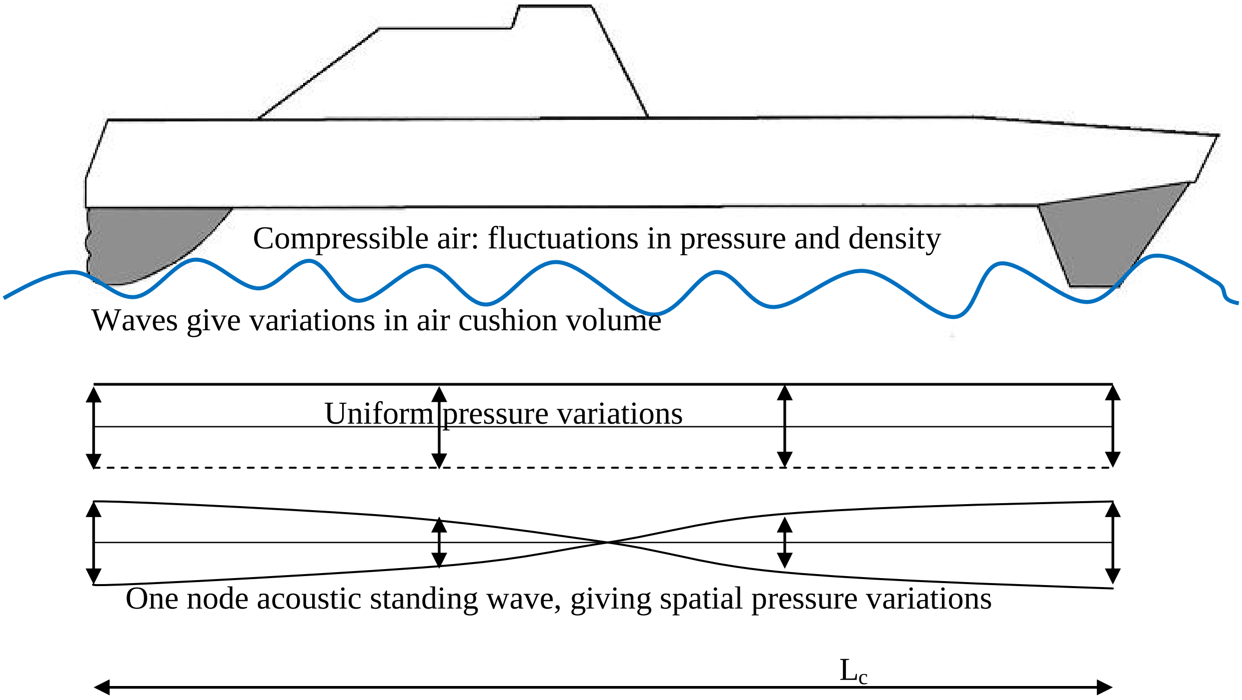


Figure 14: Cobblestone oscillations: Uniform and acoustic pressure variations

An example of power spectrum for cobblestone oscillations in heave is shown in Figure 15. The first peak at about 2 [hz] is due to uniform pressure resonance. The peak at about 5[hz] is due to acoustic wave resonance. Another example is given in Figure 20.

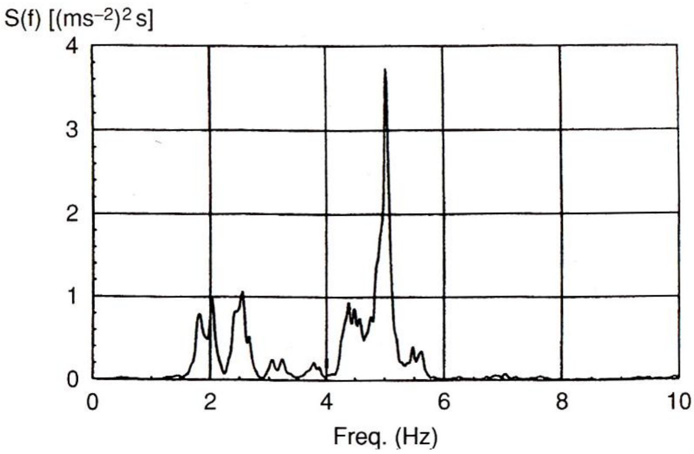


Figure 15: Example of power spectrum for cobblestone oscillations for a ship with length 35[m] [FALTINSEN 05 p.149]. Peak around 2 [Hz] due to uniform pressure variation, peak around 5 [Hz] due to spatial pressure variations

Steen [STEEN 93] concludes from measurements for a 35 [m] SES that uniform pressure variations dominate at frequencies below 4 [Hz]. Spatial pressure variations dominate beyond 5 [Hz], and both are present at 4-5 [Hz].

Faltinsen [FALTINSEN 05, p.150-153] shows how an expression for the uniform cobblestone natural frequency can be obtained. Hydrodynamic forces on the vessel are omitted in his derivation, and the air cushion equations are linearized.

In VERES and VeSim the air cushion pressure is assumed uniform throughout the air cushion. It should therefore be possible to obtain the resonant cobblestone oscillations due to uniform pressure variations. Faltinsen conclude with the following expression for the frequency for resonant cobblestone oscillations:

$$\omega_n = \sqrt{\frac{A_b \gamma (p_0 + p_a)}{M h_b}} \tag{3.6}$$

Where ω_n denotes natural frequency, M denotes weight of the vessel and A_b and h_b is defined as shown in Figure 16. γ is the specific heat for air.

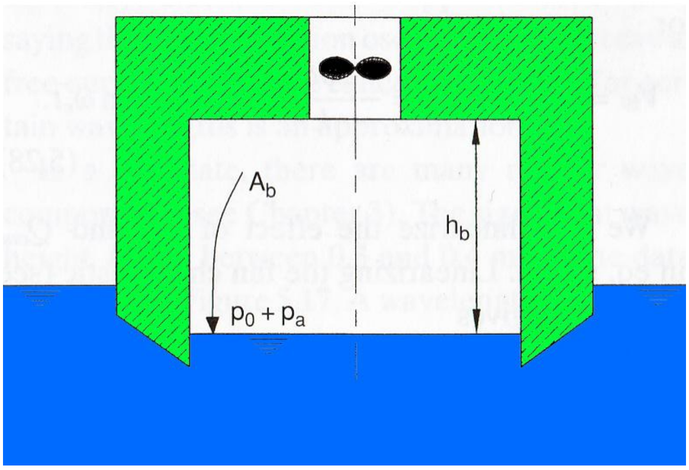


Figure 16: Definition of A_b and h_b [FALTINSEN 05, figure 5.20]. The figure is modified by the author of this report

Only homogenous air cushion pressure is used throughout this report. This means that no calculations will capture the effect of acoustic wave resonance. In accordance with Figure 14, the different resonance frequency nodes can be found by the following equation:

$$\omega_n = \frac{nc\pi}{L_c} \tag{3.7}$$

Where ω_n denotes the resonance frequency of the nth acoustic mode, n is the acoustic node, c is velocity of sound in air and L_c is the length of the air cushion. Acoustic node n has the value {1,2,…}. n equal to 1 gives the first node, which is a standing wave with wave length equal to $2L_c$ (see Figure 14). N equal 2 gives the second node with a wave length equal to L_c. It is generally so that the lower resonance frequency nodes dominate (n = 1).

With different vessel speed, the waves exciting cobblestone oscillations will vary. The exciting waves can be decided by using the frequency of encounter in combination with the dispersion relation. The dispersion relation for deep water reads:

$$\omega_0^2 = kg \tag{3.8}$$

And for shallow water:

$$\omega_0^2 = kg \tanh kh \tag{3.9}$$

Where h is water depth. The k used above in both equations is wave number. It is defined as

$$k = \frac{2\pi}{\lambda} \tag{3.10}$$

2.3 Fore and aft seals

The air cushion is sealed off in fore and aft. The devices with which this is done are commonly referred to as seals or skirts. There are several different designs for these sealing devices, with different effect on the vessel. There are for instance the difference between rigid panels and flexible bags. Here, the seal designs are limited to what dealt with in Sverre Steen's dr.ing thesis [STEEN 93]. These include the designs used in SES of the Norwegian Navy, namely the MCMV (Mine Counter Measure Vessel) and the Skjold class Fast Attack Craft. Information in this chapter is fetched from Steen's thesis [STEEN 93] and Faltinsen's Hydrodynamics of High-Speed Marine Vehicles [FALTINSEN 05].

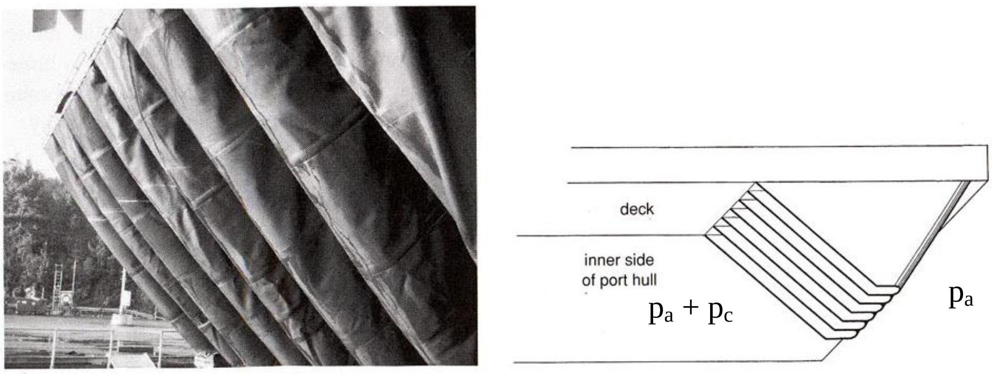


Figure 17: Fore seal system. Elastic finger skirt [FALTINSEN 05, figure 5.2 and 5.3]

The fore seal can be an elastic finger skirt design (see previous figures). The skirt consists of fingers hinged to the roof of the air cushion. On the inside there will be overpressure, giving a net pressure force forward (see Figure 17). With forward speed the fingers behave as a flag flapping in the wind, with a wear rate proportional to a high power of the vessel speed, U^4 [FALTINSEN 05, p147]. There will be air leakage from the air cushion underneath the seal and between the seal and the hull sides.

Two different types of aft seals were investigated by Steen [STEEN 93], flexible bag and one rigid panel design.

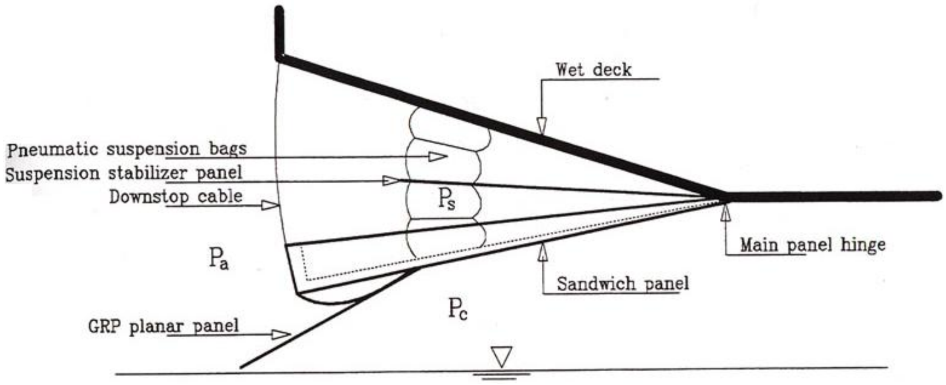


Figure 18: Aft seal system. Rigid planing panels [STEEN, figure 3.1]

The panel design consists of several sandwich panels mounted side by side. The GRP (Glass Reinforced Plastic) panels are planing on the water surface. Low stiffness of the plates allows them to follow the incoming waves within a certain limit. In larger waves the sandwich panels move to take up the wave motion.

The panels have quite large mass. They are built heavy and strong to resist the wave impacts in higher sea states. This large mass gives poor sealing ability in large sea states compared to the flexible bag system. The large mass also leads to large inertial forces when exposed to large motions in high sea states. This, together with loads from wave impacts, may lead to structural problems. According to Steen [STEEN 93, p37] the system was mounted on several commercial craft, but was later removed. This was due to problems with strength and durability of the sandwich panels and the main panel hinges in heavy seas.

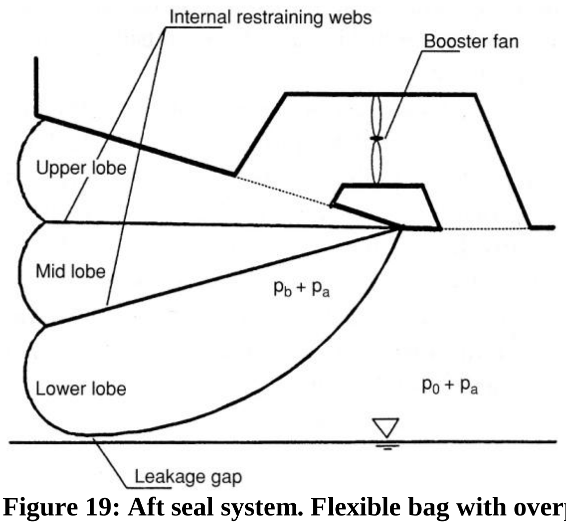


Figure 19: Aft seal system. Flexible bag with overpressure $p_b > p_0$ [FALTINSEN 05, figure 5.10]

The flexible bag aft seal consists of loops of a flexible material (normally reinforced rubber). The bag is shaped into two or three loops by internal webs. It is open towards the side hulls. There is overpressure inside the bag compared to outside and inside the air cushion. This forces the bag towards the water surface, giving a sealing effect. The low weight also help the bag follow the water surface, thus giving better sealing effect in large sea states compared to the rigid planing system.

In low sea states, there is a small gap between the bag and the water surface. This leads to a significant increase in high frequency vertical accelerations in low sea states compared to the rigid planing system [STEEN 93, p90]. The reason is variation in air gap with craft motions and incoming wave elevation.

The acoustic resonance frequencies occur at lower frequencies. This is due to the flexibility of the bag, which causes the particle velocity at the end of the air cushion to be non-zero. The acoustic waves in the air cushion are standing waves (see Figure 14) with fixed loops and nodes. The deflection of the bag moves the aft acoustical wave node from the cushion end to somewhere outside the cushion. It thereby increases the effective acoustic length [STEEN 93, p90]. Reviewing the equation for the acoustic resonance frequencies (eq 3.7), it is clearly seen that using an effective length larger than L_c decreased the resonance frequency.

The seal model used in the simulator is a very simplified version. Making complete realistic models of the different seal design was considered a vast task, and the gain of doing it was seen upon as too small to justify the effort. Both the fore and aft seal is therefore modeled as rigid plates, extending from the wet deck and down to a height given as input to the program. The bottom is just a straight edge between the demi hulls. This is a conservative approach, as this gives larger variation in air gap than using a dynamic seal model (larger variation in leakage gives higher high frequency vertical accelerations).

2.4 Ride Control System (RCS)

Ride control system allows for controlled leakages from the air cushion. The system consists of louvers that are controlled in order to give favorable leakages. This gives a very significant contribution in damping cobblestone oscillations, as seen in Figure 20.

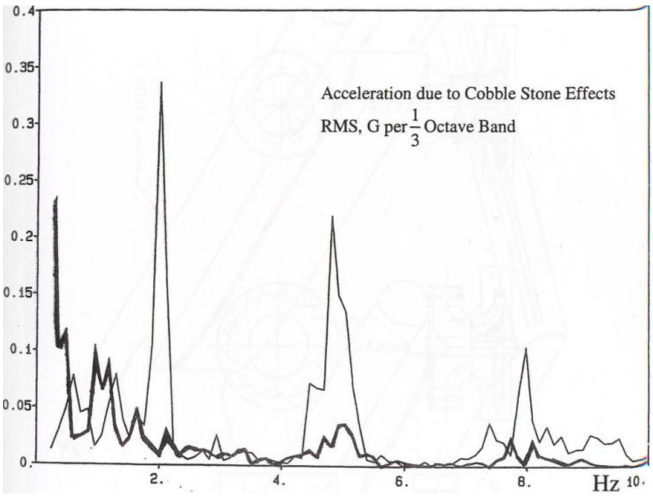


Figure 20: Power spectrum for vertical accelerations on a SES. Figure from [MINSAAS 06]. The peaks due to cobblestone oscillations are clearly visible. The broadest line shows the significant effect of the ride control system

As seen from the figure, the RCS reduces peaks due to both uniform and spatially varying air cushion pressure. To have an impact on the spatially varying pressure, it is important that the louvers are positioned as far fore and aft as possible. The same is true for position of the air fans.

Ride control is not included in the simulator. It is considered a secondary task, as the main goal is capturing the principal SES behavior.

2.5 Governing equations

Two equations govern the air cushion pressure.

1. Continuity equation for the air mass

$$\rho_a Q_{in} - \rho_a Q_{out} = \frac{d}{dt}(\rho_c \Omega) = \frac{d\rho_c}{dt}\Omega + \rho_c \frac{d\Omega}{dt} \tag{3.11}$$

   Where ρ_a is atmospheric density of air, Q is air flow in and out, ρ_c is density of air inside air cushion and Ω is air cushion volume.

2. Adiabatic relation between pressure and mass density (air is treated as ideal gas)

$$\frac{p}{p_0 + p_a} = \frac{p_a + p_c(t)}{p_0 + p_a} = \left(\frac{\rho_c}{\rho_a}\right)^{\gamma} \tag{3.12}$$

   γ is the specific heat for air, which is equal to approximately 1.4 [-].
   Adiabatic relation is justified by rapid pressure changes (too rapid to allow heat transfer)

The enclosed volume of the air cushion, Ω, is a function of wave elevation and vessel motion. Ω is found for each time step by numerically integrating the air cushion volume. The equation is:

$$\Omega(t) = \iint_{A_c} \left[\left(h_c(x,y) + \eta_3(t) + y\eta_4(t) - x\eta_5(t)\right) - T - \zeta(x,y,t)\right] dA \tag{3.13}$$

Here h_c is height from baseline to the roof inside the air cushion and T is the draught of the vessel. The lengths are shown in Figure 21. Motions of the vessel and wave elevation are inputs to the program from the CSI.

The wave elevation is expressed as a function of incident wave height and air cushion pressure.

$$\zeta = \zeta_{incident} - \frac{p_c}{\rho g} \tag{3.14}$$

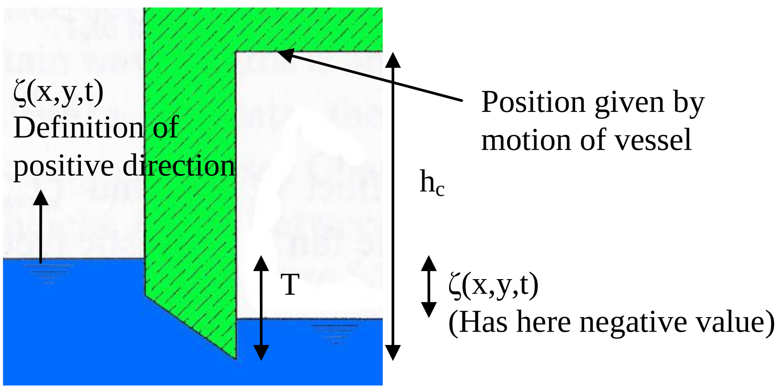


Figure 21: Definition of lengths used to calculate enclosed air cushion volume

$\frac{d\Omega}{dt}$ is the difference in air cushion volume between two succeeding time steps.

$$\left(\frac{d\Omega}{dt}\right)_n = \frac{\Omega_n - \Omega_{n-1}}{dt} \tag{3.15}$$

where dt is the constant time increment between each program run. n is run number.

The air flow in is due to fans, and the air out is due to leakages. The leakages can be somewhat controlled by valves. Equations for the two components are given as

$$Q_{in}(t) = Q_0 + Q_{inDyn}(t) \tag{3.16}$$

$$Q_{out}(t) = c_n A_L \sqrt{\frac{2p_c(t)}{\rho_a}} = c_n A_L \sqrt{\frac{2p_0\left(1 + \mu_u(t) + \mu_{sp}(x,t)\right)}{\rho_a}} \tag{3.17}$$

The components in equations for fans are shown in Figure 22. Q_0 is the mean flow rate due to air fans, c_n ($0 <= c_n <= 1$) is correction for orifice and A_L is leakage area. Leakage area is calculated for each time step in the program. Ride control can be incorporated into the leakage area (ride control is controlled motion damping. This is achieved by governing the air flow in and out of the air cushion. It might be incorporated into the model later).

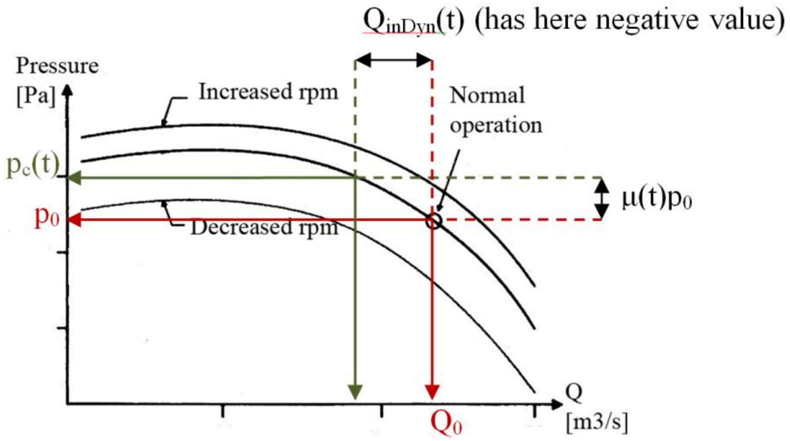


Figure 22: Example of fan characteristics [FALTINSEN 05, figure 5.6]. Normal operation is static condition, while the "example operation" shows the effect of increased air cushion pressure

Taylor expansion of the leakage gives:

$$Q_{out}(t) \approx c_n A_L \sqrt{\frac{2p_0}{\rho_a}} + c_n A_L \sqrt{\frac{2p_0}{\rho_a}} \frac{1}{2}\mu_u(t) + c_n A_L \sqrt{\frac{2p_0}{\rho_a}} \frac{1}{2}\mu_{sp}(x,t) \tag{3.18}$$

At steady state, Q_{out} should equal Q_{in}. Then the flow out should equal Q_0 plus something describing the change from static conditions. This can be described by giving A_L a static (A_{LStat}) and a dynamic part (A_{LDyn}).

$$A_L(t) = A_{LStat} + A_{LDyn}(t) \tag{3.19}$$

where A_{LStat} can be solved from the static conditions

$$A_{LStat} = Q_{out}\frac{1}{c_n}\sqrt{\frac{\rho_a}{2p_0}} = Q_0\frac{1}{c_n}\sqrt{\frac{\rho_a}{2p_0}} \tag{3.20}$$

New equation for the leakage is

$$\begin{aligned} Q_{out}(t) \approx & \underbrace{Q_0 + c_n A_{LDyn}(t)\sqrt{\frac{2p_0}{\rho_a}}}_{\substack{leakage\ due\ to \\ constant\ overpressure}} \\ & + \underbrace{c_n\left(A_{LStat} + A_{LDyn}(t)\right)\sqrt{\frac{2p_0}{\rho_a}}\frac{1}{2}\mu_u(t)}_{leakage\ due\ to\ dynamic\ uniform\ overpressure} \\ & + \underbrace{c_n\left(A_{LStat} + A_{LDyn}(t)\right)\sqrt{\frac{2p_0}{\rho_a}}\frac{1}{2}\mu_{sp}(x,t)}_{leakage\ due\ to\ dynamic\ spatially\ varying\ overpressure} \end{aligned} \tag{3.21}$$

Inserting into continuity equation gives

$$\begin{aligned} & Q_{inDyn}(t) - c_n A_{LDyn}\sqrt{\frac{2p_0}{\rho_a}} \\ & \quad - c_n\left(A_{LStat} + A_{LDyn}\right)\sqrt{\frac{2p_0}{\rho_a}}\frac{1}{2}\left[\mu_u(t) + \mu_{sp}(t)\right] \\ & \quad = \frac{d\rho_c}{dt}\Omega + \rho_c\frac{d\Omega}{dt} \end{aligned} \tag{3.22}$$

2.6 Air cushion pressure

2.6.1 No air entering or leaving the air cushion

The program includes the possibility of doing calculations with no air entering/leaving the air cushion. There is only uniform air cushion pressure. The overpressure inside the air cushion is then defined as:

$$p_c(t) = p_u(t) \tag{3.23}$$

Where p_u denotes uniform dynamic air cushion pressure.

The continuity equation becomes

$$\frac{d}{dt}\left(\rho_c(t)\Omega(t)\right) = 0 \tag{3.24}$$

Integrating this gives the air cushion air density and air cushion volume as:

$$\rho_c(t)\Omega(t) = \mathrm{C} \tag{3.25}$$

C is a constant, which can be calculated by initial conditions of no motion and no waves. When there are no waves and motions, the overpressure inside the air cushion equals the static overpressure p_0. The air cushion volume is found by

$$\Omega_0 = \iint\limits_{A_c} \left[\mathrm{h_c}(\mathrm{x},\mathrm{y}) - \mathrm{T} + \frac{\mathrm{p_0}}{\rho \mathrm{g}}\right] \mathrm{dA} \tag{3.26}$$

Where Ω_0 denotes static air cushion volume and $p_0/(\rho g)$ is wave elevation due to static air cushion pressure. The static air mass density, p_{c_0}, inside the air cushion can be found from the adiabatic relation between air cushion pressure and mass density.

$$\frac{p_0 + p_a}{p_0 + p_a} = \left(\frac{\rho_{c\_0}}{\rho_a}\right)^{\gamma} \Rightarrow \rho_{c\_0} = \rho_a \tag{3.27}$$

Then the constant C is given by:

$$c = \rho_c(t)\Omega(t)|_0 = \rho_a\Omega_0 \tag{3.28}$$

The dynamic pressure can be solved from the same adiabatic relation, only this time with dynamic overpressure, air cushion volume and air density inside air cushion:

$$\begin{aligned} \frac{p_0 + p_a + p_0\mu_u(t)}{p_0 + p_a} = \left(\frac{\rho_c(t)}{\rho_a}\right)^{\gamma} &= \left(\frac{C}{\rho_a\Omega(t)}\right)^{\gamma} \\ &= \left(\frac{\rho_a\Omega_0}{\rho_a\Omega(t)}\right)^{\gamma} \end{aligned} \tag{3.29}$$

$$p_0\mu_u(t) = (p_0 + p_a)\left[\left(\frac{\Omega_0}{\Omega(t)}\right)^{\gamma} - 1\right]$$

2.6.2 Uniform air cushion pressure, air entering/leaving the air cushion

The overpressure in the air cushion is here defined identical as in last chapter.

$\frac{d\rho_c}{dt}$ can now be solved from the adiabatic condition. First rearranging the equation and Taylor expansion (linearizing):

$$\frac{\rho_c}{\rho_a} = \left(1 + \frac{\mu_u(t)p_0}{p_0 + p_a}\right)^{\frac{1}{\gamma}} \approx 1 + \frac{1}{\gamma}\frac{\mu_u(t)p_0}{p_0 + p_a} \tag{3.30}$$

Differentiating with respect to time:

$$\frac{d\rho_c}{dt} = \frac{\rho_a}{\gamma}\frac{p_0}{p_0 + p_a}\frac{d\mu_u}{dt} \tag{3.31}$$

Air leakage now becomes:

$$\begin{aligned} &Q_{out}(t) \\ &\approx \underbrace{Q_0 + c_n A_{LDyn}(t)\sqrt{\frac{2p_0}{\rho_a}}}_{\substack{leakage\ due\ to \\ constant\ overpressure}} \\ &+ \quad \underbrace{c_n\left(A_{LStat} + A_{LDyn}(t)\right)\sqrt{\frac{2p_0}{\rho_a}}\frac{1}{2}\mu_u(t)}_{leakage\ due\ to\ dynamic\ uniform\ overpressure} \end{aligned} \tag{3.32}$$

Combining the different expressions gives:

$$\begin{aligned} Q_{inDyn}(t) - c_n A_{LDyn}\sqrt{\frac{2p_0}{\rho_a}} - c_n\big(A_{LStat} + A_{LDyn}\big)\sqrt{\frac{2p_0}{\rho_a}}\frac{1}{2}\mu_u(t) \\ = \frac{\rho_a}{\gamma}\frac{p_0}{p_0 + p_a}\frac{d\mu_u}{dt}\Omega + \rho_a\left(1 + \frac{1}{\gamma}\frac{\mu_u(t)p_0}{p_0 + p_a}\right)\frac{d\Omega}{dt} \end{aligned} \tag{3.33}$$

Only unknown is $\mu_u(t)$. This gives:

$$\frac{d\mu_u(t)}{dt} + A\mu_u(t) = B \tag{3.34}$$

Where

$$\begin{aligned} A &= \frac{\gamma}{\rho_a}\frac{p_0 + p_a}{p_0}\frac{1}{\Omega}\left\{c_n\big(A_{LStat} + A_{LDyn}\big)\sqrt{\frac{2p_0}{\rho_a}}\frac{1}{2} + \frac{\rho_a}{\gamma}\frac{p_0}{p_0 + p_a}\frac{d\Omega}{dt}\right\} \\ &= \frac{\gamma}{\rho_a}\frac{p_0 + p_a}{p_0}\frac{1}{\Omega}c_n\big(A_{LStat} + A_{LDyn}\big)\sqrt{\frac{2p_0}{\rho_a}}\frac{1}{2} + \frac{1}{\Omega}\frac{d\Omega}{dt} \end{aligned} \tag{3.35}$$

$$B = \frac{\gamma}{\rho_a}\frac{p_0 + p_a}{p_0}\frac{1}{\Omega}\left\{Q_{inDyn}(t) - c_n A_{LDyn}\sqrt{\frac{2p_0}{\rho_a}} - \rho_a\frac{d\Omega}{dt}\right\} \tag{3.36}$$

The equation is solved by using the integration factor

$$p(t) = \int_{t_{n-1}}^{t_n} A\,dt \tag{3.37}$$

$$\mu_u(t) = e^{-p(t)}\int_{t_{n-1}}^{t_n} Be^{p(t)}dt \tag{3.38}$$

2.6.3 Spatially varying air cushion pressure

The main drawback of using uniform air cushion pressure is that pitch forces from longitudinal distribution of air cushion pressure are missing. In fact, with an air cushion that is symmetric about amidships, the pitch forces become zero. However, the air moves very fast (speed of sound: c = 340 [m/s]), so an uneven pressure distribution will quickly be neutralized. The pitch forces resulting from spatially varying air cushion pressure is thus of high frequency, and it is those resulting in acoustic wave resonance (see chapter about cobblestone oscillations, chapter 2.2) that are of importance. It is this phenomenon that promotes introducing spatially varying air cushion pressure in VeSim.

Quite some time was spent on finding a way to calculate spatially varying air cushion pressure. It was however recommended by Odd M. Faltinsen [FALTINSEN 08] to abandon this in favor of added resistance calculations (see chapter 3). This advice was followed up. The arguments are that making a good time-domain numerical tool for this would probably be very complex . There also exist rather good linear calculation tools for this. And last, the importance of the results is not that great as good ride control systems exists to counter act the oscillations. Therefore the added resistance problem was more interesting.

This chapter will only give a brief introduction to how the spatially varying air cushion pressure can be interpreted. Some definitions of Sverre Steen [STEEN 93] will be shown.

Generally, the importance of spatially varying pressure increases with increasing vessel length. VERES omit spatially varying air cushion pressure.

Modeling the spatially varying pressure could have been done in several different ways. One could for instance use a finite or boundary element method. During this thesis work, a 2D boundary element model was roughly sketched. The model is presented in appendix A.2.

Sverre Steen presents in his Dr.ing thesis [STEEN 93] a modal approach to decide the air cushion pressure. The benefits of a modal approach is that it is relatively simple compared to an element model, it can be calculated fast in numerical calculations and it captures the most important feature; difference between air cushion pressure fore and aft in the air cushion. The thesis shows good results using modal approach.

The pressure is only considered to vary in longitudinal direction. This is justified by the large length to beam and height ratio of the air cushion. The spatially varying air cushion pressure is presented by mode shape functions. These functions are chosen so they satisfy the boundary conditions on the seals. One solution is the following functions:

$$r_j(x) = \cos\left(\frac{j\pi}{L_c}\left[x + \frac{L_c}{2}\right]\right), \quad x \in \left[-\frac{L_c}{2}, \frac{L_c}{2}\right], \ j = 1,2,3, \dots \tag{3.39}$$

These follow an air cushion-centered coordinate system. The origin is in the geometrical center of the air cushion water plane area. The x-axis is positive forward, the y-axis towards starboard and the z-axis upwards. This system is also used in the air cushion program (to ease introduction of spatially varying air cushion pressure in the calculations). The solution will be on the form:

$$\mu_{sp}(x;\omega_e) = \sum_{j=1,2,3,\ldots} A_j(\omega_e) r_j(x) \tag{3.40}$$

Where Aj is a frequency dependent amplitude. It is shown by Steen [STEEN 93] that it is the first mode shape function that is the most important one. This gives the largest pitch force and the most important pitch cobblestone oscillation. The first two functions are shown in Figure 23.

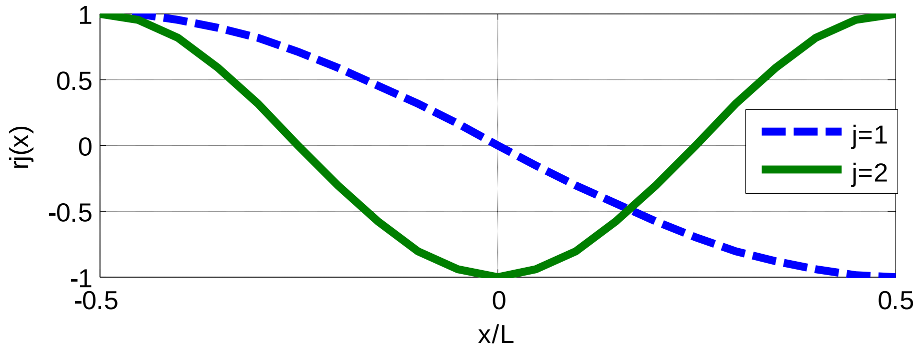


Figure 23: The two first mode shape functions, j=1 and j=2

According to the mode shape theory, the spatially varying air cushion is presented by “standing air waves” (illustrated in Figure 23). That is, the air travels through the air cushion at the speed of sound. Using this idea, the belonging natural frequency to each mode shape is given by

$$\omega_j = c\frac{j\pi}{L_c} \tag{3.41}$$

With c being the velocity of sound in air.

Important factors to decide the spatially varying air cushion pressure is:

- Fans and the longitudinal position of them
- Leakages and the longitudinal positions
- Control actions, i.e. control of louvers and fans
- The uniform air cushion pressure
- Longitudinal change of the air cushion volume. That is, wave elevation and vessel motion is important

The theories of Steen [STEEN 93], which are further developed by Tore Ulstein [ULSTEIN 95], are linear theories. Each wave component is dealt with separately. In the simulator this is currently not possible. Only the resulting wave elevation is possible to obtain. This means that their equations cannot be used directly, and is one of the reasons that the spatially varying air cushion lost the gain/effort contest.

Combining uniform and spatially varying air cushion pressure gives the air cushion over pressure as:

$$p_c(x,t) = p_0 \underbrace{[1+\mu_u(t)}_{\substack{uniform \\ air\ cushion\ pressure}} + \underbrace{\mu_{sp}(x,t)]}_{\substack{spatially\ varying \\ air\ cushion\ pressure}} \tag{3.42}$$

2.7 Force calculation

When the air cushion pressure is known, the forces and moments are calculates using equations:

$$\boldsymbol{F_c}(t) = -\iint_S p_0\left(1 + \mu(t) + \mu_{sp}(x,t)\right)\boldsymbol{n}dS \qquad (3.43)$$

and

$$\boldsymbol{M_c}(t) = -\iint_S p_0\left(1 + \mu(t) + \mu_{sp}(x,t)\right)(\boldsymbol{r} \times \boldsymbol{n})dS \qquad (3.44)$$

n denotes the normal vector on the hull surface towards cushion (local coordinates) and S is the hull surface. **r** is a vector containing the distance to the point about where the moments are calculated. Boundary of the hull surface is the free surface inside the cushion, given by z = ζ(x,y,t). The water surface includes wave elevation due to air cushion pressure.

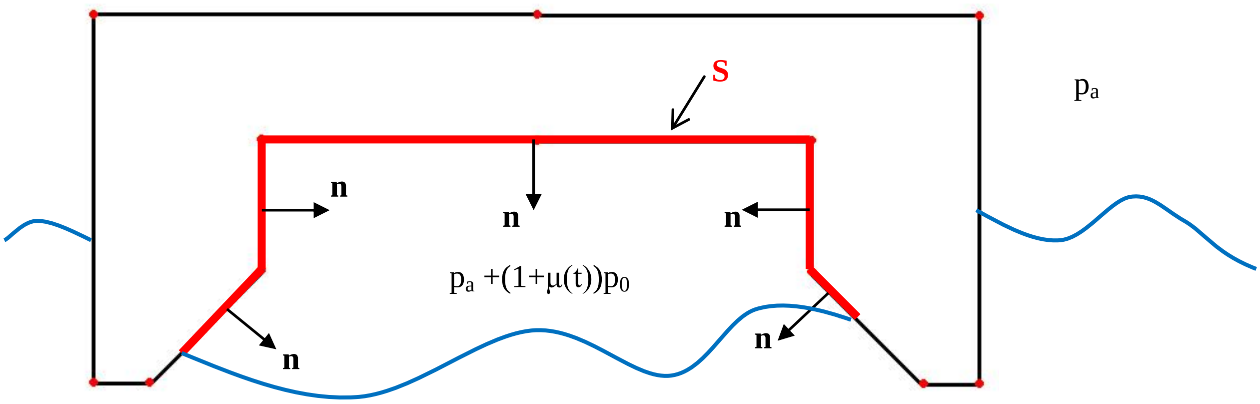


Figure 24: Hull surface used when calculating air cushion forces

It is possible divide the air cushion forces into damping and restoring forces, as discussed 1.5.3 with the hydrodynamic forces. Then the air flow in and out of the cushion gives damping forces. The air cushion itself works as a spring (change of volume gives forces) and gives thus restoring forces. Excitation forces are alteration of the air cushion volume, which is made by motion of the vessel and change of wave elevation.

3 Resistance theory

One of the main advantages of the SES is the resistance. It has, compared to other vessel types, low resistance at high vessel speed. It has also one drawback, and that is rather large involuntary speed loss in high sea states, see Figure 25 and Figure 26. The figure shows that the speed loss for a SES is larger than that for a catamaran.

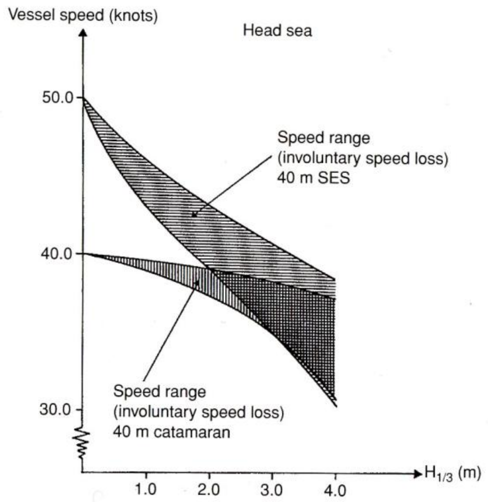


Figure 25: Involuntary speed loss in waves for SES and catamaran [FALTINSEN 05, figure 5.27]

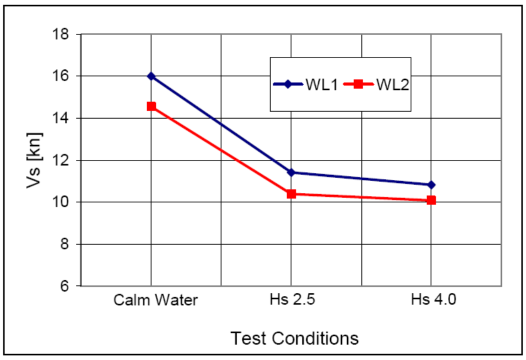


Figure 26: Example of speed loss in waves, the Alta class Mine Counter Measure Vessel (MCMV). The plots are for two different load conditions [WINES et.al. 07]

For an SES, involuntary speed loss is due to:

- Added resistance in waves
- Shallow water effects
- Reduced efficiency of the propulsor due to ventilation of air inlet for water jet
- Wind
- *Increased air leakage from the air cushion due to vessel motion/large waves*

It is the last effect (italicized in the list) that is thought to be the main contributor that makes the larger speed loss for the SES. This thesis work looks on the problem and tries to develop a method to calculate this increased resistance.

The ship resistance can be divided into different components. Then the total resistance is the sum of all the components. The general calm water case could be divided into the following main resistance components:

$$R_T = R_P + R_V + R_A \tag{3.1}$$

Table 1: Main resistance components of a SES

Resistance components	*Description*
R_T	Total vessel resistance
R_P	Pressure related resistance
R_V	Viscous resistance
R_A	Air resistance

The largest contributors to pressure resistance is wave making. Both the hull, the air cushion pressure field and fore and aft seal makes their own wave systems. Viscous resistance is frictional resistance, and its main parameter is how much wet surface that is present. The large benefit of the SES is little wet surface and therefore less viscous resistance than other vessel types. Air resistance is normally friction due to the superstructure. For a SES, air momentum drag is in addition. This is caused by change of momentum in the air used to lift the craft (change of momentum when the air enters through fans and when it exits by leakage).

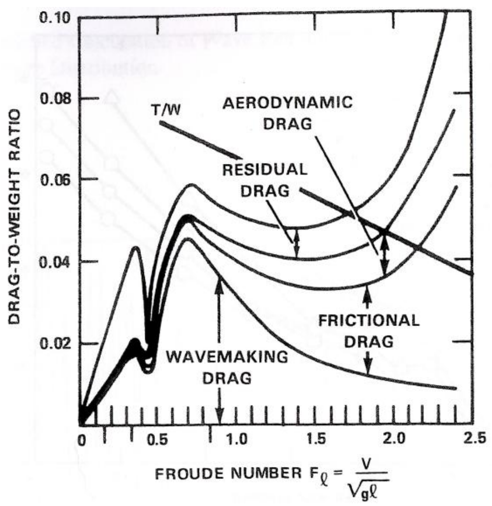


Figure 27: The different resistance component for a SES with l/b ratio of 2.0 [MINSAAS 06, figure 135.b]

Figure 27 shows a typical calm water resistance curve for a SES. Residual drag resistance is connected to pressure. The humps in the beginning are due to air cushion induced wave making, and are characteristic for the SES resistance curve. It is clearly seen that at large vessel speeds, the viscous resistance is most important. At lower vessel speed, wave making is more important. This means that for lower vessel speeds, a catamaran will have lower resistance than a SES (a SES makes more waves while it is wide and the air cushion pressure induces waves). At higher speeds the SES will have less resistance. The SES is designed for high speeds, and the importance of viscous resistance calculations is therefore obvious.

3.1 Viscous resistance

This report will greatly simplify all other resistance components than the viscous resistance. As the seakeeping model was developed, Faltinsen [FALTINSEN 08] suggested utilizing it on one of the drawbacks of the SES: Speed loss in waves. The physics is that when there are large waves, the leakage will increase. Increased leakage gives increased submergence, which then gives larger wet surface and therefore larger viscous resistance. This effect is not easily calculated using linear theory. The hope is that the time domain theory used in the VeSim can be used to fetch the effect. There are also other effects giving increased resistance in waves, but the increased viscous resistance problem is here most interesting. It is directly connected to the seakeeping of the SES, and is also unique for the SES design.

Viscous resistance can be described using a viscous resistance coefficient C_V:

$$R_V = \frac{1}{2}\rho U^2 S C_V \tag{3.2}$$

U denotes vessel speed (in [m/s]) and S is wetted surface. The viscous resistance coefficient is found by:

$$C_V = (1+k)(C_F + \Delta C_F) \tag{3.3}$$

k denotes form factor, C_F is frictional resistance and ΔC_F is correction for roughness. C_F is found using ITTC-57 correlation line. This is an empirical formula based on Reynold's number:

$$R_n = \frac{U L_{pp}}{\nu} \tag{3.4}$$

Where υ is kinematic viscosity coefficient. The ITTC-57 correlation line is based on a flat plate. This is the reason for the form factor k (which there exists several ways to estimate, which none of will be discussed here). The correction for roughness can be found using an empirical equation:

$$\Delta C_F = [110.31(H \cdot U)^{0.21} - 403.33] \cdot {C_F}^2 \tag{3.5}$$

H denotes hull roughness factor, and should be in the order of magnitude 75-200 [μm] [MINSAAS 06, p.47]. A low value should be chosen for a high speed vessel as the SES. If ΔC_F ends up negative, the component is neglected. The ITTC-57 correlation line is:

$$C_F = \frac{0.075}{(\log_{10} R_n - 2)^2} \tag{3.6}$$

3.2 Resistance in VeSim

The resistance in VeSim is fed using a polynomial based on vessel speed. This is the total calm water resistance in longitudinal direction. For vessel speed, meeting particle velocity at amidships is used. This velocity includes contributions from waves, surge speed and current. In addition there are air resistance and drift forces, but these are here omitted.

The total vessel resistance polynomial is simplified to a linear curve, having roughly correct magnitude of force. The only interesting term is the viscous resistance, and change of this in relation with vessel speed and sea state. That is, the resistance problem is described by:

$$R_T = C \cdot U^2 + \Delta R_V = R_V + R_{OTHER} \tag{3.7}$$

Where C is a constant [Ns^2/m^2] and ΔR_V is change in viscous resistance, i.e. viscous speed loss in waves. R_{OTHER} is resistance that is not related to viscous resistance. Viscous resistance is further divided:

$$R_V = \frac{1}{2}\rho U^2 (S_0 + \Delta S) C_V \tag{3.8}$$

S_0 denotes static wet surface, and ΔS is the dynamic change in wet surface. Now, at every time step, added resistance is calculated by:

$$\Delta R_V = \frac{1}{2}\rho U^2 \Delta S C_V \tag{3.9}$$

Relative importance is simply:

$$\delta = \frac{\Delta R_V}{R_V} = \frac{\frac{1}{2}\rho U^2 \Delta S C_V}{\frac{1}{2}\rho U^2 (S_0 + \Delta S) C_V} = \frac{\Delta S}{S_0 + \Delta S} = \frac{S - S_0}{S} = 1 - \frac{S_0}{S} \tag{3.10}$$

Point of attack of the viscous resistance is for simplicity set at x = lpp/2, y = 0 and z = 0.5 *static draught. It is also defined to work strictly in local surge direction.

4 VeSim – Vessel Simulator

VeSim is a time-domain vessel calculation tool developed by Marintek. It can be applied to all sorts of calculations. Marintek has to this day developed the simulator to cope with normal displacement vessel. The task for this master thesis is to extend it for use with SES as well.

4.1 VeSim architecture

The architecture of the Marintek Vessel Simulator (VeSim) is based around a server called Common Simulation Interface (CSI) (see Figure 29). The CSI connects the different simulation programs by synchronizing their time and exchanging data. The simulation programs are called simulation federates, and it is one of these that the author has developed.

Inputs to the simulator are made using the different plug-ins in ShipX. ShipX is a hydrodynamic workbench that holds the geometry of the vessels. Then plug-ins, as VERES, uses this data to do various calculations.

When inputs are made, the simulator can be started individually or from ShipX. What happens in the simulator is visualized by the web server and SimVis (visualization program). Both are federates in the VeSim system. Results from calculations (e.g. vessel motions) are shown in the web server. The motions of the vessel and the wave surface are shown in SimVis. See Figure 28

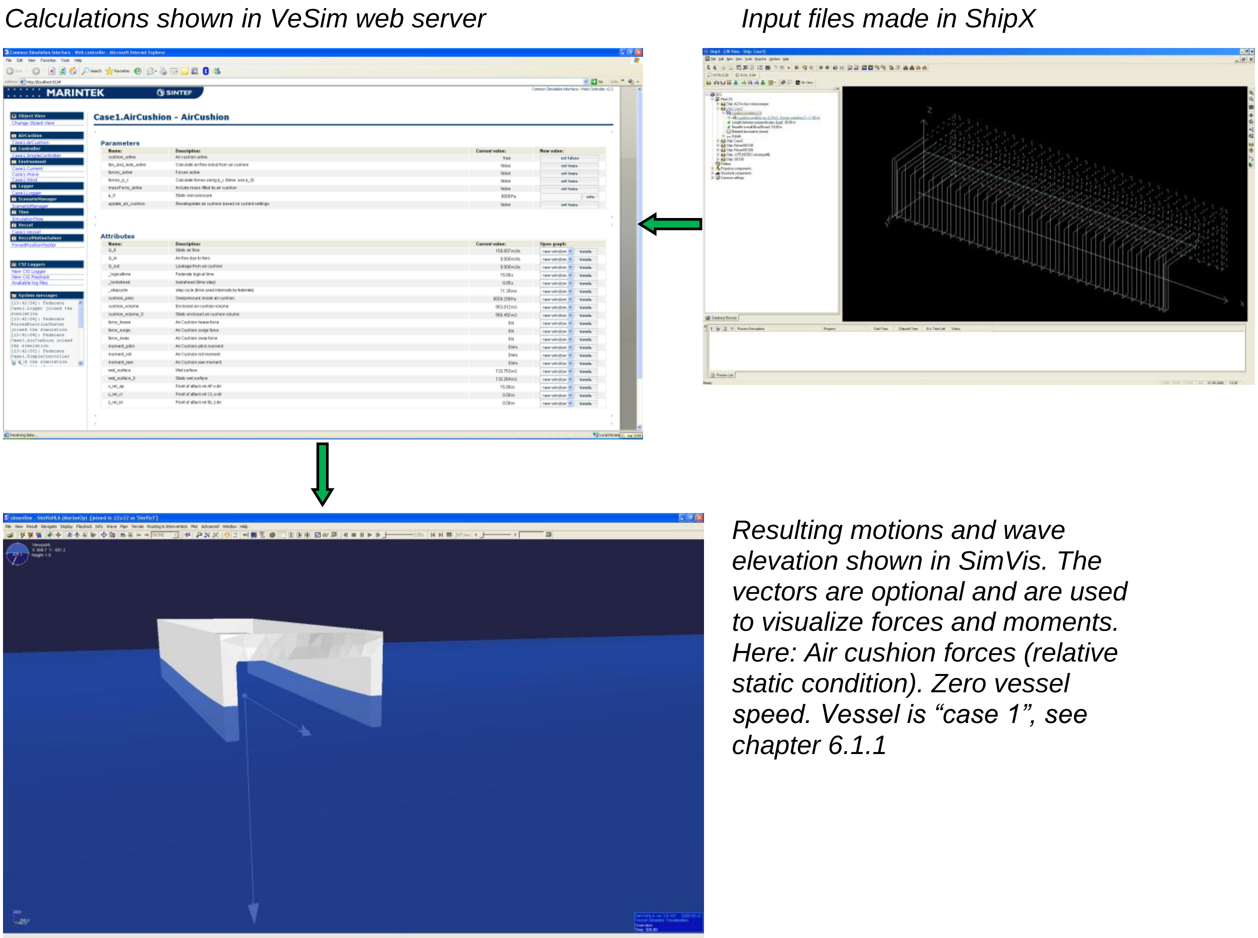


Figure 28: Interfaces encountered using VeSim. Inputs are made in ShipX. The calculations are shown in the WEB interface. Resulting motions and wave elevation are shown in SimVis

The following figure and table show an example of which federates could be used in the setup for an analysis. A run may have more or less federates than shown in the figure.

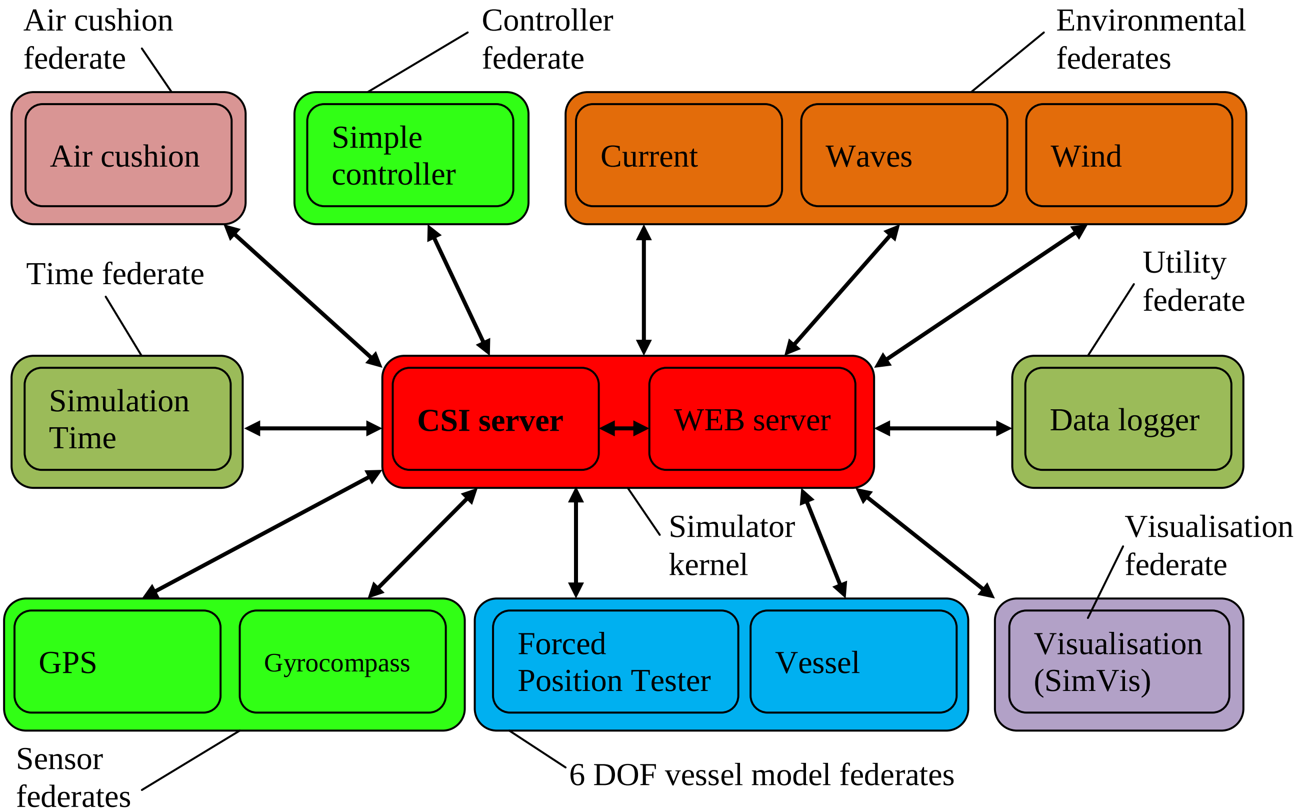


Figure 29: Example of the architecture of VeSim. Arrows show flow of information

Table 2: Short description of the different parts in the VeSim architecture

Federate	***Federate type***	***Description***
Air cushion	Air cushion	Calculates forces due to an air cushion. Calculates added viscous resistance in waves
Vessel	6 DOF Vessel model	Reads in all forces. Main outputs are total forces and 6 DOF motions. Also calculates viscous forces and roll damping, retardation functions (added mass and damping) and restoring forces
Forced Position Tester	6 DOF Vessel model	Alternative to the motion solver. This locks the position of the vessel to user defined values. Utilized to verify calculations
Current	Environmental	Calculates forces due to current
Waves	Environmental	Calculates forces due to waves. Gives wave elevation
Wind	Environmental	Calculates forces due to wind
Simulation Time	Time	Keeps track of time
CSI server	Simulator kernel	Connects the different federates by synchronizing their time and exchanging data
WEB server	Simulator kernel	The interface of the simulator. Feeds dynamic web pages with the data information available
Data logger	Utility	Stores data for later processing
GPS	Sensor	Input is data from vessel. Main outputs are north/south and east/west coordinates
Gyrocompass	Sensor	Input is data from vessel. Outputs are gyrocompass data
Simple controller	Controller	Reads data from vessel. Calculates forces in order to keep given heading and velocity
SimVis	Visualization	Displays the vessel’s movements and the wave environment

A federate can have two kinds of variable: attributes and parameters. Parameters are settings for the federate (e.g. wave heading for the wave federate) and attributes are data it sends out that are possible for the other federates to obtain (e.g. forces due to current). The attributes can be obtained by the other federates using so-called “subscriptions”. The flow is sent from the federate giving the attribute to the CSI server. The CSI server passes the data on to the subscribing federate. This structure is one of the main advantages of the CSI, allowing the programmer to build federates independently of the other federates and calculations that are done. The different federates may also freely enter and exit the system during the run.

4.1.1 Necessary input

VeSim needs inputs from the following plug-ins. They are all plug-ins in ShipX.

- VERES (Vessel Response program)
- Maneuvering Calculation (hullvisc)
- Panel Generator
- Ship Model for the Vessel Simulator
- Setup for the Vessel Simulator

VERES is the main calculation tool. It supplies most (linear) hydrodynamic properties. This includes mass and restoring properties, damping and added mass. The “Ship Model for the Vessel Simulator” calculates retardation functions from linear damping and added mass. Retardation functions are “translations” from linear coefficients in the frequency plane. They take into account the load history. See [HYDRO VESIM 04].

The maneuvering run gives linear maneuvering properties. That is cross-flow drag, speed-dependent calm water resistance and various hydrodynamic maneuvering coefficients. Unfortunately, the maneuvering plug-in is not designed for multi-hull vessels. Maneuvering coefficients were therefore here found using another vessel (a dummy vessel). The output was then modified some to fit better (most of coefficients were sat equal to zero). This solution to the problem is not a good solution. For instance, as the coefficients include cross-flow drag, these are bound to be incorrect. The vessel also need to be the same length (lpp) and weight as otherwise the forces will be of wrong magnitude. Weight is though easily changed in the output file. Length (that is used to divide the hull into sections) is more complicated.

The panel generation makes a 3D model of the geometrical data. It is used for non-linear forces and visualization (optional) in SimVis.

The “Ship Model for the Vessel Simulator” creates a ship model archive. This contains all input from VERES, maneuvering and the panel generation. Propulsors, sensors and similar are also added in “Ship Model for the Vessel Simulator”.

The Vessel Simulator setup run is used to give environmental parameters (wind, waves and current). Other federate parameters can also be set here. It creates the *.xml files (process.xml and federation.xml). In addition, this plug-in can be used to launch VeSim and SimVis.

The two *.xml documents, “federation.xml” and “process.xml”, governs the simulator setup. “federation.xml” states what parameters and attributes each federate uses. It also tells how the federates interact with each other. “process.xml” governs the startup and termination of the simulation. It tells which federates to start (federates can also be started manually later), and can also start optional postprocessors after termination. The *.xml files are automatically generated by the "Setup for the Vessel Simulator" plug-in in ShipX. They can (and must in the case of SES) however be edited manually afterwards.

The parts of the *.xml files dealing directly with the air cushion federate are shown in Appendix B.1 and Appendix B.2.

4.2 Forces in VeSim

4.2.1 Origin of forces

Force and motion calculations are centered on the vessel federate. It solves the vessel motions in 6 degrees of freedom, thus holding all terms on the left hand side in the equations of motions (see equation 1.39). These terms include mass and restoring coefficients, damping and added mass (the last two represented as retardation functions). Also all forces from the maneuvering run are handled by the vessel federate.

"External" forces from the other federates (e.g. controller forces, rudders, propulsion, waves, air cushion) are sent to the vessel federate.

The wave federate provides wave excitation forces. These are: first order wave excitation, wave drift forces (these are omitted throughout the thesis work) and non-linear modification of restoring and incident wave forces (NMRIWF). The NMRIWF takes into account the actual geometry of the vessel (NMRIWF omitted in this thesis work). The input is the panel generation run, which gives a 3D panel model of the entire hull (over and above waterline). The output forces are Froude- Kriloff and restoring forces based on the actual hull geometry and position and properties of the incident waves. See Figure 31 and Figure 32.

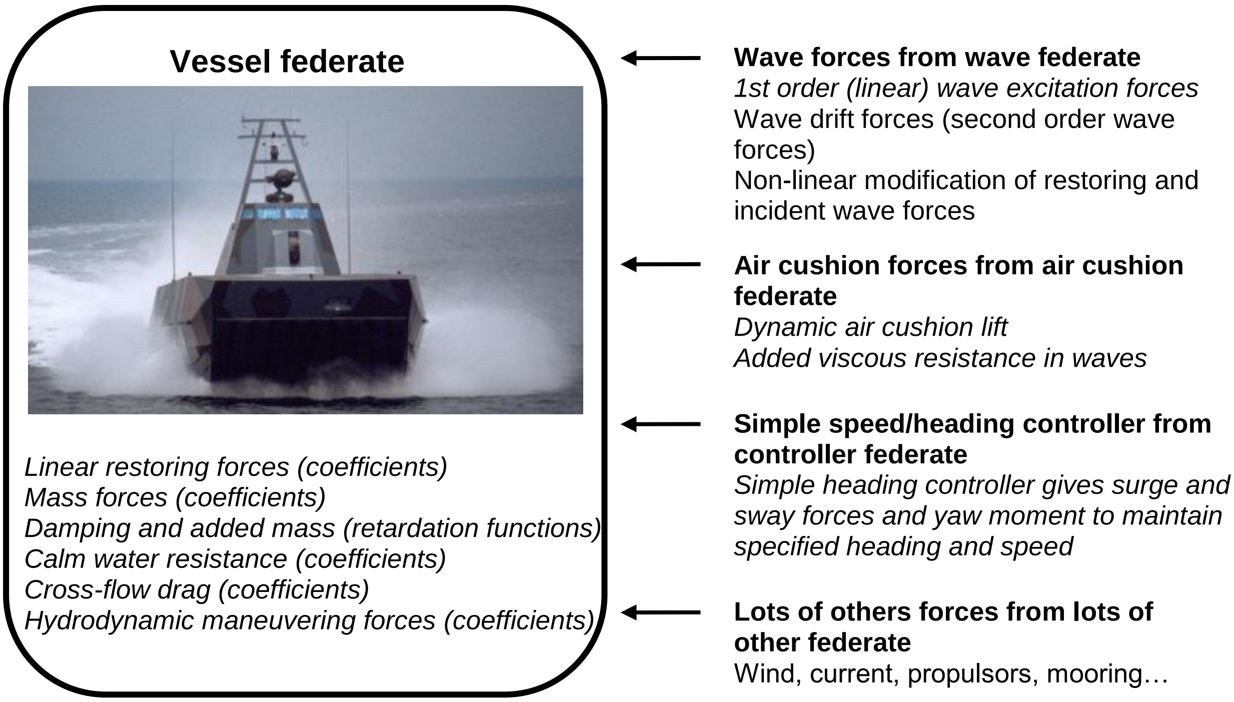


Figure 30: Origin of forces in VeSim. Forces used in the setups used in this report's calculations are written in italic

4.2.2 Non-linear and linear forces

VeSim applies a two-step approach in calculating forces. First linear hydrodynamics coefficients are calculated in the frequency domain (using ShipX (VERES and Maneuvering plug-ins)). These are found using strip-theory, and are function of vessel geometry (submerged part), vessel motion, vessel heading and wave condition. Forces from these coefficients are calculated in the time domain. In addition, dominant non-linear contributions are added.

An attempt to show how the different forces are deduced is shown in next two figures.

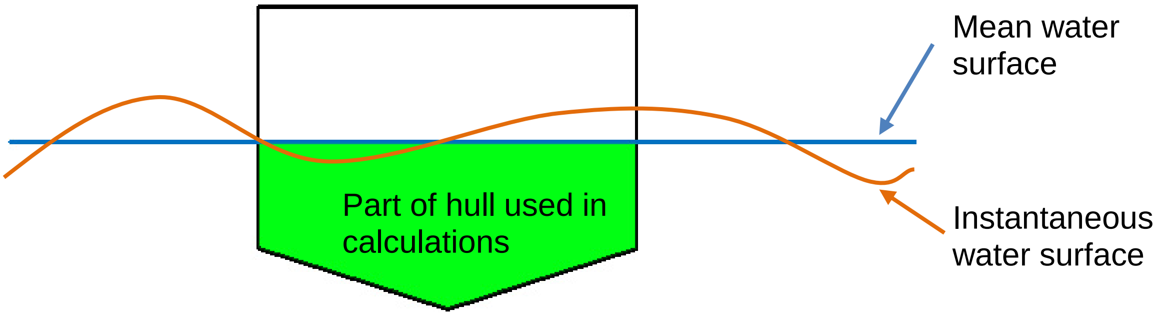


Figure 31: Forces calculated using mean water surface: Diffraction, added mass, hydrodynamic damping

The green area corresponds to the area used in strip theory. In linear theory, all forces are calculated based on this area. In time-domain calculations, the green area is used to obtain hydrodynamic coefficients (diffraction forces (first order wave forces), added mass, hydrodynamic damping).

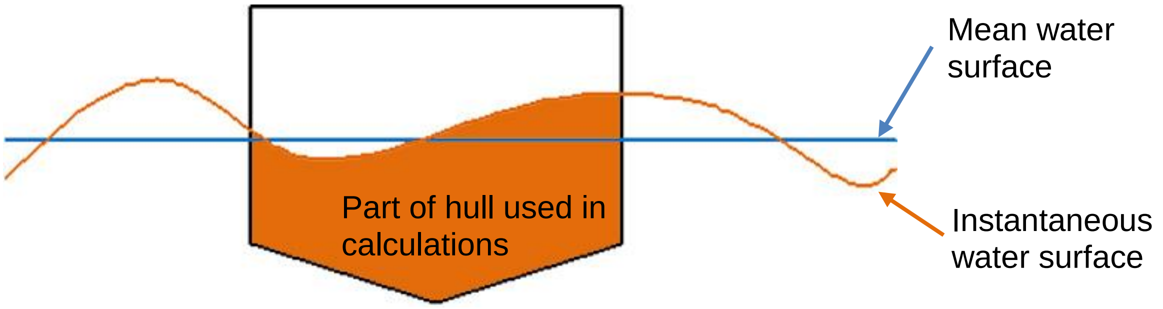


Figure 32: Forces due to the instantaneous water surface: Froude-Kriloff and restoring forces

Figure 32 shows area used in addition in the time-domain calculations. The instantaneous water surface is used to deduce Froude-Kriloff forces (pressure forces due to undisturbed waves) and restoring forces and moments (buoyancy). This is done in the wave federate.

Forces from the air cushion federate are non-linear and based on the instantaneous conditions. The vessel is assumed to be at mass equilibrium when no forces are given (setting in the vessel federate). Therefore static (linear) air cushion forces are calculated. Instantaneous forces are subtracted the static forces, thus only dynamic forces are fed to the vessel federate.

Note that in the calculations done in this report, only the air cushion forces are non-linear. All other forces are based on linear theory (Figure 31). It should also be noted that all forces due to water do not consider the changed wave elevation due to air cushion pressure.

4.3 Walkthrough of a VeSim run

A run is started with initial conditions. These are for instance the heading, position in space, wave condition and vessel velocity.

All external forces are calculated for each time step. These are e.g. instantaneous current, wave condition, motion of the vessel and so on. These forces are then given as inputs to the vessel federate.

The vessel federate calculates what defined as internal forces. These include restoring forces, retardation function (see [HYDRO VESIM 04]), viscous forces and damping in roll (based on input from the maneuvering plug-in). It then solves the equations of motion and gives accelerations, velocities and position of the ship as output.

Now the external force federates will calculate new forces based on output from the vessel federate, give these as input to the vessel federate, and so on.

Example

Calm water. Large single wave approaches. Head sea. What happens?

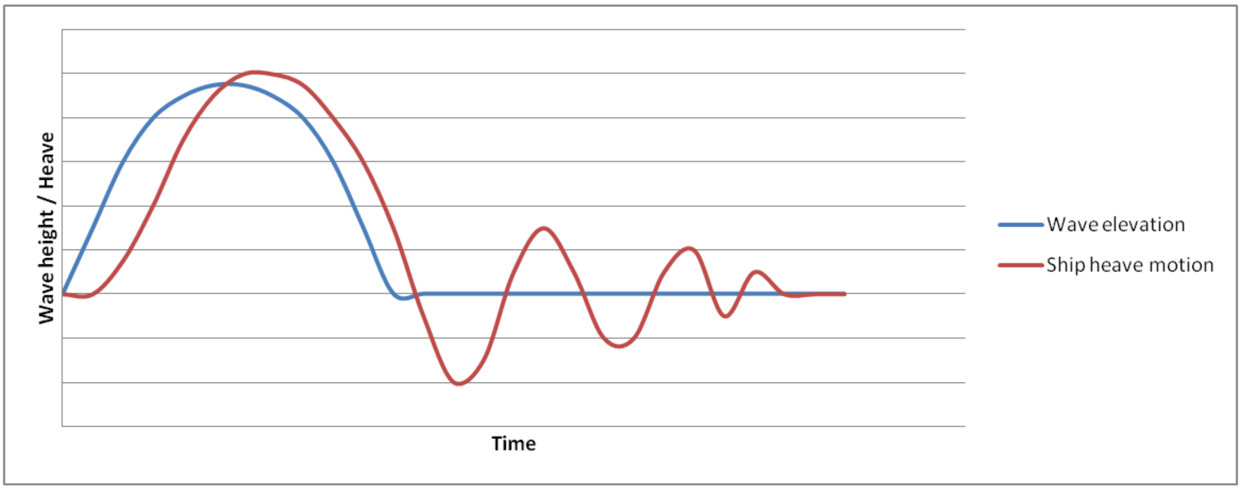


Figure 33: Illustration of single wave exciting heave motion

- As the wave comes towards the vessel, the wave elevation will increase. This is calculated in the wave federate. As the wave "hit" the vessel, large wave forces are calculated in the wave federate
- The SES federate receives information about wave elevation from the wave federate. Large wave elevation gives large air cushion pressure (see the air cushion federate chapter 5.3), which results in a large heave force positive upwards
- The vessel federate receives the large forces from the wave federate and the air cushion federate. It receives wave elevation from the wave federate. Calculating motions, the vessel will get heave (positive upwards) and pitch accelerations that will induce velocities in the next time step
- Wave federate and air cushion federate calculate new forces based on motions and position from the vessel federate
- The vessel federate receives new forces and calculates new motions
- At some point the vessel will move more upwards than the wave elevation (the red line crosses the blue line in Figure 33). Then the vessel federate will start to give negative heave acceleration, and the vessel will start to descend

- After the wave has passed, the vessel will move up and down until it reaches equilibrium condition. The wave federate will now return zero force. When heave is negative, the air cushion federate will return force pointing upwards. When heave is positive, the belonging force will point downwards. The same goes for restoring force calculated in the vessel federate. Thus the motion will damp out, as illustrated in Figure 33

5 The air cushion federate

The air cushion federate is, as the rest of VeSim, made using Java. NetBeans IDE 6.0.1 has been used as editor. The compiler was Sun JDK (Java Development Kit) v1.6, which is integrated in NetBeans. As the author was a complete novice at programming using Java, the main workload of this thesis work has been to learn Java and writing the federate. The code is neither very efficient nor advanced, and has probably large room for improvement. Effort has been made to design the code as logical and easy to understand as possible. Classes and variables have been given meaningful names, and the code is full of comments to ease reading.

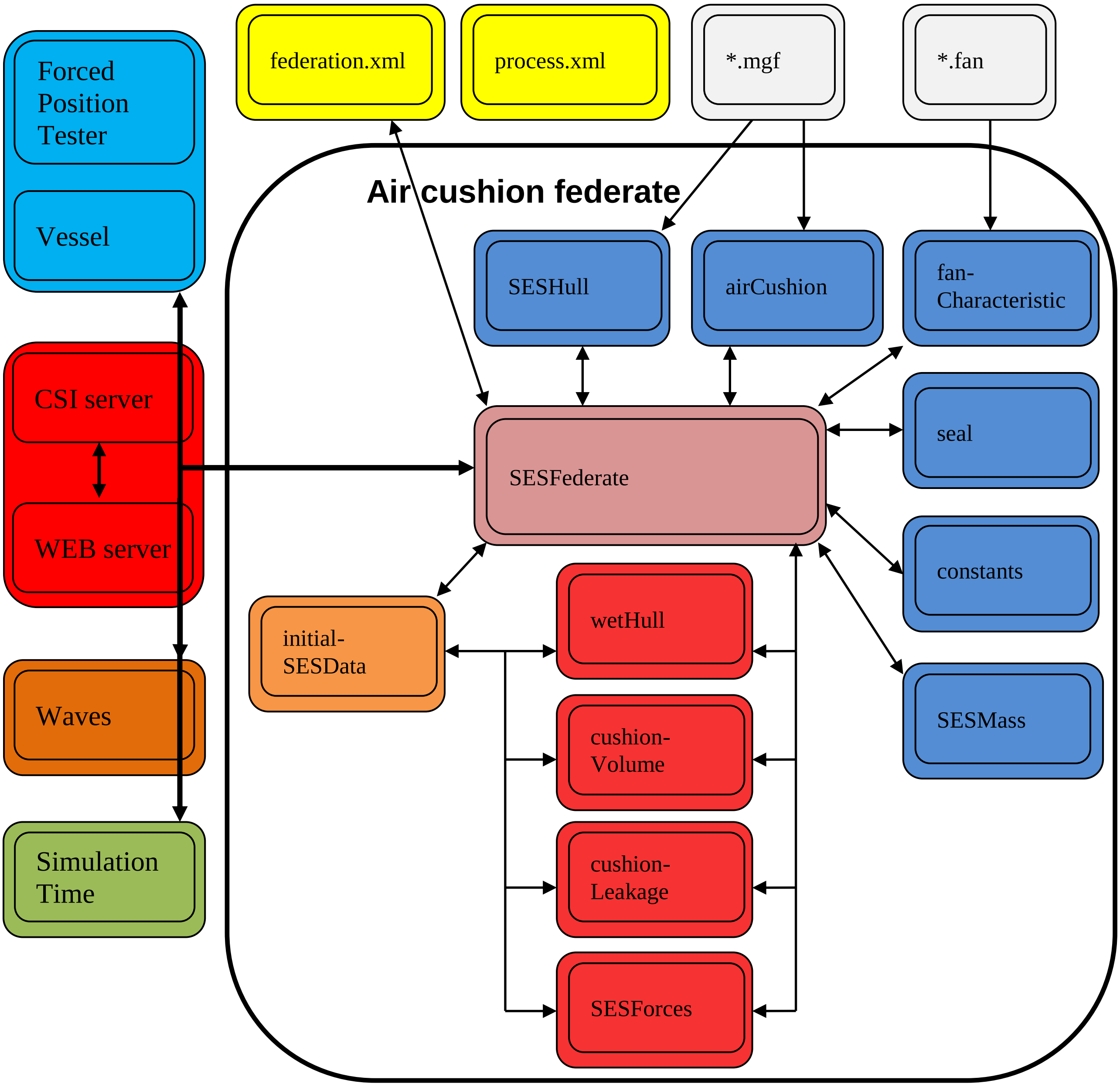


Figure 34: The architecture of the air cushion federate. Inside the main black line are the classes that makes up the air cushion federate. Outside are the federates that it communicate with (sending and receiving attributes), input files and federation setup files (*.xml)

Figure 34 shows the overview over how the different classes in the air cushion federate work together, and also what other federate data is collected from. Java is an object orientated programming. That means that it consists of classes. These are shown inside the main black line in Figure 34. When the program is executed, the classes are realized as objects. Each class can be realized as several objects at the same time. For instance is the seal class realized as two objects: One for the stern seal and one for the bow seal.

The different parts of Figure 34 are shortly described in the following table.

Table 3: Short overview of the different classes, federates and files used by the air cushion federate

Name	***Type***	***Description***
Forced Position Tester	Federate: 6 DOF Vessel model	Alternative to the motion solver. This overrides the vessel federate and locks the position of the vessel to user defined values. Utilized to verify the air cushion federate
Vessel	Federate: 6 DOF Vessel model	Reads in all forces. Main outputs are total forces and 6 DOF position, velocity and acceleration. Also calculates viscous forces and roll damping, retardation functions (added mass and damping) and restoring forces
Waves	Federate: Environmental	Calculates forces due to waves. Gives wave elevation
WEB server	Federate: Simulator kernel	The interface of the simulator. Dynamic web pages with the data information available
Simulation Time	Federate: Time	Gives simulation time
*.fan	Input file	Gives fan characteristic
*.mgf	Input file	Gives geometry of the vessel
airCushion	Java class	Contains geometry of the air cushion
constants	Java class	Contains the different constant/properties used in the various calculations
cushion-Leakage	Java class	Calculates leakages from the air cushion
cushion-Volume	Java class	Calculates enclosed air cushion volume
fan-Characteristic	Java class	Deals with the fan. Reads characteristics from the *.fan file
initialSESData	Java class	Calculates initial or static data, using the java classes marked with red in Figure 34
seal	Java class	Geometry of a seal
SESFederate	Java class	Main air cushion federate file. Controls all other air cushion classes. Calculates air cushion pressure and resistance. Communicates with the CSI server, exchanging data
SESForces	Java class	Calculates forces due to cushion over pressure
SESHull	Java class	Contains geometry of the vessel
SESMass	Java class	Deals with vessel mass
wetHull	Java class	Calculates wet surface
CSI server	Simulator kernel	Connects the different federates by synchronizing their time and exchanging data
federation.xml	Simulator setup	States what parameters and attributes each federate uses. Tells how the federates interact with each other
process.xml	Simulator setup	Governs the startup and termination of the simulator

What described here is the basic function of the federate. How the different physics are interpreted. Underlying equations are shown in chapter 2. Appendix C shows more in detail exactly what the different classes do. The java code is also thoroughly commented in the different java files.

5.1 Assumptions

A few more assumptions are introduced in the java code. They are listed here

- Static air cushion is used to find wave elevation when dealing with volume and leakage calculations. All other calculations use dynamic air cushion pressure. This is believed credible as it is the change in air cushion volume that is important for air cushion pressure. The wave systems made by the hull and air cushion pressure is rather static, meaning that it is the motions of the vessel and the incident waves that are of importance. Steen [STEEN 93] refers to work reporting good agreement between computations and experiments, without taking the diffraction effect into account
- The geometry in the *.mgf file is dense enough to give accurate calculations
- The seals are modeled as static rigid walls. This will give somewhat too large leakage areas (seals are dynamic and will follow the water surface in small sea states)
- The air cushion pressure is uniform
- Contraction of air flow, which actually is a function of leakage area and shape, is assumed constant for all leakages
- Fan curves have to be 1-to-1 with pressure and air flow, so that air flow is strictly decreasing with increasing air cushion pressure. A real fan curve has a 2-to-1 relation between air flow and pressure when pressure is higher than a certain value (see Figure 22)
- The vessel is initially at mass equilibrium. That is, with the air cushion federate turned off the vessel floats as it was on cushion
- Point of attack of added viscous resistance: (x,y,z) = (Lpp/2, 0, Static draught/2) relative AP, CL, BL

5.2 Input and output

All inputs to the air cushion federate are given or controlled by the federation.xml file (see appendix B.2). It gives variable values, e.g. static air cushion pressure, and the path to other input files. The different inputs can be divided into three different types:

A. Data that is given in the federation.xml file. Two types: Attributes which are constant data, parameters which are data that can be changed during the simulation run. In addition also the time step (how often the air cushion federate makes its calculations) is given in the federation.xml file. Standard value for this is the same as the vessel federate, 0.05 [s]
B. Data that is obtained from the other VeSim federates. The data which the different federates has access to from other federates is defined in the federation.xml file
C. Data given by an input file (geometry of vessel, fan characteristic, vessel displacement mass). The paths to these files are given in the federation.xml file

Attributes are mostly used to give both input and output from the federate. The following table show the attribute input data to the federate. These must be given manually in the federation.xml file. In the long run, these data should be incorporated as data given in ShipX when setting up the simulator for a SES run.

Table 4: Input attributes in the air cushion federate. Must be given manually in the federation.xml before startup of the simulator run

Name	*Unit*	*Description*
c_n	-	Contraction of air flow out from the air cushion
H	μm	Hull roughness factor
k	-	Hull form factor
p_a	Pa	Atmospheric pressure
seal_bow_x_rel_ap	m	Longitudinal position of bow seal relative aft perpendicular
seal_bow_z_rel_bl	m	Vertical position of bow seal relative baseline
seal_stern_x_rel_ap	m	Longitudinal position of stern seal relative aft perpendicular
seal_stern_z_rel_bl	m	Vertical position of stern seal relative aft baseline
rho_a	kg/m^3	Mass density of air at atmospheric pressure
v	m^2/s	Kinematic viscosity

Parameters are federate settings and are input to the federate. These can be changed during a simulator run. The following table shows the different parameters used by the federate.

Table 5: Parameters in the air cushion federate. Initial values must be given manually in the federation.xml. Parameters can be changed during a simulator run

Name	*Unit*	*Description*
cushion_active	-	Air cushion active. True or false
on_cushion	-	True: Calculate air cushion forces from the air cushion. False: Off cushion
fan_and_leak_active	-	Calculate air flow in/out from air cushion. True or false
fan_number	-	Number of active fans
forces_p_c	-	Calculate air cushion forces using p_c (false: use p_0). True or false
p_0	Pa	Static over pressure
update_air_cushion	-	Reset/update air cushion based on current settings. True or false
viscous_forces	-	Include added viscous resistance. True or false

In data of type B are mainly two things:

- Incident wave elevation (and wave particle velocity for the resistance calculation) from the wave federate
- Motion of the vessel, principally where the vessel is placed in the global axis system (see Figure 39). This is obtained from the vessel federate

There are also a few other data, like the simulation time (used to obtain correct wave data) and current velocity (for resistance calculation). All type C data is obtained from the vessel, the wave and the simulation time federates.

Three more files are used as input to the federate. The paths to these must be specified in the federation.xml file. The first path is the path to the temp folder where the simulator unpacks the vessel model archive. This archive contains input files from ShipX that is used by the simulator. In the future also all files needed for the air cushion federate should be included here. The second path is to the *.mgf file, containing the geometry of the vessel. The last one is for the *.fan file, containing fan characteristic.

In the vessel model archive the retfun.re10 file is accessed. Among other things, this file contains the mass matrix of the vessel. The (1,1) position in the matrix gives vessel mass. The standard option in VERES is that vessel mass equals displacement. Until only one week before completion of this report, the author did not know that there was an override setting, and had programmed the air cushion federate to deal with the missing weight of the vessel. This gave bad results for natural periods in VeSim. Now the vessel mass found in the retfun.re10 is used simply to give the air cushion lift vs. vessel weight ratio.

To obtain compatibility with ShipX, the hull geometry the federate uses is the same as the one used by VERES (filename *.mgf, see appendix B.3). The path to this is the second path that must be specified in the federation.xml file.

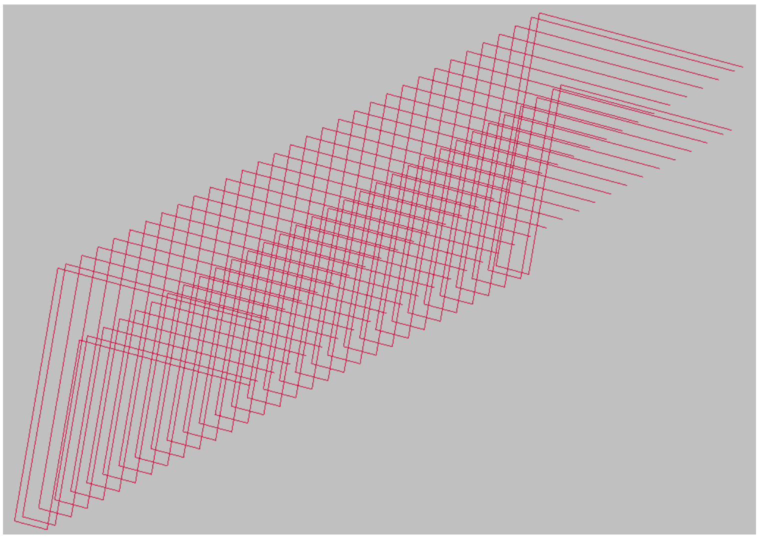


Figure 35: Geometry defined by section curves. Here one half section of case 1, figure from AutoCAD

Using the *.mgf system, the hull is defined by cross sections. Each section has an x-coordinate and then pairs of y- and z-coordinates giving nodes defining the shape of the section. The model of case 1 is shown in Figure 35. The lpp is also given in the mgf file. All calculations dealing with geometry in the air cushion federate uses the section data. No interpolation is done between the sections, so that each section has an extension in x-direction, dx, which is used in the calculations. The distances between the sections decide thus the accuracy of the calculations. Along the sections the geometry is split into lengths of a default maximum length of 1 [m]. This length can easily be altered in the java code.

The last file is the *.fan file. The layout of the file is shown in appendix B.4. It is basically pairs of (Q,p) making up the pressure versus air flow curve. The file cannot be altered during the simulation run, but it is possible to alter how many fans that are used (they are all identical) using the "fan_number" parameter.

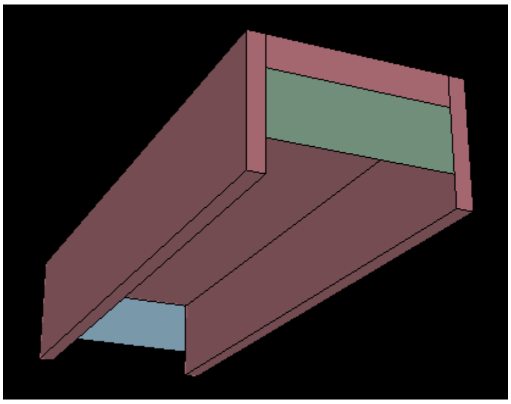


Figure 36: Simplified square SES (case 1), fitted with stern and bow seals. Figure from GLview

The seals are modeled by a thin vertical and rigid wall, described by a pair of x- and z-coordinates. The x-coordinate is the longitudinal position, and the z-coordinate is the bottom of the seal. Leakages between the demi hulls and the seal are included in the static leakage area. Sideways geometry of the seals is decided by the closest section from the hull geometry file.

Output from the program is mainly forces. Out data are given as attributes. The following table gives an overview.

Table 6: Output attributes from the air cushion federate. These are given for each time step (default is every 0.05 [s]. The attributes are shown in the web interface and can also be logged, given as output in a log file. The * means respectively surge, sway and heave, roll, pitch and yaw

Name	*Unit*	*Description*
Q_0	m^3/s	Static air flow from fans
Q_in	m^3/s	Air flow from fans
Q_out	m^3/s	Leakages from air cushion
cushion_force_*	N	Force due to air cushion over pressure
cushion_moment_*	Nm	Moment due to air cushion over pressure
p_c	Pa	Air cushion overpressure
cushion_volume	m^3	Enclosed air cushion volume
cushion_volume_0	m^3	Static enclosed air cushion volume
force_*	Pa	Total force from the air cushion federate
lift_ratio	-	Air cushion lift/total mass. Ratio
lift_ratio_0	-	Static air cushion lift/total mass. Ratio
moment_*	Nm	Total moment from the air cushion federate
viscous_added_force_surge	N	Added viscous surge force
viscous_added_moment_pitch	Nm	Added viscous pitch moment
viscous_calm_force_surge	N	Viscous surge force for calm water
viscous_calm_moment_pitch	Nm	Viscous pitch moment for calm water
viscous_force_surge	N	Total viscous surge force
viscous_moment_pitch	Nm	Total viscous pitch moment
wet_surface	m^2	Wet surface
wet_surface_0	m^2	Static wet surface
x_rel_ap	m	Point of attack (of force) relative AP x-dir
y_rel_cl	m	Point of attack (of force) relative CL y-dir
z_rel_bl	m	Point of attack (of force) relative BL z-dir

5.3 The federate's manner of operation

This chapter gives a brief overview over how the federate functions. Figure 34 and Table 3 to Table 6 should be frequently reviewed while reading it to easier grasp the manner of operation. Figure 38 to Figure 42 showing the different axis system could also be worth a glance.

Very summarized, the federate calculates all values needed for the air cushion pressure equation (see chapter 2.6). Then it solves the equation. It calculates forces by integrating the pressure over the dry surface of the air cushion. In addition, it calculates added viscous resistance based on wet surface and vessel speed. These forces are sent to the vessel federate which calculates the vessel's motions.

At startup, the air cushion federate loads its parameters (see Table 5). If the “air_cushion_active” is set to “true”, it will read and set up all geometry objects (see Table 3: SESHull, airCushion, seal), fan characteristic (see Table 3: fanCharacteristic) and the calculations constants (see Table 3: constants). Next step is to calculate static and initial conditions. This is done by making an “initalSESData” object (see Table 3. The object contains all static data). In addition, initial values to the integration variables are calculated based on the static data. Integration means here the integration that is done to calculate air cushion pressure. The startup sequence that just was described is repeated every time “update_air_cushion” is set to true. Altering the value of the other parameters will make no difference in the calculations until update_air_cushion” is set to “true”.

Next step is to calculate values needed in the air cushion pressure equation. The geometry objects are updated with global coordinates (coordinates in the global NED system, see Figure 39), and enclosed air cushion volume, wet surface and leakages are calculated by the cushionVolume, wetHull and cushionLeakage classes. An important parameter in finding wet surface, volume and leakages are the crossing point between the water surface and the hull and seal.

The enclosed air cushion volume is defined as the volume enclosed by the wet deck, the demi hulls, the two seals and the water surface. The water surface inside the air cushion is lowered a constant height due to the air cushion pressure. The principal is shown in Figure 5 and Figure 37.

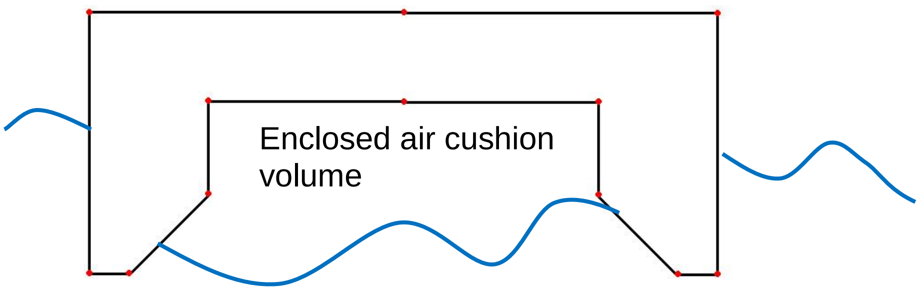


Figure 37: Definition of enclosed air cushion volume

The leakage is separated into three contributions:

A. Leakages under hull
B. Leakages under seals
C. Leakages through louvers, between seals and hull, and the like

Figure 37 shows no leakages under the hull. At normal operation it would never do either. Most leakages happen under the seals. The third type of leakages, type C, depends on the static/initial condition. It is a requirement that at static condition the air flow into the cushion equal that out of the cushion. Based on the position of the seals, the static leakages are divided into type B and C (calculate leakage type B, remaining leakage to fulfill static leakage is type C). Leakage of type C is constant based on the static conditions. Leakages under seals and hull are calculated for each time step.

Two methods are available for how to calculate the air cushion pressure. Both calculate uniform air cushion pressure. One takes air flow in and out of the air cushion into account and the other does not. The air flow from the fans is found using the fan characteristic object.

Air cushion pressure is given as input and the air flow is returned. The air cushion pressure is found using the equations in chapter 2.6.

Up till now, air cushion pressure from last time step (or static condition if first time step) has been used in the calculations. For leakages and air cushion volume, the static air cushion pressure is used (the model comes unstable and unphysical if using instantaneous air cushion pressure). For the force calculations, the air cushion pressure for the current time step is used.

The main air cushion forces, those that is due to the air cushion over pressure, is calculated by the SESForces class. The forces are calculated by integrating the air cushion over pressure over the dry surface of the air cushion. The parameter "on_cushion" decides whether these forces should be calculated or not. "forces_p_c" decides if the calculated air cushion over pressure, p_c, or the static over pressure, p_0, should be used. The vessel is assumed to be initially at mass equilibrium at the given draught (given in ShipX). So forces given by the air cushion federate are only the dynamic forces. Mathematically:

$$F_{dyn,i}(t) = F_i(t) - F_{stat,i} \tag{5.1}$$

Static forces are calculated for all 6DOF in the initial calculations. Setting the parameter "on_cushion" false is equal to set the vessel off cushion (see movies on attached CD-ROM). Then $F_i(t)$ is equal to zero and $F_{dyn,i}(t)$ is equal to $-F_{stat,i}$.

Final calculation is the added viscous resistance. The calculation is quite straight forward, simply using the equations given in chapter 3. The vessel speed used is the relative velocity between the vessel and the water particles (taking current and waves into account).

When all calculations have been carried out, final values are sent to the CSI server and out to the web interface as attributes. The values can be shown in real-time plots or logged to text files.

What described in this chapter will be repeated for each time step.

5.3.1 Example of an air cushion federate run

The federate functions as described in last chapter. This chapter gives a more practical view, following up on the example run given in chapter 4.3. Only this time seen from the air cushion federate's point of view. Figure 33 could be reviewed. One long and large wave approaches the SES. What happens?

- Before the wave arrives. Still water. The vessel is laying still. The wave is moving straight towards the SES, head sea
- The wave lifts the water in the fore part of the air cushion:
    - The enclosed air cushion volume decreases as the water is lifted
    - The leakage from the bow decreases (or remains unchanged), as the wave closes the gap between the seal and the water surface. This depends on the vertical position of the seal
    - Less volume and less leakage increase the air cushion pressure
    - Increased pressure gives increased heave force
- The vessel federate receives a larger heave force from the air cushion federate. The vessel receives a heave motion upwards

- As more and more water is lifted by the wave, the cushion volume will keep increasing (but some counteracted by the vessel also moving upwards). Increased air cushion pressure will also give less air input from the fan system. This will counteract some of the increased air cushion pressure, i.e. the fan system damps the vessel motions
- The vessel federate receives new forces and calculates new motions
- At some point the vessel will move more upwards than the wave elevation. Air cushion volume will increase. Leakages will increase. The air cushion pressure will decrease and start to pull the vessel down. Note that the increase in leakage area counteract the vessel motion (motion upwards), thus damps the motion. This contribution can become very large. Not much leakage area is needed to decrease the pressure significantly. As air cushion pressure decreases, the fan system will provide more air into the cushion. Thus the fan system once again damps the vessel motion.
- The wave will here at some time pass by the SES
- The vessel federate receives new forces and calculates new motions. The vessel will start to descend
- As the vessel descends bellow static position, air cushion volume will decrease below static value giving larger pressure forces and therefore push the vessel upwards. The opposite happens when the vessel moves above static position. The vessel will be pushed downwards
- The fan system will damp the motions, giving less input when the air cushion pressure is above the static pressure, and more when it is lower
- The leakage will damp the motions, reducing air cushion pressure when the vessel moves above static condition
- The motion will damp out as shown in Figure 33

Note that with no leakages and fans, the motions would only be damped by hydrodynamic damping. The magnitude of these forces is too low, so the vessel will get increased oscillations instead of reduced. More about this is found in chapter 6.2.2.

5.4 Coordinate systems

In VeSim and ShipX a total of five different axis systems are used. This has been major a troublemaker throughout the thesis work. To clarify when and how which system is used, here is an overview.

Figure 38 shows the axis system for the *.mgf geometry files. These files are automatically generated in ShipX and are used both by VERES and the air cushion federate in VeSim. The geometry is defined by cross sections. The sections are sets of y and z coordinates, and are numbered from stern (first number) to bow (last number). See Figure 35 to see an example of a section curve.

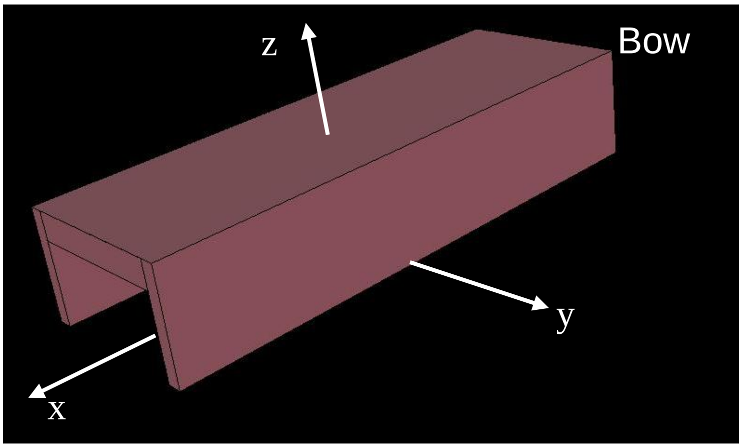


Figure 38: Axis system used in *.mgf geometry files. Origin at Lpp/2, CL and BL. Right hand system

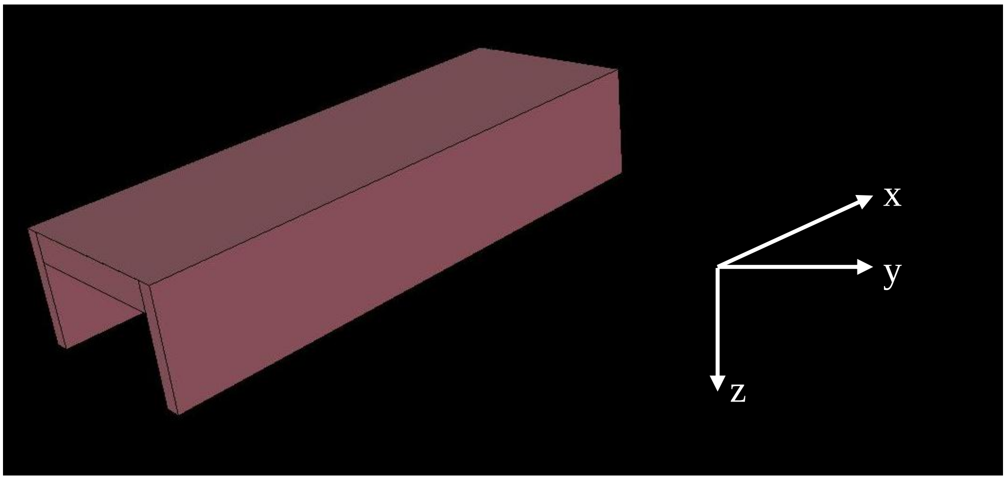


Figure 39: Global NED system. Right hand system

The NED system (Figure 39) is given relative to the earth ellipsoid. The xy-plane is tangent to the surface of the earth (equal the mean water surface). The z axis is pointing downwards towards the center of the earth, and its origin is at the loaded draught of the vessel. The position and orientation of the vessel is given in the simulator in the global NED system.

All positions of vessel-related objects are given in the body system showed in Figure 40. Examples are positions of seals and position of center of gravity. The air cushion uses its own axis system, shown in Figure 41. The geometry of the cushion is stored in this. The main reason to introduce this system was to simplify calculations with spatially varying air cushion pressure. Everything related to the air cushion uses this system for calculations. For example are forces due to air cushion pressure calculated using this system (origin of these forces equals origin of the air cushion axis system).

The motions of the vessel are given relative the system shown in Figure 42. It also tells the positive direction of forces given as input to the vessel federate.

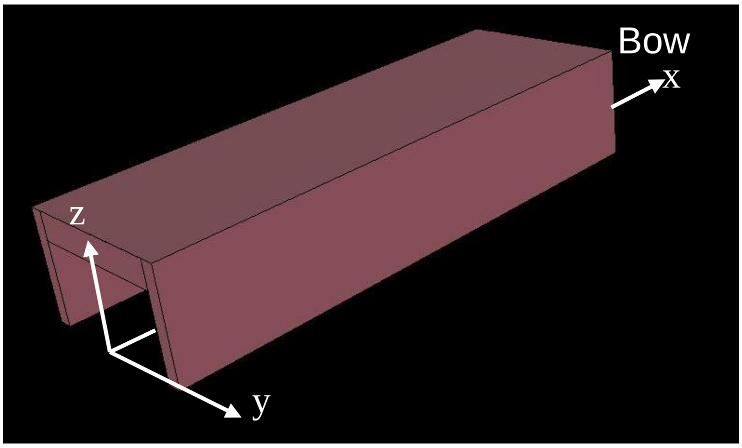


Figure 40: Body position system. This is locked to the vessel and is used to give positions on (or relative to) the vessel. Origin at AP, CL and BL. Left hand system

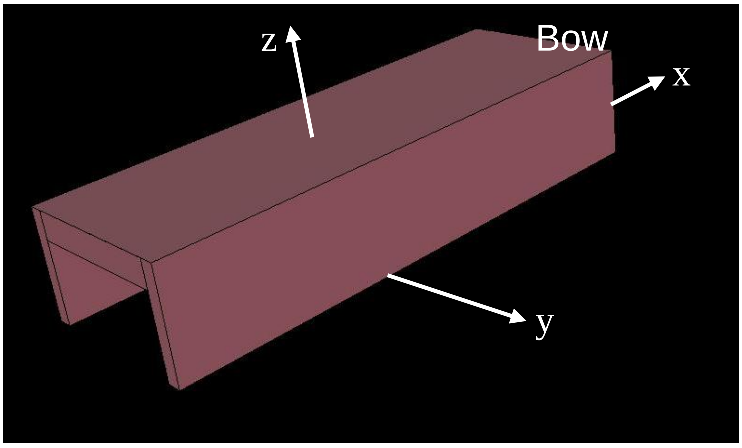


Figure 41: Air cushion axis system. It is locked to the vessel. Origin at longitudinal center of the air cushion, CL and BL. Left hand system.

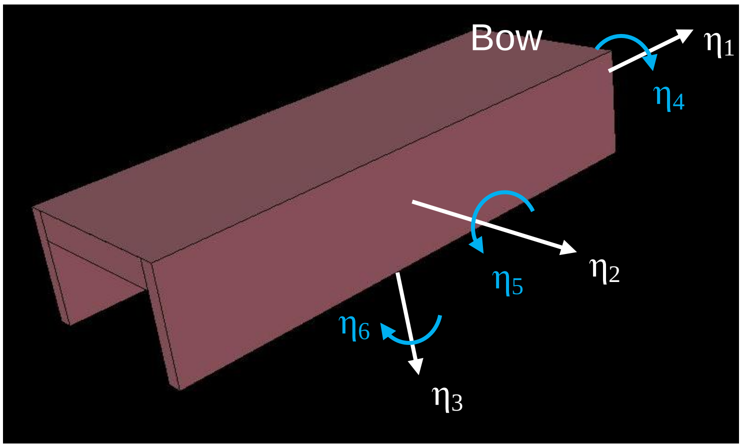


Figure 42: Body fixed reference frame (b-frame). The axis system is coinciding with the principal axis of iniertia. Right hand system

A few examples of transition between the different axis systems

- The air cushion federate is started up. It reads the geometry of the vessel from a *.mgf file, using the *.mgf axis system. Longitudinal positions of seals are given using the body position system. The seal positions decide aft and fore extensions of the air cushion, and therefore also decide the origin of the air cushion axis system. The geometry of the air cushion (read from the *.mgf axis system) is stored in the air cushion axis system

- For each time step the wet surface is calculated. The global position of the hull is therefore needed in order to compare this with the wave elevation. To find the global position a motion converted in the Marintek Java package is used. The input here must be in the body position system, output is in the global NED system. I.e. the data is collected from the air cushion axis system, translated to the body position system and returned in the global NED system
- Air cushion pressure forces are calculated in the air cushion axis system. The position of center of attack is the origin of this system and must be given as input to the vessel federate along with the forces. The convention of positive direction on the forces must be translated from the air cushion axis system (Figure 38) to the b-frame system (Figure 42). This means that heave forces and yaw moments must change sign.

As can be seen, it quickly becomes quite complicated.

6 Calculations and results

6.1 Presentation of cases

Four different vessel designs have been used in VeSim and the air cushion federates. Three of them were very simple designs (case 1, 1.1 and 2). All of them were used in developing the software (the verification part). As the mass balance finally came out correct (the setup for VeSim has default total mass value as displacement), it turned out that case 1 was very unstable (see video on attached CD-ROM) and thus useless to obtain physical ok results with using the simulator. Therefore case 1.1 was made.

Case 1.1 has been used extensively to test seakeeping and resistance results (comparing with linear calculations from VERES). The last design (case 3) is a real vessel. This was also used to test seakeeping and resistance, presenting a more real-life vessel design. Data for the real vessel was obtained from lieutenant-commander Christian Wines [WINES 08]).

Only one loading conditions pr vessel are used in all calculations, giving one static overpressure and one vessel mass. All input data for case 1 and 2 are on the attached CD-ROM (including geometry files, ShipX-exchange files, etc). Data for case 3 is not published, as this would render this report confidential.

6.1.1 Case 1: Square SES

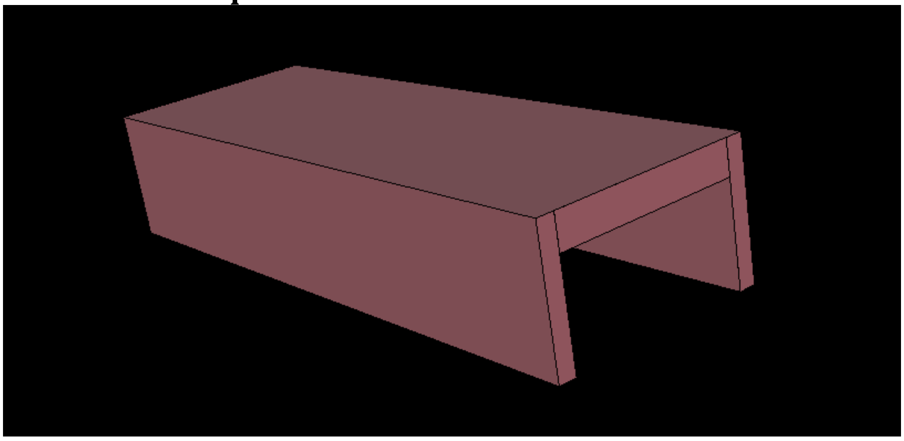


Figure 43: Vessel model used in case 1. Figure from GLview

The square SES was used to verify the calculations done in the air cushion federate. The simple shape made results easy to compare with hand calculations. The following tables show main model inputs to ShipX and VeSim.

Table 7: Case 1. Main dimensions and data

Main details	Symbol	Value	Unit
Length over all	L_{oa}	30	m
Length between perpendiculars	L_{pp}	30	m
Height to wet deck relative base line	h_c	4.5	m
Moulded depth	D_r	6	m
Breadt	B	10	m
Breadt demi hull	B_d	0.8	m
Draught	T	1.50	m
Wetted surface (zero speed)	S_0	180	m^2
Static air cushion volume	Ω_0	956	m^3
Cushion length	L_c	30	m
Mean cushion breadth	B_c	8.4	m
Static overpressure in air cushion	p_0	8000	Pa
Ride control parameter	k	0	[-]
Displacement	-	74	ton
Air cushion support	-	74	%
Air cushion lifted	-	206	ton
Total mass	M	279	ton
Vertical center of gravity, relative base line	V_{CG}	3	m
Longitudinal center of gravity, relative stern	L_{CG}	15	m
Distance between sections in mgf file	-	1	m

Table 8: Case 1. Fan characteristics

Design condition	Value	Unit
Design air flow	150	m3/s
Design air pressure	8000	Pa
	795.6	mmWc
dQ/dp, one fan	-0.028	m3/sPa
	-0.284	m3/smmWC

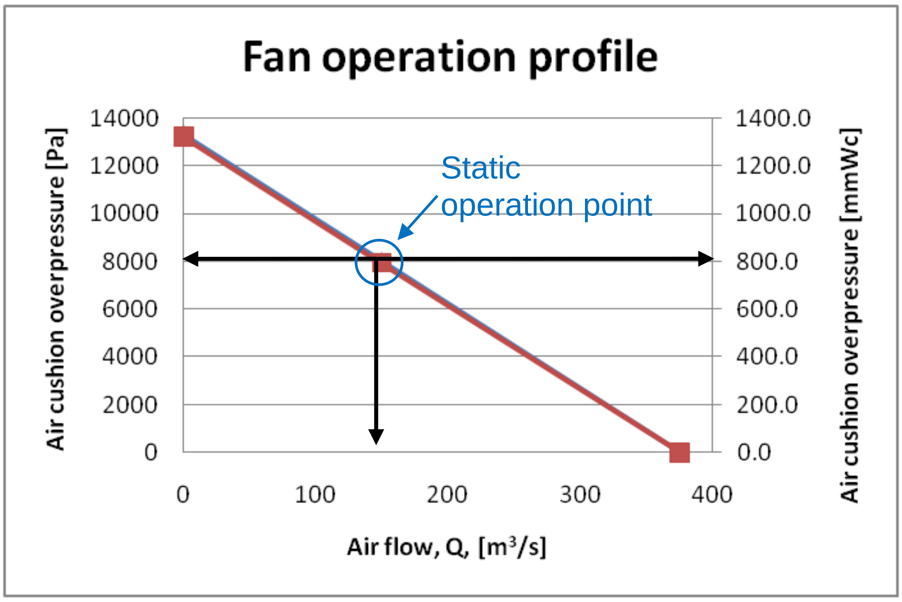


Figure 44: Case 1. Fan operation profile. Linear profile

Table 9: Case 1. Seal characteristics

Bow	Value	Unit
Type	Finger	-
Seal angle (from horizontal)	45	degrees
Horizontal position relative AP	30	m
Position of bottom relative BL	0.1	m

Stern	Value	Unit
Type	Lobe bag	-
Seal angle (from horizontal)	45	degrees
Lobe bag overpressure percentage	15 %	-
Overpressure	600	Pa
	59.7	mmWc
Horizontal position relative AP	0	m
Position of bottom relative BL	0.15	m

Table 10: Case 1. Radii of gyration

Radii of gyration	r44	r55	r66	Unit
Source: [ITTC 1999]	3.5	7.5	7.5	m

VERES can do a “data check”. This is quite useful to do before a calculation, as a control of given input data and initial data that can easily be controlled. The data checks for case 1 are given in Appendix A.4. Note that VERES do not calculate the air cushion volume correctly, while seemingly not taking the water elevation due to air cushion pressure into account.

6.1.2 Case 1.1: Simple SES

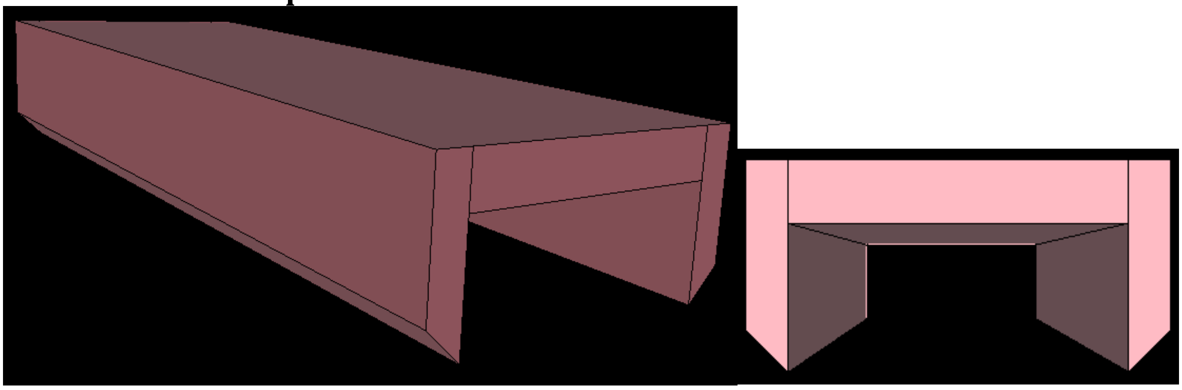


Figure 45: Vessel model used in case 1.1. Figure from GLview

The square SES was used for seakeeping and resistance calculations in VERES and VeSim. The simple straight sided shape made it easy to compare linear with non-linear calculations. The following tables show main model inputs to ShipX and VeSim.

Table 11: Case 1.1. Main dimensions and data

Main details	Symbol	Value	Unit
Length over all	L_{oa}	30	m
Length between perpendiculars	L_{pp}	30	m
Height to wet deck relative base line	h_c	3.5	m
Moulded depth	D_r	5	m
Breadt	B	10	m
Breadt demi hull	B_d	1	m
Draught	T	1.20	m
Wetted surface (zero speed)	S_0	144.98	m^2
Static air cushion volume	Ω_0	647	m^3
Cushion length	L_c	30	m
Mean cushion breadth	B_c	8	m
Static overpressure in air cushion	p_0	4000	Pa
Ride control parameter	k	0	[-]
Displacement	-	42	ton
Air cushion support	-	70	%
Air cushion lifted	-	98	ton
Total mass	M	140	ton
Vertical center of gravity, relative base line	V_{CG}	3	m
Longitudinal center of gravity, relative stern	L_{CG}	15	m
Distance between sections in *.mgf file	-	1	m

Table 12: Case 1.1. Fan characteristics

Design condition	Value	Unit
Design air flow	75	m3/s
Design air pressure	4000	Pa
	397.8	mmWc
dQ/dp, one fan	-0.018	m3/sPa
	-0.186	m3/smmWC

In the calculations, the vessel uses two fans, giving design air flow of 150 [m^3/s].

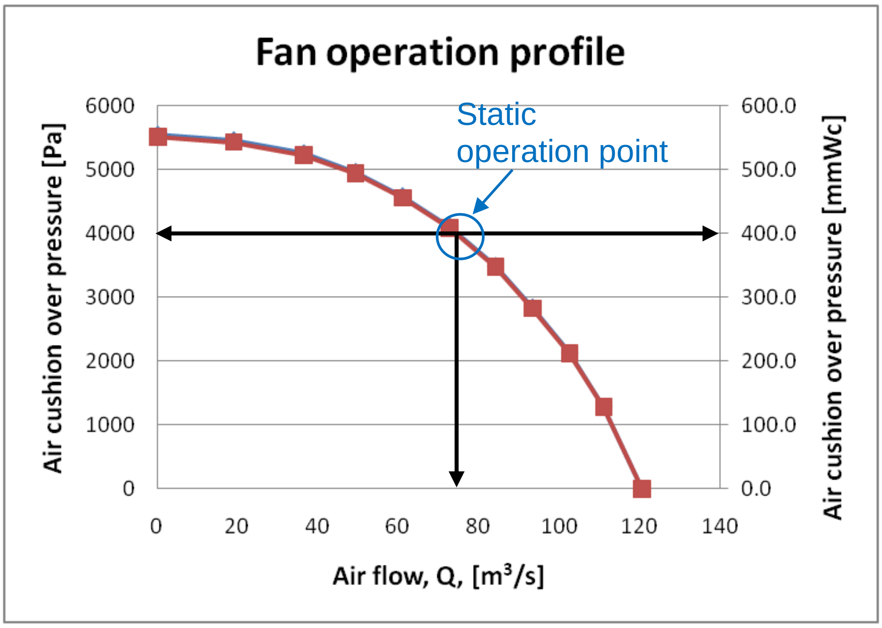


Figure 46: Case 1.1. Fan operation profile

Table 13: Case 1.1. Seal characteristics

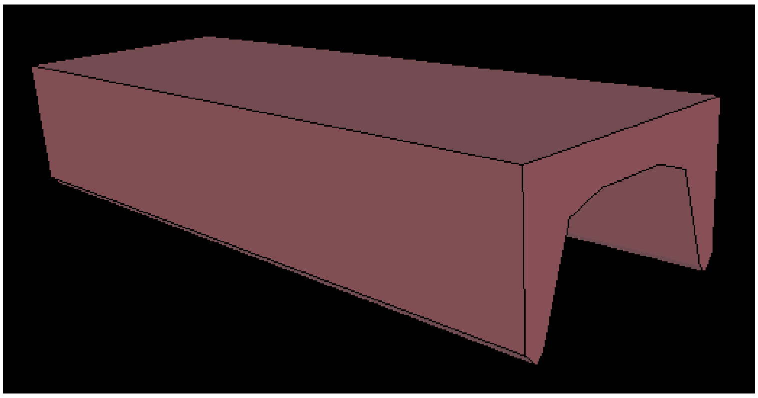
Bow	Value	Unit
Type	Finger	-
Seal angle (from horizontal)	45	degrees
Horizontal position relative AP	30	m
Position of bottom relative BL	0.1	m

Stern	Value	Unit
Type	Lobe bag	-
Seal angle (from horizontal)	45	degrees
Lobe bag overpressure percentage	15 %	-
Overpressure	600	Pa
	59.7	mmWc
Horizontal position relative AP	0	m
Position of bottom relative BL	0.15	m

Radii of gyration are the same as for case 1 (same main dimensions).

The VERES data checks for case 1.1 are given in Appendix A.5. Note the huge difference between longitudinal and transverse metacentric height compared to case 1. This vessel is a lot more stable.

6.1.3 Case 2. Inclined SES



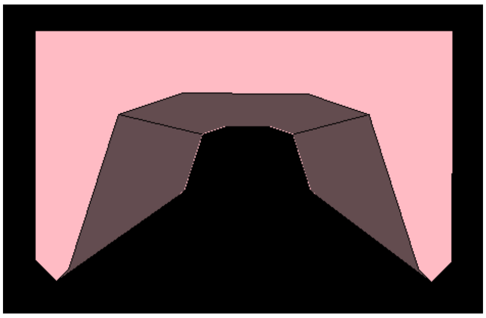


Figure 47: Vessel model used in case 2. Figure from GLview

The inclined SES was only used to verify the calculations done in the SES federate, presenting a little more challenging geometry than the square SES in case 1. This has been especially been useful to verify volume calculations.

Table 14: Case 2. Main dimensions and other data

Main details	Symbol	Value	Unit
Length over all	L_{oa}	30	m
Length between perpendiculars	L_{pp}	30	m
Height to wet deck relative base line	h_c	4.5	m
Moulded depth	D_r	6	m
Breadth	B	10	m
Breadth demi hull at water line	B_d	1.19	m
Draught	T	1.50	m
Static wetted hull surface	S_0	153	m^2
Static air cushion volume	Ω_0	766	m^3
Cushion length	L_c	30	m
Cushion breadth	B_c	8.14	m
Overpressure in air cushion	p_0	8000	Pa
Ride control parameter	k	0	[-]
Displacement	-	78	ton
Air cushion support	-	72	%
Air cushion lifted	-	199	ton
Total mass	M	277	ton
Vertical center of gravity, relative base line	V_{CG}	3	m
Longitudinal center of gravity, relative stern	L_{CG}	15	m
Distance between sections in *.mgf file	-	1	m

Fan and seal characteristics and radii of gyration are the same as for case 1.

The VERES data checks for case 2 are given in appendix A.6.

6.1.4 Case 3. Alta class Mine Counter Measure Vessel (MCMV)

Figure 48: The Alta class MCMV. Picture from UMOE Mandal web page [UMOE 08]

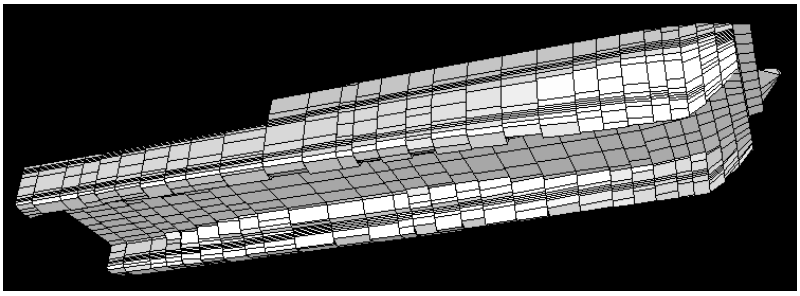


Figure 49: The Alta class MCMV. Panel generator run from ShipX. Figure from GLview

Most information about the MCMV is confidential. For this report not to become confidential as well, most data is therefore omitted. Here only main dimensions that also are available elsewhere are listed. In the results only relative and non-numbered data is shown.

Table 15: Case 3. Main dimensions and data

Main details	Symbol	Value	Unit
Length over all	L_{oa}	55.2	m
Length between perpendiculars	L_{pp}	52	m
Height to wet deck relative base line	h_c	2.9	m
Mean draught on cushion	T	1.40	m
Breadth	B	13.55	m
Weight fully loaded	M	395	ton

The fan characteristics are modeled from the actual fan characteristics on the MCMV. The seals are modeled as rigid walls. The seals reach down to about BL. This makes it necessary to have some motion/waves before leakage occur.

6.2 Verification of software

A significant amount of work has been made on verification of the programs. In fact, this has been one of the major efforts in the thesis work.

The two main tools used in the verification is printout to the command window and the forced position tester. The forced position tester allows the vessel to stay fixed in space in a user-specified position. This (often in combination with zero wave elevation) allows for simple tests to whether the federate calculates correct values.

The different tests that have been carried out are a vast number, therefore only a little selection is displayed here.

Examples of verification

- Throughout the programming work, data has been extracted to the command window and compared to manual calculations. For instance position of seals, coordinate conversions, height between hull and water, etc
- Datasets has been plotted and verified according to what expected. For instance the instantaneous calculated air cushion volume should oscillate around the constant air cushion volume
- Many of the calculations have been tested on a single section first. Dealing with only one section, and using the forced position tester, makes it a lot easier to control the federate with hand calculation
- Increase number of fans and see that air cushion pressure gets a lower oscillation amplitude and/or increases
- Lift the vessel high above water
    - See that the air cushion volume and leakages increases correspondingly to the height
    - Give it a large pitch in addition, see that the air cushion volume decreases (the air cushion volume is interpreted as the vertical distance times extension of geometry in global xy-plane) and that the leakages are about the same
    - Give it large roll in addition and see that the air cushion volume increases while the leakage stay about the same
- Move vessel under water and see that volume and leakages becomes zero

6.2.1 Volume calculation

The volume calculation is one of the most important calculations done. Only a small change of enclosed air cushion volume gives a large change of pressure. Therefore, lots of time has been spent making certain that the volume is calculated as correctly as possible.

Most of the time has been spent studying printouts from the calculations in the command window, but also more easy and more visual tests have been carried out. The wave period T for a wave with length L is given by (assuming deep water):

$$T = \sqrt{\frac{2\pi}{g} L} \tag{6.1}$$

By putting L equal L_c/n where n is an integer, head sea, the corresponding wave period should give static cushion volume. The same goes for beam sea with breadth of cushion instead of length. Using the equation for n equal 1, 3 and 5 for case 1, the following wave periods are obtained:

Table 16: Canceling wave periods for air cushion volume. case 1

	n = 1	n = 3	n = 5
L_c = 30	4.38	2.53	1.96
B_c = 8.4	2.32	1.34	1.04

Figure 50 shows the results from n equal 1 and wave in longitudinal direction. Table 17 shows all results. Wave height in all calculations was 3 [m].

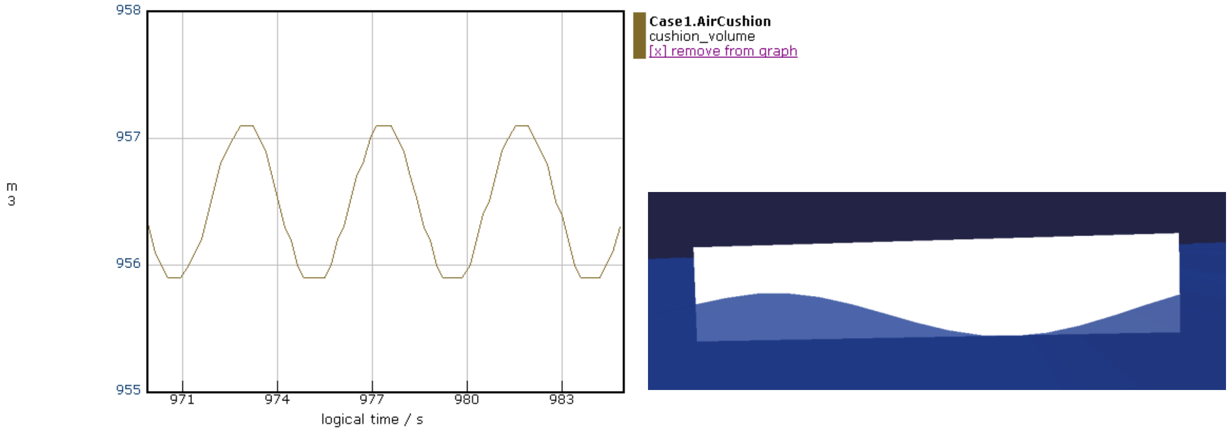


Figure 50. Example of verification of air cushion federate. Wave with length equal to cushion length passing through vessel that is standing still

Figure 50 shows very good results for a wave equal to cushion length. The static air cushion volume is about 956.5 [m^3], meaning that the deviation of the oscillation amplitude is only 0.07 % off. Similar calculations are done for the rest of the wave periods shown in Table 16. The results are shown in Table 17.

Table 17: Calculated enclosed air cushion volume with wave lengths equal to what should give constant air cushion volume. Case 1

	Head sea. Wave period [s]			Beam sea. Wave period [s]		
	4.38	2.53	1.96	2.32	1.34	1.04
Ω_0 [m^3]	956.5	956.5	956.5	956.5	956.5	956.5
Max deviation [-]	0.07 %	0.05 %	0.09 %	0.02 %	0.03 %	0.00 %

Ω_0 in Table 17 is static air cushion volume found as mean calculated air cushion volume. As seen, all equal the correct value of 956.5 [m^3]. The calculated volume oscillates about this value, and the max deviation is the amplitude of the oscillation. As seen, the deviation is very small. The same kind of analysis has also been done with different vessel position (different roll, pitch, yaw, vertical position). Also here, the deviation was very small. It is therefore concluded that the calculated air cushion volume is accurate.

Video from head sea with T equal 2.53 and beam sea with T equal 2.32 has been put on the attached CD-ROM.

6.2.2 No air flow entering or leaving air cushion

The SES federate offers the possibility to omit air flow in/out of the air cushion. A movie from SimVis showing this in calm water is included on the attached CD-ROM. A freeze-frame from this is shown in Figure 51.

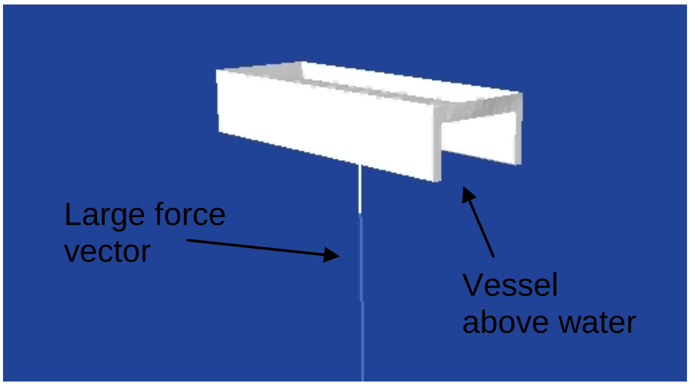


Figure 51: Case 1 with no air flow in/out from air cushion. This gives a too large vertical force. Here: The vessel is above water with a large pressure force telling it to come down. The full video is on the attached CD-ROM

Air flow in/out of the air cushion is what damps the system. Without this, the model got extremely unstable with the ship oscillating extremely more and more vertically, before flipping totally out of control. This is of course extremely unphysical and can never be validated by a full-scale model, but the result is considered numerically credibly. The magnitude of the air cushion force is a lot larger than the hydrodynamic damping forces. Therefore, the vessel oscillates out of control.

6.3 Seakeeping

Seakeeping calculations were carried out to control the program. Responses from VeSim for different wave heights are compared to linear results from VERES. The smallest wave height, 0.1 [m], is used in hope of getting results similar to those of VERES (low wave height should give approximately linear results). The largest wave heights (respectively 3 and 2 [m]) are used to trigger non-linearities. The last one, 1 [m], is used as an intermediate value. Air cushion forces are the only non-linear forces present in the calculations (all hydrodynamic forces are linear). This is used to easier see the non-linear effects of the air cushion.

Main RAO (Response Amplitude Operator) functions have been found for main motions (heave and pitch) and air cushion pressure, head sea. Three velocities have been used for case 1.1, and two for case 3. Also a few other cases were tried.

VERES calculations were done for several other conditions than told above, and for case 1, 1.1 and 2. These include heave, pitch and air cushion pressure in head sea and heave and roll for beam sea. These results are included in the appendix D.

All VeSim results were obtained using regular long crested waves. When starting the runs, the vessel is at zero motion. After a transition phase steady state is achieved. The RAO values are defined as the amplitude between max and min response value at steady state. For heave and air cushion pressure this amplitude is divided with wave height. For pitch it is divided with the wave slope (= wave number (k) multiplied with wave height).

Hand calculations for cobblestone oscillations, as shown in chapter 2.2, are carried out for all cases.

6.3.1 Case 1.1. Simple SES

The static water level difference between inside and outside air cushion:

$$h = \frac{p_0}{\rho g} = \frac{4000}{1025 \cdot 9.81} \approx \underline{0.398[m]} \tag{6.2}$$

Uniform pressure resonance:

$$\omega_n = \sqrt{\frac{A_b \gamma (p_0 + p_a)}{M h_b}} = \sqrt{\frac{240 \cdot 1.4(4000 + 101325)}{140000 \cdot 2.70}} \approx \underline{9.68[Hz]} \tag{6.3}$$

Spatial pressure resonance:

1. node:

$$\omega_n = \frac{nc\pi}{L_c} = \frac{340 \cdot \pi}{30} \approx \underline{35.60\ [Hz]} \tag{6.4}$$

2. node:

$$\omega_n = \frac{nc\pi}{L_c} = \frac{2 \cdot 340 \cdot \pi}{30} \approx \underline{71.21\ [Hz]} \tag{6.5}$$

Beam sea,
1.node:

$$\omega_n = \frac{nc\pi}{B_c} = \frac{1 \cdot 340 \cdot \pi}{8} \approx \underline{133.52\ [Hz]} \tag{6.6}$$

Table 18: Case 1.1. Cobblestone oscillations, resonance frequencies in different operating conditions

	ω_n [Hz]	f_n [Hz]	U = 0 [kn]		U = 15 [kn]		U = 27 [kn]	
			ω_0 [Hz]	λ [m]	ω_0 [Hz]	λ [m]	ω_0 [Hz]	λ [m]
Acoustic waves, n=1	35.60	5.67	35.60	0.05	6.12	1.64	4.67	2.82
Acoustic waves, n=2	71.21	11.33	71.21	0.01	8.90	0.78	6.75	1.35
Acoustic waves, beam sea n=1	133.52	21.25						
Uniform pressure	9.67	1.54	9.67	0.66	2.93	7.19	2.28	11.81

Table 19: Case 1.1. VeSim settings

Setting	Value	Unit
Air cushion active	True	[-]
Calculate air cushion forces using p_c (false: use p_0)	True	[-]
Calculate air flow in/out from air cushion	True	[-]
Condition on cushion. False: Off cushion	True	[-]
Include added viscous resistance	True	[-]
Number of fans active	2	[-]
Static over pressure	4000	[Pa]
Include first order wave forces	True	[-]
Include slowly varying wave drift forces	False	[-]
NMRIWF	False	[-]
Significant wave height, Hs	0.1, 1 and 3	[m]
Wave direction	0	[°]
Wave spectrum	Regular waves	[-]
Initial forward speed	0, 7.72, 13.89	[m/s]
Initial heading	0	[°]

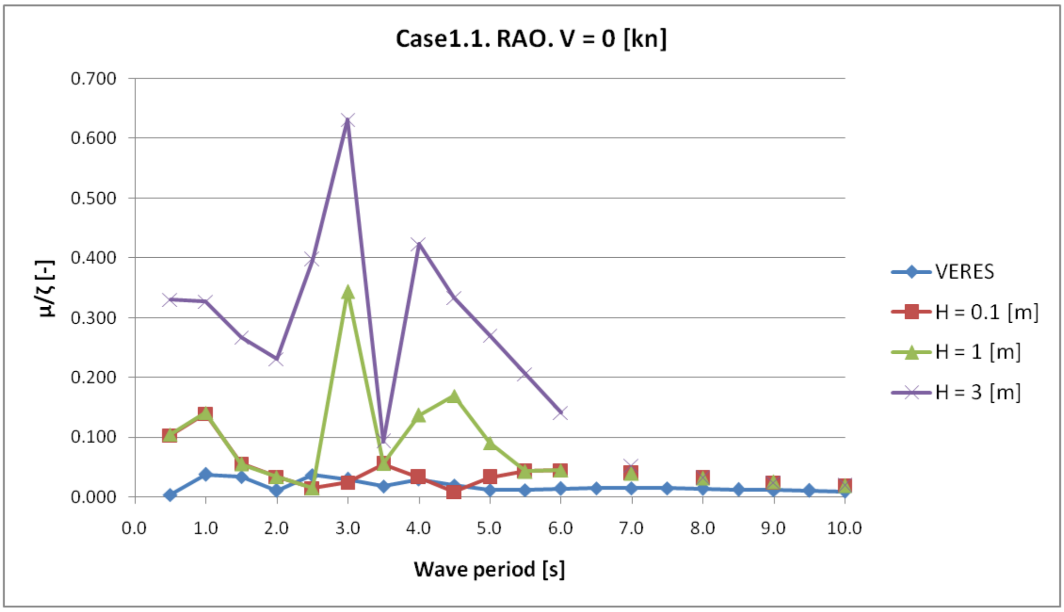


Figure 52: Case 1.1. RAO. Over pressure. V = 0 [kn]

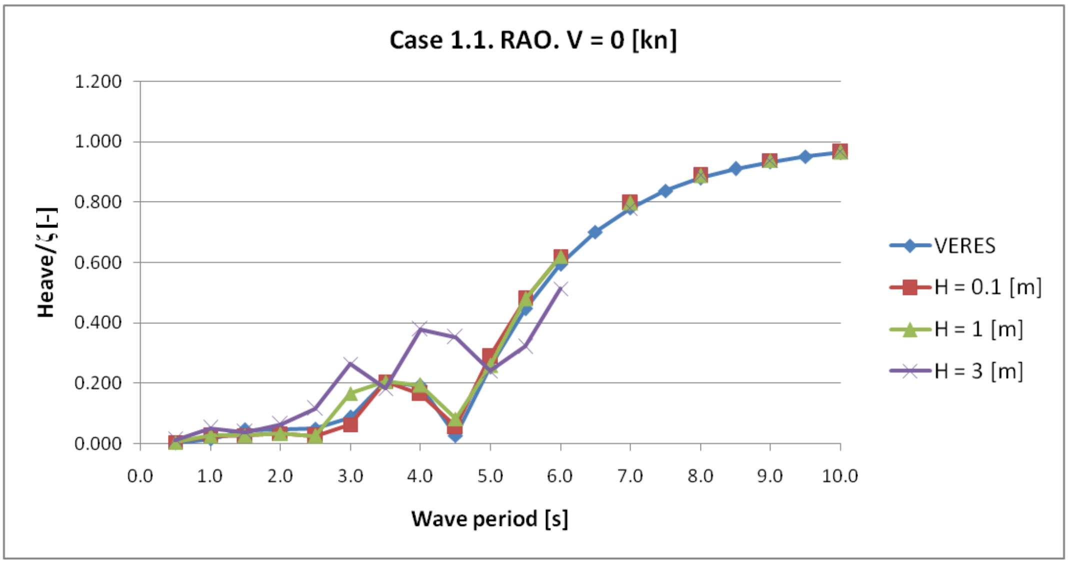


Figure 53: Case 1.1. RAO. Heave motion. V = 0 [kn]

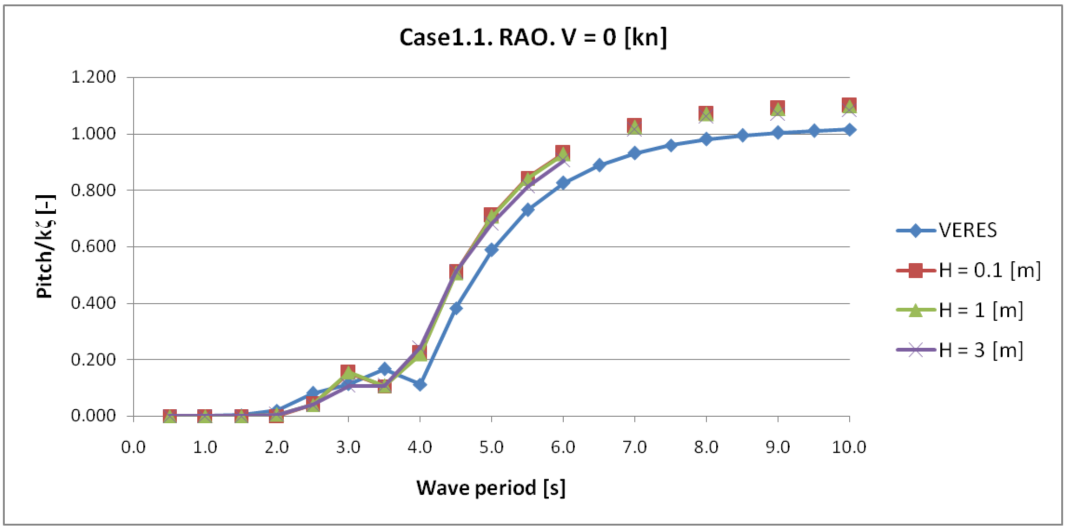


Figure 54: Case 1.1. RAO. Pitch motion. V = 0 [kn]

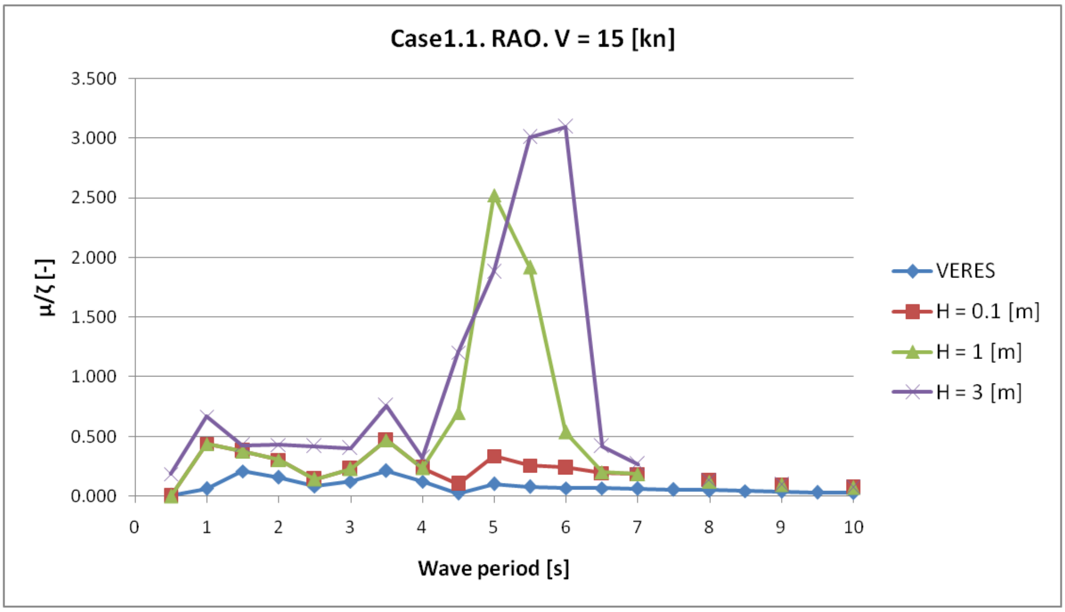


Figure 55: Case 1.1. RAO. Over pressure. V = 15 [kn]

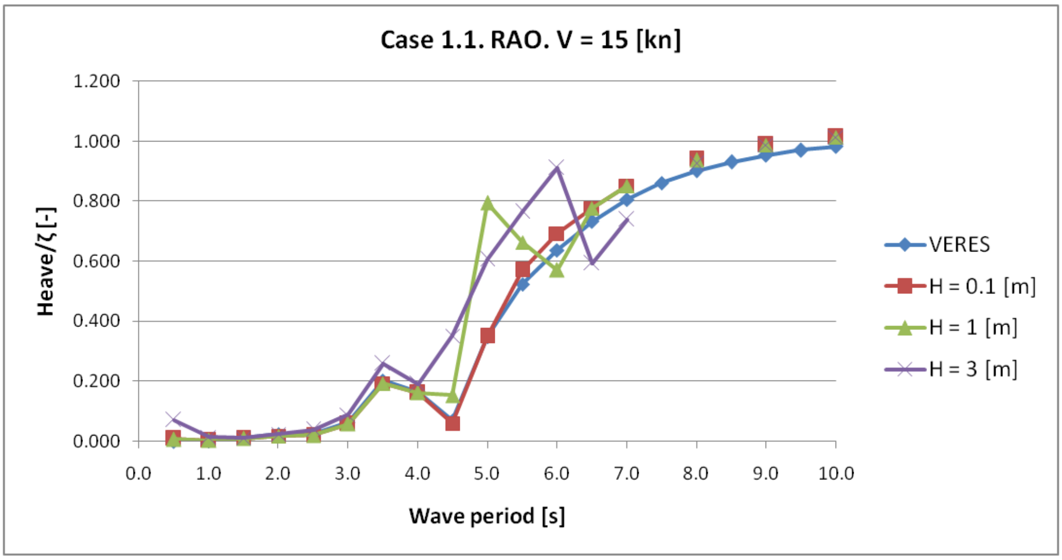


Figure 56: Case 1.1. RAO. Heave motion. V = 15 [kn]

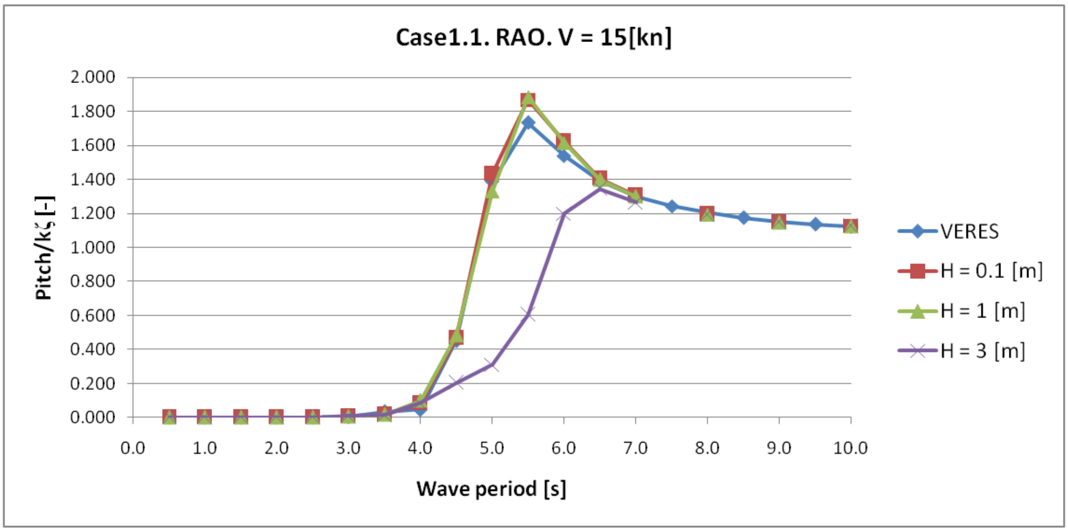


Figure 57: Case 1.1. RAO. Pitch motion. V = 15 [kn]

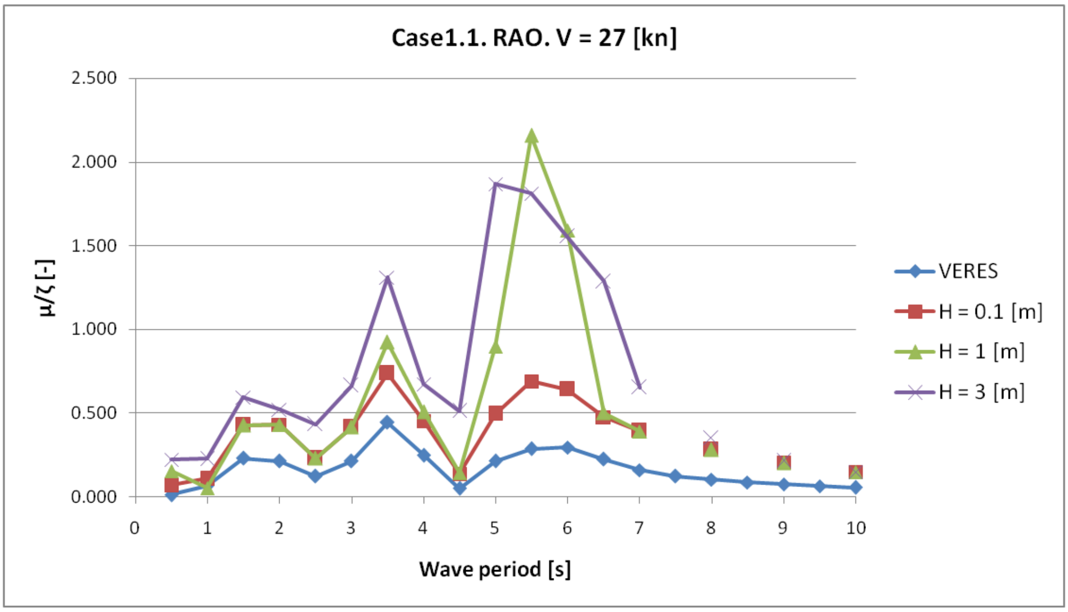


Figure 58: Case 1.1. RAO. Over pressure. V = 27 [kn]

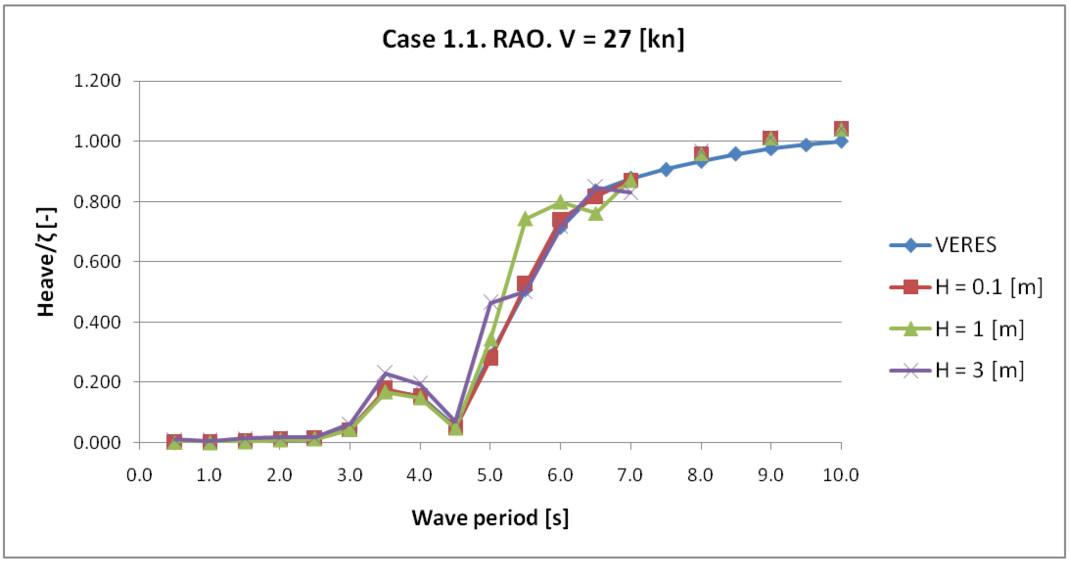


Figure 59: Case 1.1. RAO. Heave motion. V = 27 [kn]

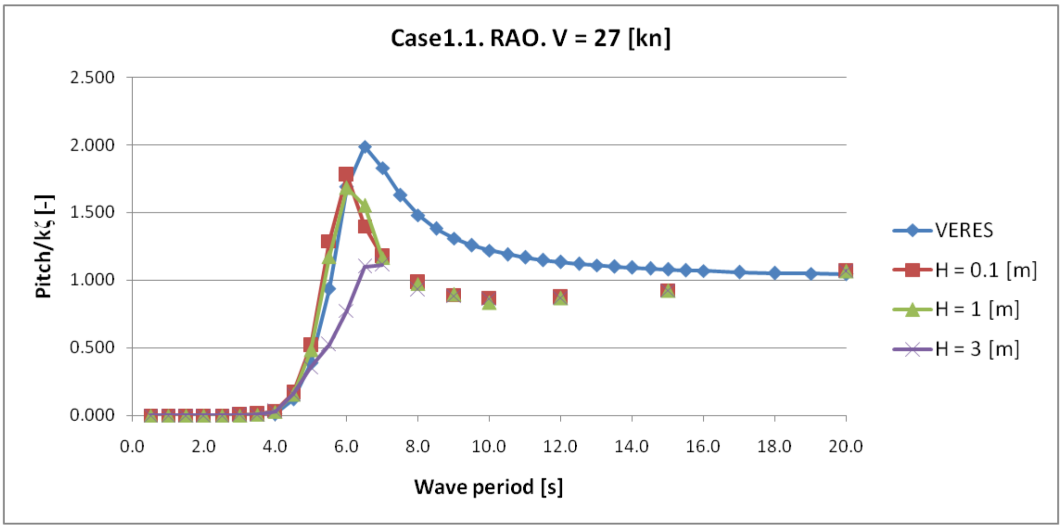


Figure 60: Case 1.1. RAO. Pitch motion. V = 27 [kn]

6.3.2 Error in fan system

There was, less than a day before delivery of this thesis, discovered an error in the fan characteristic code. This made the fan system deliver a lot more air then it should at high air cushion pressure (at mean air cushion pressure the error was small. This was the reason that the error was discovered this late). There was this late in the work no time to repeat the runs with correct fan system. Therefore a few plots were developed to investigate the error's influence.

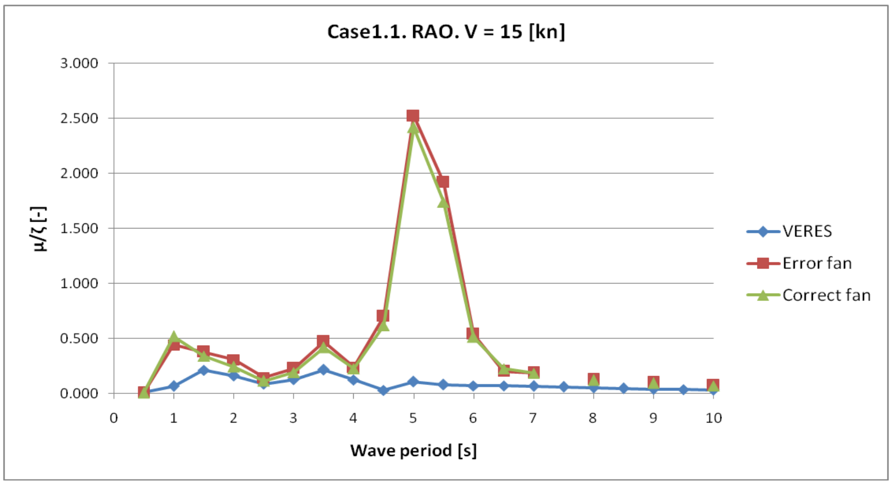


Figure 61: Case 1.1. Fan error. Air cushion pressure RAO, head sea, vessel speed 15 [kn]

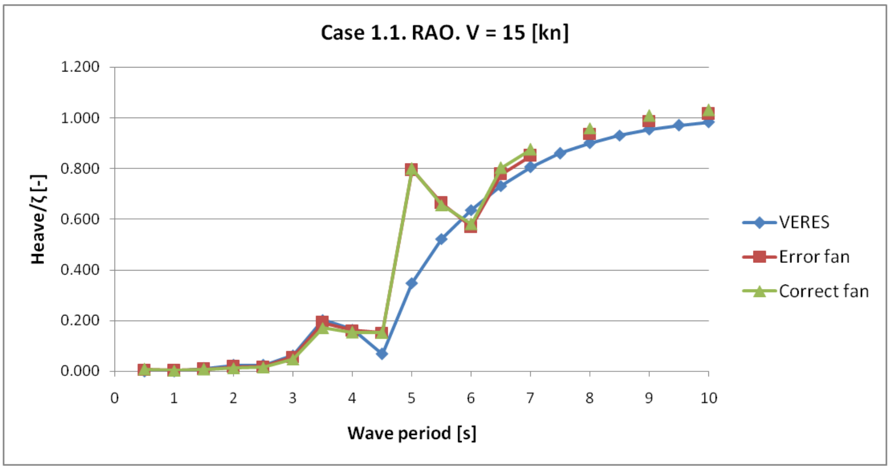


Figure 62: Case 1.1. Fan error. Heave RAO, head sea, vessel speed 15 [kn]

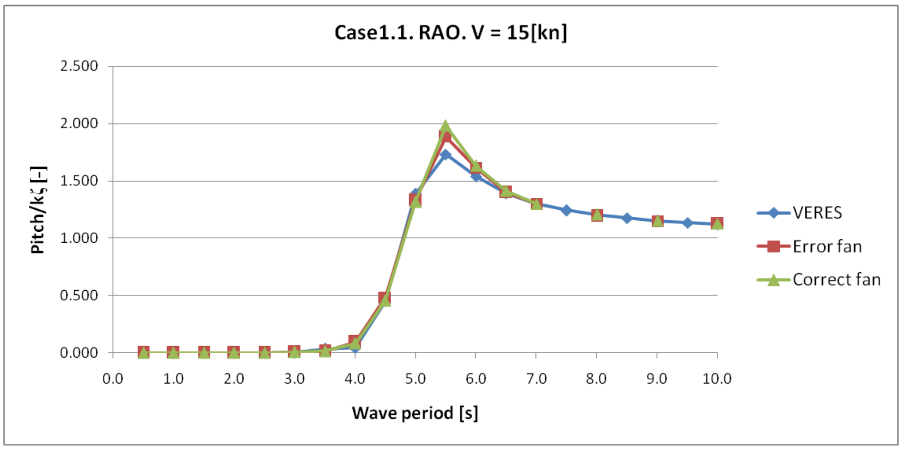


Figure 63: Case 1.1. Fan error. Pitch RAO, head sea, vessel speed 15 [kn]

6.3.3 More leakage area

Case 1.1 was analyzed with the seals placed higher. This makes leakage start occur at smaller motions. Both seals were placed 0.9 [m] from baseline (about .1 [m] initial leakage height between water surface and seal).

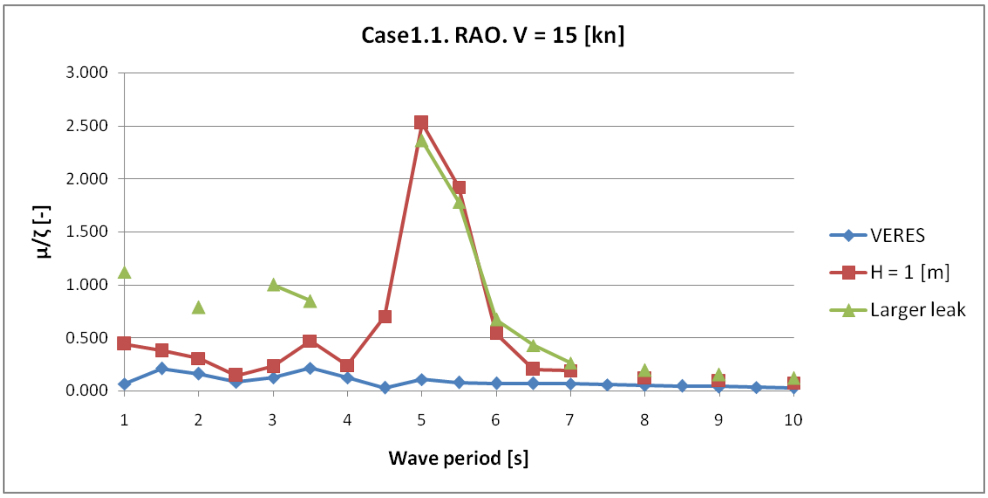


Figure 64: Case 1.1. Larger leakage area. Air cushion pressure RAO, head sea, vessel speed 15 [kn]

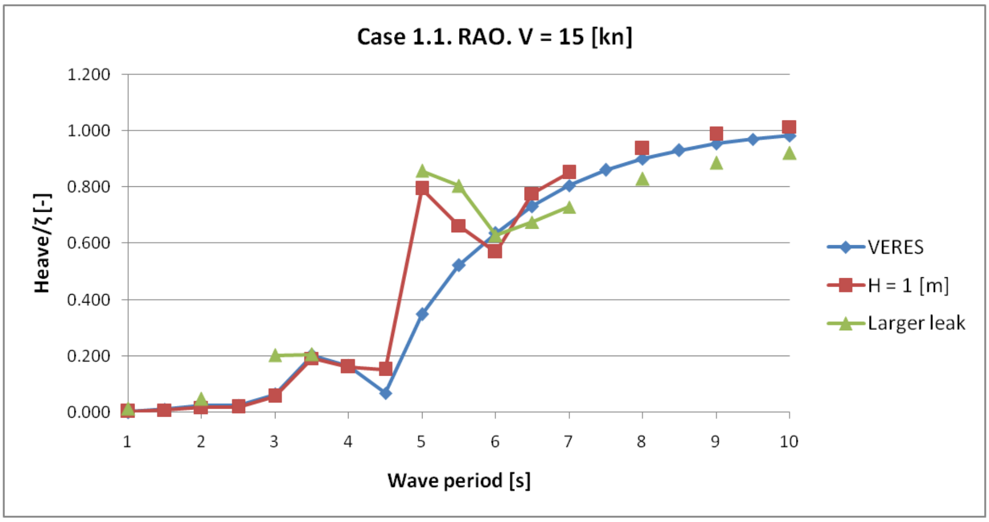


Figure 65: Case 1.1. Larger leakage area. Heave RAO, head sea, vessel speed 15 [kn]

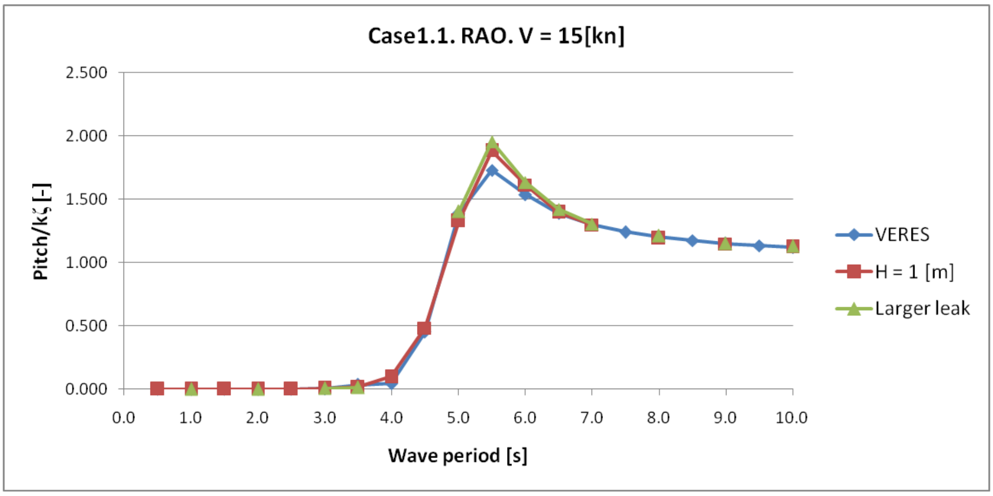


Figure 66: Case 1.1. Larger leakage area. Pitch RAO, head sea, vessel speed 15 [kn]

6.3.4 Case 2. Inclined SES

The static water level difference between inside and outside air cushion:

$$h = \frac{p_0}{\rho g} = \frac{8000}{1025 \cdot 9.81} \approx \underline{0.796[m]} \tag{6.7}$$

Uniform pressure resonance:

$$\omega_n = \sqrt{\frac{A_b \gamma (p_0 + p_a)}{M h_b}} = \sqrt{\frac{244.13 \cdot 1.4(8000 + 101325)}{277000 \cdot 3.80}} \approx \underline{5.96\ [Hz]} \tag{6.8}$$

Spatial pressure resonance:

1. node:

$$\omega_n = \frac{nc\pi}{L_c} = \frac{340 \cdot \pi}{30} \approx \underline{35.60\ [Hz]} \tag{6.9}$$

2. node:

$$\omega_n = \frac{nc\pi}{L_c} = \frac{2 \cdot 340 \cdot \pi}{30} \approx \underline{71.21\ [Hz]} \tag{6.10}$$

Beam sea,
1.node:

$$\omega_n = \frac{nc\pi}{B_c} = \frac{1 \cdot 340 \cdot \pi}{8.14} \approx \underline{131.22\ [Hz]} \tag{6.11}$$

Table 20: Case 2. Cobblestone oscillations, resonance frequencies in different operating conditions

	ω_n [Hz]	f_n [Hz]	U = 0 [kn]		U = 15 [kn]		U = 30 [kn]	
			ω_0 [Hz]	λ [m]	ω_0 [Hz]	λ [m]	ω_0 [Hz]	λ [m]
Acoustic waves, n=1	35.60	5.67	35.60	0.05	6.12	1.64	4.45	3.11
Acoustic waves, n=2	71.21	11.33	71.21	0.01	8.90	0.78	6.42	1.50
Acoustic waves, beam sea n=1	131.26	20.89						
Uniform pressure	5.96	0.95	5.96	1.73	2.19	12.85	1.65	22.50

Figure 67: Case 2. Natural periods from VERES

		Natural periods [s]			
Speed [kn]	Heading [°]	Heave	Roll	Pitch	Cushion pressure
0	0	1.5 & 4		3.5	1.5 & 3 & 4 & 6.5
0	90	NA	5		0.5 & 1.5 & 3
15	0	2 & 3.5		5.5	2 & 3.5 & 5.5
15	90	NA	5		0.5 & 1.5 & 3
30	0	2.5 & 4		7	2.5 & 3.5 & 5.5
30	90	NA	4.5		0.5 & 1.5 & 3

Linear VERES results are shown in appendix D.

6.3.5 Case 3. MCMV

The same VeSim settings as for case 1.1 are applied to case 3. Exceptions are vessel speeds (here 0 and 18 [kn]) and maximum wave height used. The difference in geometries makes non-linearities appear in case 3 at lower wave height. For confidential reasons, now more information about conditions and settings are given for case 3. Calculations of cobblestone oscillations are also excluded.

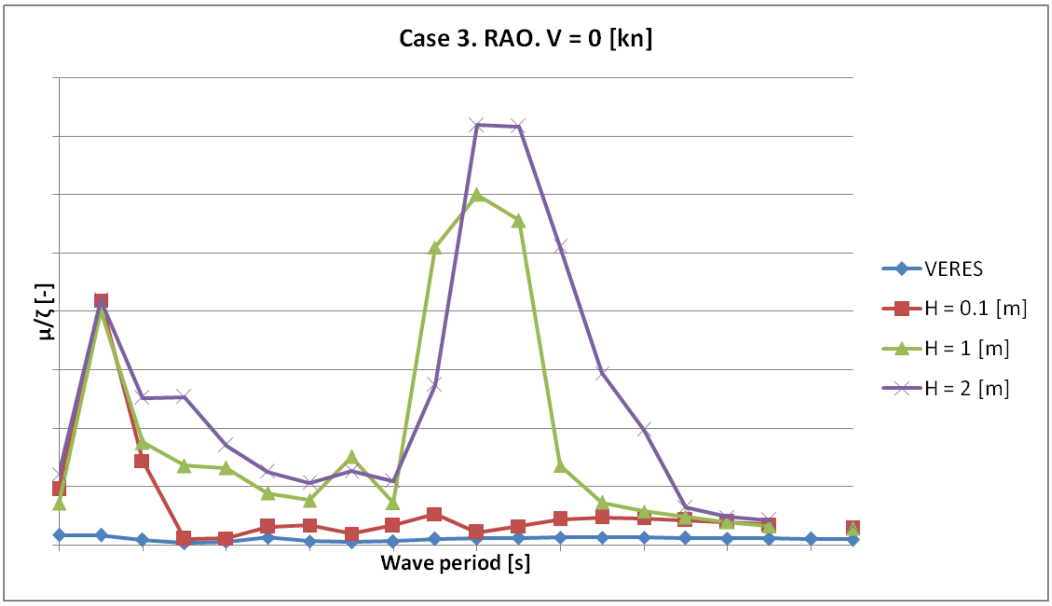


Figure 68: Case 3. Air cushion pressure RAO, head sea, vessel speed 0 [kn]

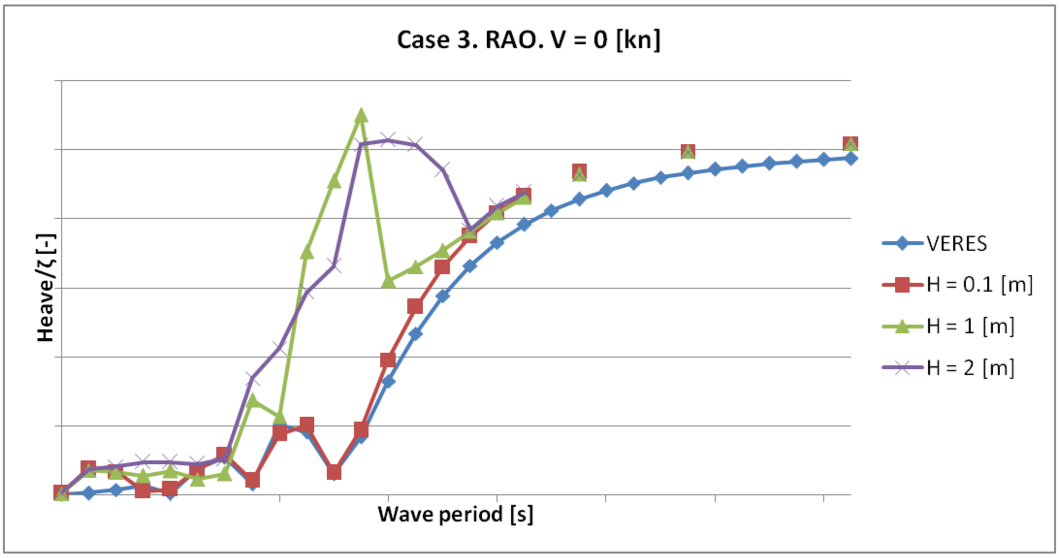


Figure 69: Case 3. Heave RAO, head sea, vessel speed 0 [kn]

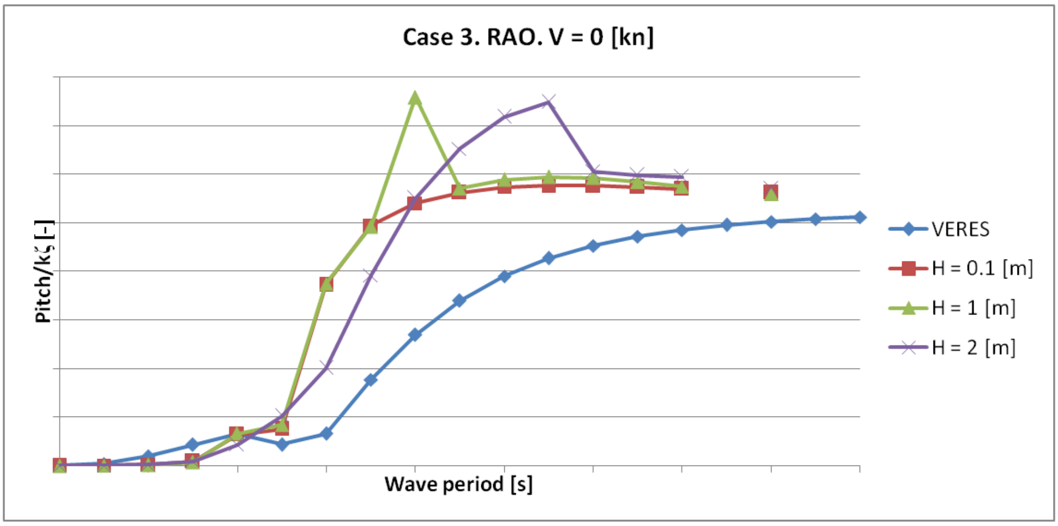


Figure 70: Case 3. Pitch RAO, head sea, vessel speed 0 [kn]

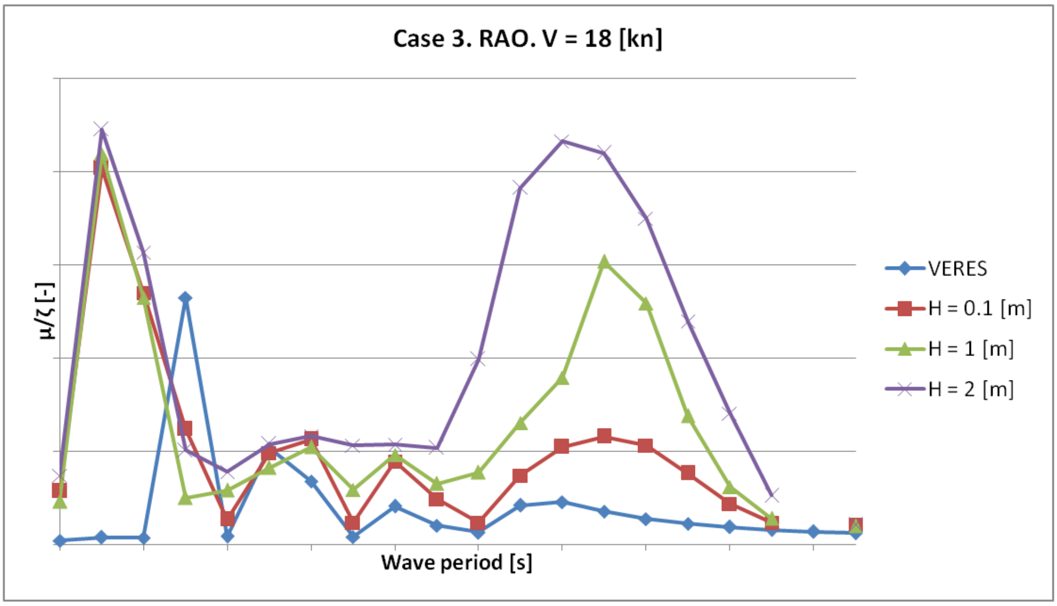


Figure 71: Case 3. Air cushion pressure RAO, head sea, vessel speed 18 [kn]

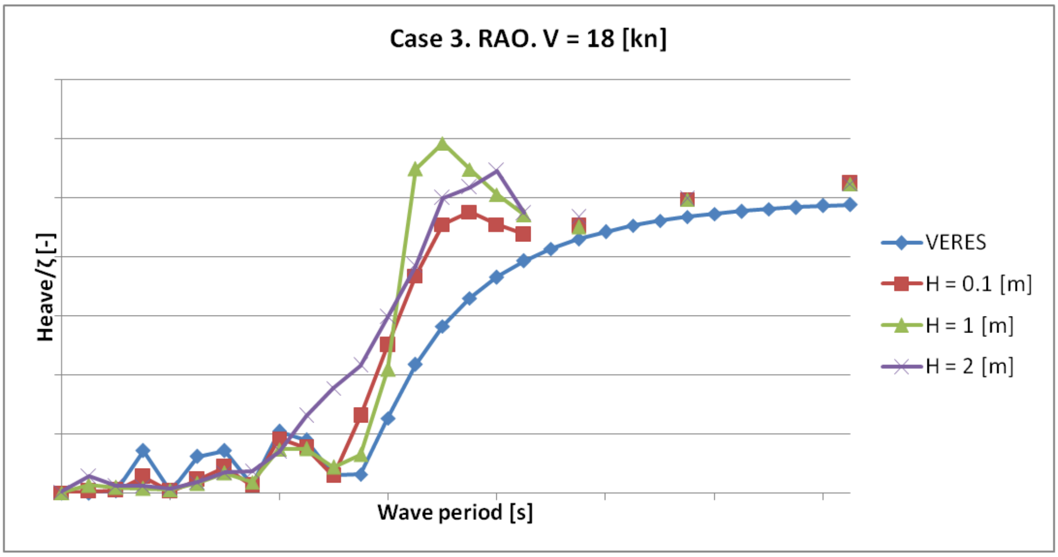


Figure 72: Case 3. Heave RAO, head sea, vessel speed 18 [kn]

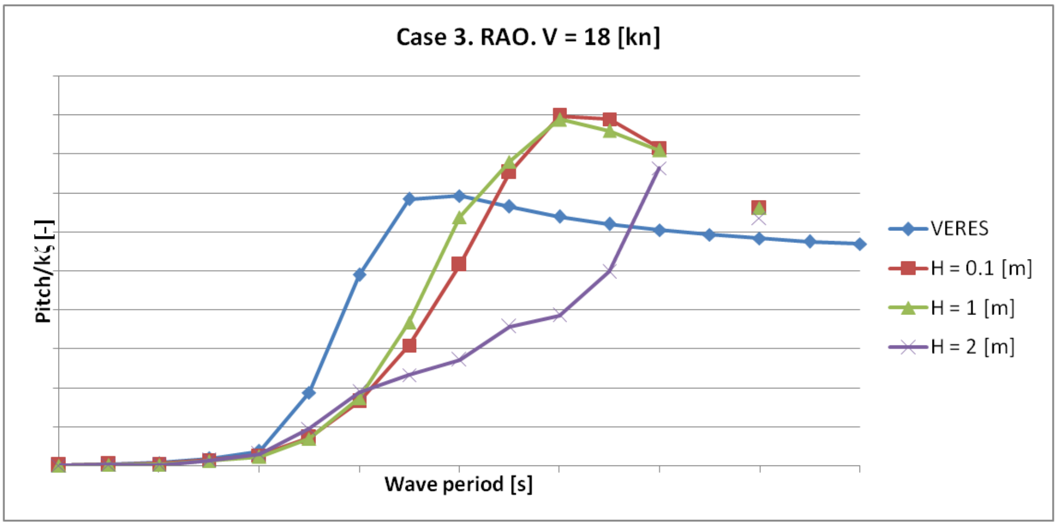


Figure 73: Case 3. Pitch RAO, head sea, vessel speed 18 [kn]

6.4 Resistance

These calculations are only too show tendencies in order to test the program. In the plots, wetted surface is defined as mean wetted surface. This was measured after the initial phases in the VeSim runs. Wetted surface is made non-dimensional using equation 3.10.

VeSim needed resistance data as input. This is supplied in the maneuvering file (HullInput.hsm. The files used for case 1, 1.1 and 2 are supplied on the attached CD-ROM). The resistance can be defined by a polynomial function in the maneuvering file. This was here solved extremely simple. Using mean wetted surface, viscous resistance was calculated for max vessel speed. Than a linear curve was calculated, assuming that the viscous resistance amounts to half the resistance at full vessel speed. Calculations are included on the "resistance.xls" file on the attached CD-ROM.

6.4.1 Case 1.1. Simple SES

Main VeSim settings are same as in the seakeeping analysis.

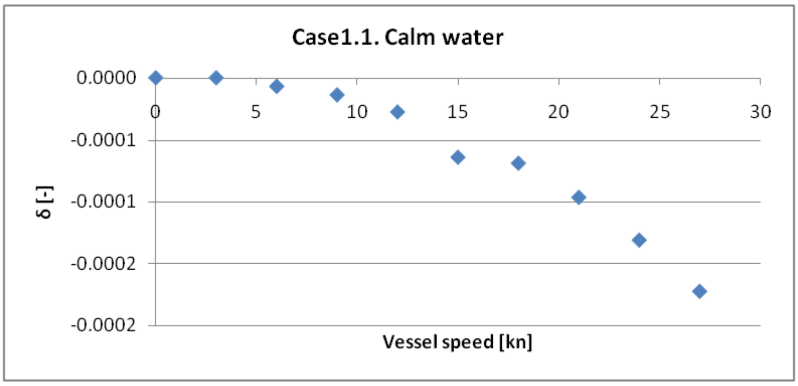


Figure 74: Case 1.1. Non-dimensional wetted surface vs. vessel speed, calm water, VeSim

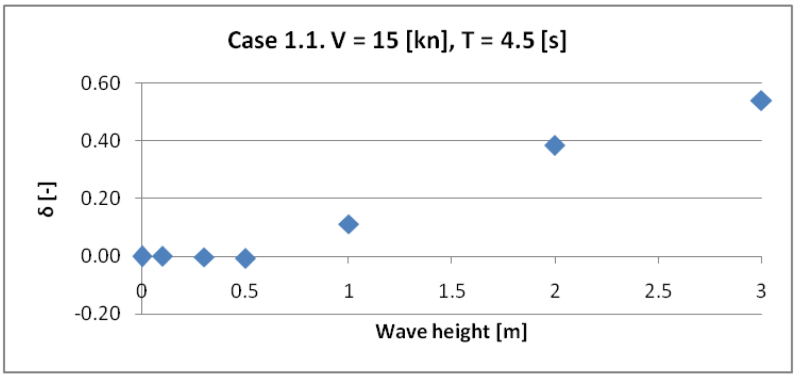


Figure 75: Case 3. Non-dimensional wetted surface vs. wave height. Wave period 4.5 [s], vessel speed 15 [kn]

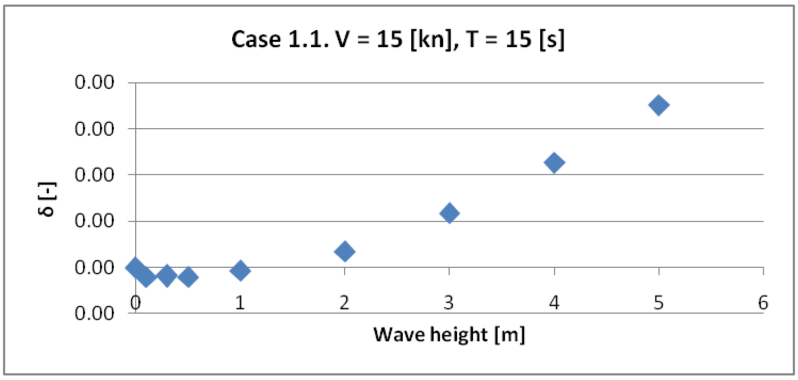


Figure 76: Case 1.1. Non-dimensional wetted surface vs. wave height. Wave period 15 [s], vessel speed 15 [kn]

6.4.2 Case 3. MCMV

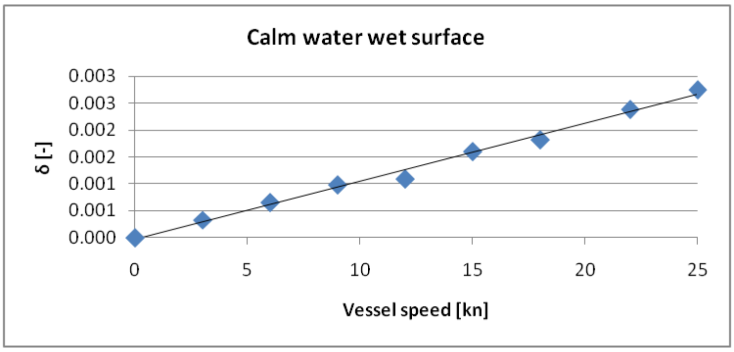


Figure 77: Case 3. Non-dimensional wetted surface vs. vessel speed, calm water, VeSim

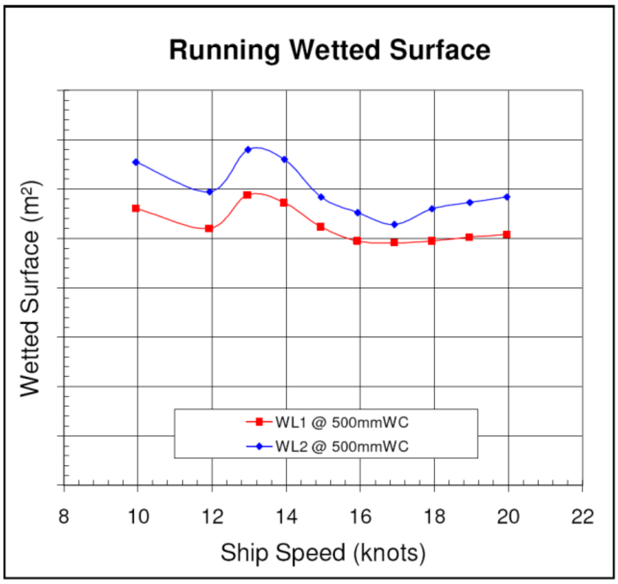


Figure 78: Case 3. Wetted surface versus ship speed, calm water, model test [WINES et.al. 07]. Plots for two different load conditions

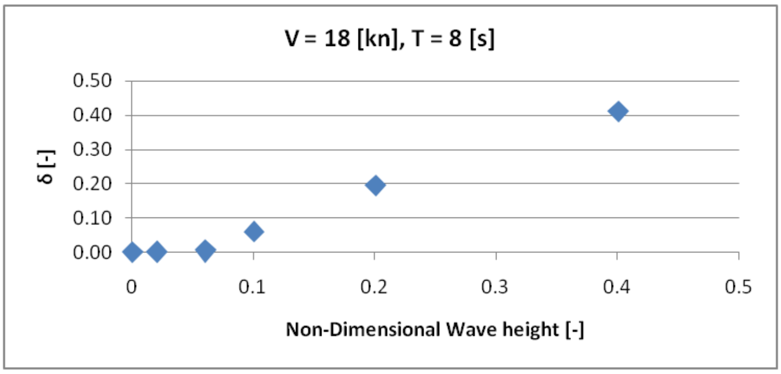


Figure 79: Case 3. Non-dimensional wetted surface vs. non-dimensional wave height. Wave period 8 [s]

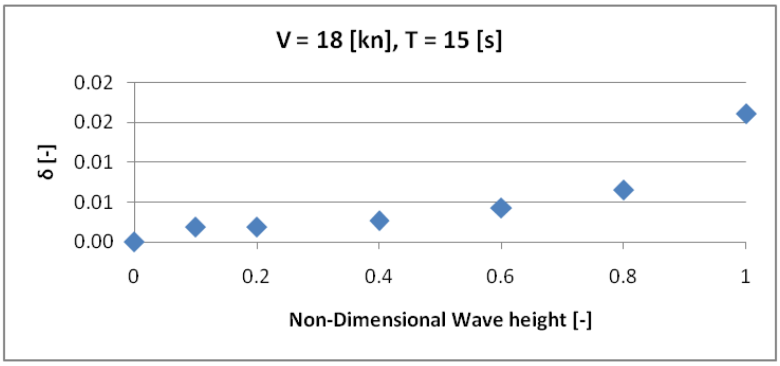


Figure 80: Case 3. Non-dimensional wetted surface vs. non-dimensional wave height. Wave period 15 [s]

Figure 79 and Figure 80 has the same non-dimensional wave height on the x-axis. Fewer measurements were done with wave period equal 8 [s], as the motions became too violent at high wave height (slamming occurred).

6.5 Discussion of seakeeping and resistance results

6.5.1 Error sources

Potential human error sources in the measurements:

- Measurements made before the steady state motion were obtained
- Misspelling of values
- Mix-up of results, writing values in wrong tables

The air flow from the fan system is wrong. This might give some errors, although Figure 61 to Figure 63 suggest that any errors are only marginal.

6.5.2 Seakeeping

The results can be judged relative to VERES results by two criteria:

A. Do the natural periods appear at the same frequencies?
B. Are the response amplitudes of the same order of magnitude? The amplitude is defined as the difference between maximum and minimum response. This gives thus a results that is equal or larger than the "real" value (at some conditions the response history does not take a sinusoidal shape)

Influence of geometry
More non-linearities are expected for case 3 than for case 1.1. This is because the hull sides are vertical on case 1.1, while they are not for case 3. Also for case 1.1, the air cushion forces attacks through center of gravity. As the vessel is symmetric about amidships, and the hull sides vertical, air cushion pitch forces do not appear as long as the wet-deck is dry. For case 3, air cushion forces attack at a position different from center of gravity, making the air cushion induce pitch forces.

The influence of air cushion pitch forces is obvious in the plots for case 3. As large air cushion forces appear, these induce large pitch forces on the vessel. Large pitch motions give large leakage under seal. And with the non-straight hull sides pitch also gives alteration in water plane area. These two effects will on the other hand give larger variations in air cushion pressure. The author believe that it is this double-acting effect that gives the large variations between VERES and VeSim results for pitch for case 3.

Another example of non-linearities is shown in Figure 69 and Figure 72, compared to e.g. Figure 53. At high wave periods, restoring terms are dominating. Restoring terms in the air cushion is the cushion itself, giving heave force if forced up or down. Case 3 has "stiffer air cushion" than case 1.1, because the water plane area decreases if the draught is increased. This gives larger change of air cushion volume than for case 1, and thus stiffer heave motion (more have force pr heave motion) compared to the linear VERES results (giving larger heave RAO value).

Small wave height, "linear results"
At small wave height, both frequencies and magnitude fit well. The air cushion pressure is at some conditions found to have higher amplitude than VERES. This is seen to affect the heave response, as one would expect. The deviation is larger for case 3 than for case 1.1. This is because the hull sides of case 3 are not strictly vertical (introducing non-linearities).

Large wave height, "non-linear results"
At higher wave height non-linearities appears. The reasons are that the ship sides are not vertical (for case 3) and because the waves hit the wet-deck (slamming). The hydrodynamic forces do not take the wet-deck into account, but the air cushion forces do. When the water hits the wet-deck, the air cushion volume decreases rapidly, giving possibly high change of air cushion pressure (depends on leakage and the general motions). Also the air cushion gives zero forces on wet surface.

For case 1.1, no air cushion pitch forces will appear before the wet-deck is wetted. This can be used to measure when slamming occurs. Two videos on the attached CD-ROM show case 1.1 in waves that hit the wet-deck.

The results of water hitting the wet-deck can clearly be seen in the different plots. It gives:
- Larger changes in air cushion pressure
- Larger changes in heave force (lift force is lost as the wet-deck becomes wet)
- Pitch forces are reduced or increased. This depends on which parts of the wet-deck that becomes wet, and if the pressure is larger or smaller than mean pressure value

Air flow, leakages
The error in fan system was investigated in Figure 61 to Figure 63. The results suggest that altering air flow through fan system will only do minor changes to motions of the vessel. In

Figure 64 to Figure 66 the seals are lifted, allowing more leakage to occur at larger water depths. It is clearly seen that the effect gives larger air cushion pressure variations, as well as larger heave forces. Pitch is as expected unchanged (point of attack is center of gravity, giving zero pitch forces).

At higher wave periods, it is seen that the heave motion is less. This is because of non-linearities. If the vessel is lifted (relative wave elevation) more leakage will occur, forcing the vessel downwards while reducing air cushion pressure. With the original seal placement more motion was needed to obtain larger leakage. This makes the air cushion less “stiff”, and gives, as opposed to what discussed under “influence of geometry”, lower RAO heave values.

It is seen that the height of the seal is important to model the air cushion pressure and heave motion correctly, obtaining correct restoring air cushion force.

Air cushion pressure

One main difference from VERES to VeSim is that VeSim gives larger air cushion pressure variations. The difference increases with increasing wave height. The air cushion pressure RAO has several peaks, and it is especially at the one at largest wave periods where the pressure peak is large.

Comparing air cushion pressure with pitch RAO, it is seen that large pitch coincides with large variations in air cushion pressure. I.e. large pitch motions give large variations in leakages which give large air cushion pressure variations. This is in accordance with observations in SimVis.

The peaks at smaller wave periods are found to be dependent on the seal model. Following Figure 64, larger leakage area gives larger variations in air cushion pressure. It is also seen in the accompanying heave RAO (Figure 65), and also the various other heave RAO plots, that the large pressure peaks at low wave periods do not give large heave motions. This complies with what discussed earlier in chapter 2.2, that high frequency pressure oscillations (cobblestone oscillations) do not give large heave motions. It will though give large heave accelerations. A dynamic seal system and ride control system should be incorporated to deal with this.

Pitch motion for case 1.1

The different VeSim pitch RAO for case 1.1 fits very well with VERES results for 0 and 15 [kn]. At 30 [kn] (Figure 60) some deviation occur in the area dominated by restoring forces. The deviation is the same for all wave heights. This may imply that the restoring forces are relatively too small compared to mass forces. Dariusz Fathi [FATHI] suggested checking if the cg_sog (speed over ground at center of gravity) value had converged. The excitation forces are calculated using this speed.

A quick control was performed. It turned out that cg_sog converges very slowly, and that it thus probably had not converged for the values used in Figure 60. This improved the results somewhat, but only decreased the deviation between VeSim and VERES with about 25% percent. The question therefore remains unsolved.

The values in Figure 60 are the original ones.

6.5.3 Resistance

At calm water, the results show that wet surface changes marginally with increasing speed. The order of magnitude is much less than 1 %. The slight change may be due to added viscous resistance forces, giving the vessels a small trim.

As seen from model testing, Figure 13 (wave pattern along hull side at different vessel speed in calm water) and Figure 78 (wetted surface vs. vessel speed in calm water), no or very small increase or decrease of wet surface should be expected. Model tests show further that change of wet surface is linked to the created wave systems. In the simulator these are not taken into account.

Conclusion is that calculated calm water wet surface is credible, taking the assumptions on wave elevation into account.

The added viscous resistance in waves calculations shows increasing wet surface with increasing wave height. This is as expected. More waves give more leakage, which give larger mean draught. Larger draught gives larger mean wet surface. It is also shown a relation between wave period and wetted surface. The wave periods giving the most violent motions (this is seen by comparing with seakeeping calculations) gives larger mean wet surface.

Model behaves as expected. Increased wave height increases viscous resistance. The order of magnitude is however not to be trusted. For a more correct estimate of wet surface, wave creation due to air cushion pressure and hull must be included in the simulator.

Conclusion

A simulation program dealing with time-domain seakeeping for a SES has been developed. The program utilizes the Marintek developed simulator VeSim (Vessel Simulator). VeSim calculates all hydrodynamic forces and the motions of the vessel. The SES program provides SES specific forces to the simulator. These include air cushion forces and moments and added viscous resistance in waves. All modeling of the air cushion is done in the SES program. A long term goal is to fully incorporate the SES program into the Marintek ShipX hydrodynamic workbench. Now input to VeSim is made in ShipX, and the simulator can also be started from here. The input data to the SES program must be made separately.

In the simulator, all forces and moments can be calculated in real-time. Using a visualization program (SimVis), the actual waves and motions of the vessel can be seen.

Effects included in the program, all dynamically calculated, are

- Size of enclosed air cushion volume
- Leakages from under seals and hull
- Air flow through fans, based on fan characteristics
- Wet hull surface, used to calculate viscous resistance

Fore and aft seals are modeled as rigid vertical walls.

The developed software has been tested using some simple SES designs, as well as a real vessel, the Alta class Mine Counter Measure Vessel. Seakeeping results for these vessels have been compared to linear calculations from the Marintek developed program VERES (Vessel Response program). The results show good resemblance. Non-linear effects on results have been identified. These include effects from non-vertical side hulls, wet-deck slamming and seal models. The effects from the SES program on seakeeping calculations are as they should be, given the simplifications and modeling.

The SES program allows for calculations of added viscous resistance in waves. Results show that this works as expected and intended. Increased wave height gives larger wetted surface, which gives added viscous resistance. It is also seen that the wave periods giving the most violent motions are also the one giving most added viscous resistance. The magnitude of the forces are however wrong. Dynamic wave elevation is modeled as being only due to incident waves (air cushion pressure giving a static uniform contribution to wave elevation). Wave elevation due to air cushion pressure and the hull must be found to improve resistance calculations.

The work with the program is judged a success. A functional program has been developed. It captures the most essential effects of a SES. The necessary theory has been found and derived. The program is however far from finished, as lots of effects have been abandoned. The next chapter lists several possibilities for future work and progress.

Further development of the air cushion federate

This is a short and incomplete list of suggestions to how to further develop the air cushion federate. The suggestions are discussed throughout the report and are here summarized.

- Incorporate a ride control system. This is very important in order to damp cobblestone oscillations
- More accurate wave elevation. Now only incoming incident waves are considered, and the air cushion pressure field gives a simple constant change in wave elevation all over the cushion area. Other effects that can be included:
    - Diffraction waves
    - Radiation waves
    - Waves due to air cushion pressure. That is, replace the constant change of wave elevation with a more complex and accurate system
    - Interaction effects between demi hulls and/or other wave systems
- Slamming calculations, slamming on the wet deck
- Include seal dynamics. To give more accurate air leakage. This is also connected to the cobblestone acoustic resonance periods. Will improve motions of the SES, especially in low sea states
- Make a maneuvering model suitable for catamarans and SES. This is one of the inputs to the simulator
- Refine java code used in the air cushion federate. Both to make it more efficient and to make a more logic build-up
- Make a streamlined transfer of model and air cushion setup from ShipX to the simulator. Now all input related to the air cushion federate must be manually established
- Calculate spatially varying air cushion pressure, not just uniform air cushion pressure. This is mostly important for high frequency pitch motions
- Validate the simulation comparing the results with full scale data
- Make the fan characteristic more complex by allowing change of rotational speed
- Add propulsion systems. Using water jet – find ventilation of the water jet system
- Make complete resistance calculations

Tables

Figures

Bibliography and references

Literature

[FATHI 07 TMR] Dariusz Fathi, “ShipX Vessel Response program (VERES)”, Lecture in subject TMR15 Numerical Methods in Marine Hydrodynamics, autumn 2007

[STEEN 07 TMR] Sverre Steen, ”Air Cushions Catamarans, Surface Effect Ships (SES)”, Lecture note, TMR4220 Naval Hydrodynamics, spring 2007

[WINES et.al. 07] Christian Wines, Sverre Steen, Magnus Tvete, Hans Olav Midtun, “Influence of Increased Weight on SES-performance in a Seaway”, Proceedings FAST2007, Shanghai, China, September 2007

[MINSAAS 06] Knut J. Minsaas, “Supplementary Lecture Notes, Naval Hydrodynamics, Ship Resistance”, February 2006, Marine Technology Centre, Trondheim, Norway, Department of Marine Technology

[FALTINSEN 05] Odd M. Faltinsen, “Hydrodynamics of High-Speed Marine Vehicles”, Cambridge University Press 2005

[STEEN et.al. 01] Sverre Steen and Gisle Strand, ”SES Performance Evaluation in Model and Full Scale”, FAST 2001: 4th-6th September 2001, Southampton, UK

[ULSTEIN 95] Tore Ulstein, “Nonlinear Effects of a Flexible Stern Seal Bag on Cobblestone Oscillations of an SES”, Dr.ing thesis, Department of Marine Hydrodynamics, The Norwegian Institute of Technology, 1995

[STEEN 93] Sverre Steen, “Cobblestone effect on SES”, Dr.ing thesis, Department of Marine Hydrodynamics, The Norwegian Institute of Technology, 1993

[FALTINSEN 90] Odd M. Faltinsen, “Sea Loads on Ships and Offshore Structures”, Cambridge University Press 1990

[HALVORSEN 07] Trygve Espeland Halvorsen, “Motion of a SES Vehicle in Waves”, Project thesis, Norwegian University of Science and Technology, 2007

Software documentation

Documentation included in the ShipX Workbench version 3.01.0047

[FATHI & HOFF 04] Dariusz Fathi and Jan Roger Hoff, “ShipX Vessel Response (VERES), Theory Manual”, MARINTEK AS, February 13, 2004

[FATHI 06] Dariusz Fathi, “ShipX Time-Series Analysis Plug-in”, 2006-03-28

[VERES SES] SES supplement to the VERES User’s Manual. No information given about author and date

[RINGEN 07 M] Edvard Ringen, “ShipX manoeuvering Plug-In”, 2007-08-20

[RINGEN, FATHI 07] Edvard Ringen, Dariusz Fathi, ”ShipX Vessel Simulator Plug-In”, 2006-03-08

Documentation included in the Marintek Vessel Simulator

[BJERMELAND & FATHI] Lasse Bjermeland and Dariusz Fathi, “Documentation of MARINTEK Vessel Simulator (VeSim) Version 3.1”, 2008-03-10

[HUSTELI XX] Nicolai Husteli, ”Common Simulation Interface, Documentation and Tutorial”, no information given about date

[FATHI VESIM1] Lasse Bjermeland, “Users’s Manual, Vessel Federate, MARINTEK Vessel Simulator - VeSim”, no information given about date

[FATHI VESIM2] Dariusz E. Fathi, “Users’s Manual, Wave Federate, MARINTEK Vessel Simulator - VeSim”, no information given about date

[HYDRO VESIM 04] 6DOF Vessel Simulator, Simulation of 6 DOF motion responses in waves, Preliminary, October 7, 2004

Conversations and e-mails

[PETTERSEN 08] Professor Bjørnar Pettersen
Marine Constructions, Department of marine engineering, NTNU
Otto Nielsens vei 10, NO-7052 TRONDHEIM, NORWAY
E-mail: bjornar.pettersen@ntnu.no

[FATHI 08] Research Manager Dariusz Erik Fathi
Department of Ship Technology, MARINTEK
P.O.Box 4125 Valentinlyst, NO-7450 TRONDHEIM, NORWAY
E-mail: Dariusz.Fathi@marintek.sintef.no

[HOFF 08] Senior Research Engineer Jan Roger Hoff
Department of Offshore Hydrodynamics, MARINTEK
P.O.Box 4125 Valentinlyst, NO-7450 TRONDHEIM, NORWAY
E-mail: Jan.R.Hoff@marintek.sintef.no

[WINES 08] Lieutenant-Commander Christian Wines
Master of Science/Siv.ing, teknisk saksbehandler
FLO/S-MK-IPS-SHR
E-mail: cwines@mil.no

[STEEN 08] Professor Sverre Steen
Marine Constructions, Department of marine engineering, NTNU
Otto Nielsens vei 10, NO-7052 TRONDHEIM, NORWAY
E-mail: sverre.steen@ntnu.no

[FALTINSEN 08] Professor Odd Magnus Faltinsen
Marine Constructions, Department of marine engineering, NTNU
Otto Nielsens vei 10, NO-7052 TRONDHEIM, NORWAY
E-mail: oddfal@ntnu.no

[BJERMELAND 08] Master of Science Lasse Bjermeland
Department of Marine Operations and Simulations, MARINTEK
P.O.Box 4125 Valentinlyst, NO-7450 TRONDHEIM, NORWAY
E-mail: lasse.bjermeland@marintek.sintef.no

[RINGEN 08] Master of Science Edvard Ringen
Department of Marine Operations and Simulations, MARINTEK
P.O.Box 4125 Valentinlyst, NO-7450 TRONDHEIM, NORWAY
E-mail: edvard.ringen@marintek.sintef.no

[SKOMEDAL 08] D.Eng Nere G. Skomedal
Head of Research and Development Umoe Mandal
Service-Box 902, NO-4509 MANDAL, NORWAY
E-mail: nere.skomedal@umoe.no

Web pages

[ITTC 99] http://ittc.sname.org/Loads%20and%20Responses.pdf
Date: 8.November 2007

[UMOE 08] http://www.mandal.umoe.no/WEB/um200.nsf/pages/mainframe
Date: 18.May 2008

Software

Autodesk AutoCAD 2007
Ceetron GLview Pro 6.5
MATLAB R2007b
Microsoft Office 2007 (Excel and Word)
NetBeans IDE 6.0.1 with Sun JDK (Java Development Kit) v1.6
ShipX, MARINTEK
Main plug-ins: Vessel Response program (VERES), Maneuvering, Panel generation, Ship model setup, Vessel simulator setup
Vessel Simulator (VeSim), MARINTEK

Appendix

Appendix A.1: Overview of attached CD-ROM

- This report
- “SESFederate”: NetBeans project folder, containing all java code
- Each case has its own folder. These contains all files associated with the cases:
    - Excel file, containing vessel properties
    - ShipX exchange files
    - Geometry files: *.geo (glview), *.dwg (AutoCAD), *.dxf (AutoCAD), *.mgf (ShipX)
    - Fan characteristic
    - Simvis files: simonline.svp (main file) and simvis.conf (configuration file) (only for case 1.1)
    - Maneuvering output file, HullInput.hsm
    - Process.xml and federation.xml (only for case 1.1)
- Folder with various videos from SimVis
- Resistance calculations, Excel file

Microsoft Excel files are stored in both Office 2007 and 2003 format

Appendix A.2: Solve spatial air cushion pressure by means of an element method

The following appendix sketches a simple element method to use to decide the spatially varying air cushion pressure. It was suggested by this author while working on a way to catch the acoustic cobblestone effects. As this work was abandoned (for the benefit of resistance calculations), the element models is not coded or verified in any ways.

The method is based upon theory for uniform air cushion pressure presented in chapter 2.6.2.

The air cushion is divided into 2D elements as shown in Figure 81. As only variation in longitudinal direction is important to catch pitch forces, the elements run from wet deck to water surface, from starboard to aport, giving no vertical or cross-directional pressure variations. This simplification might give too large elements. The air cushion pressure is assumed uniform inside each element.

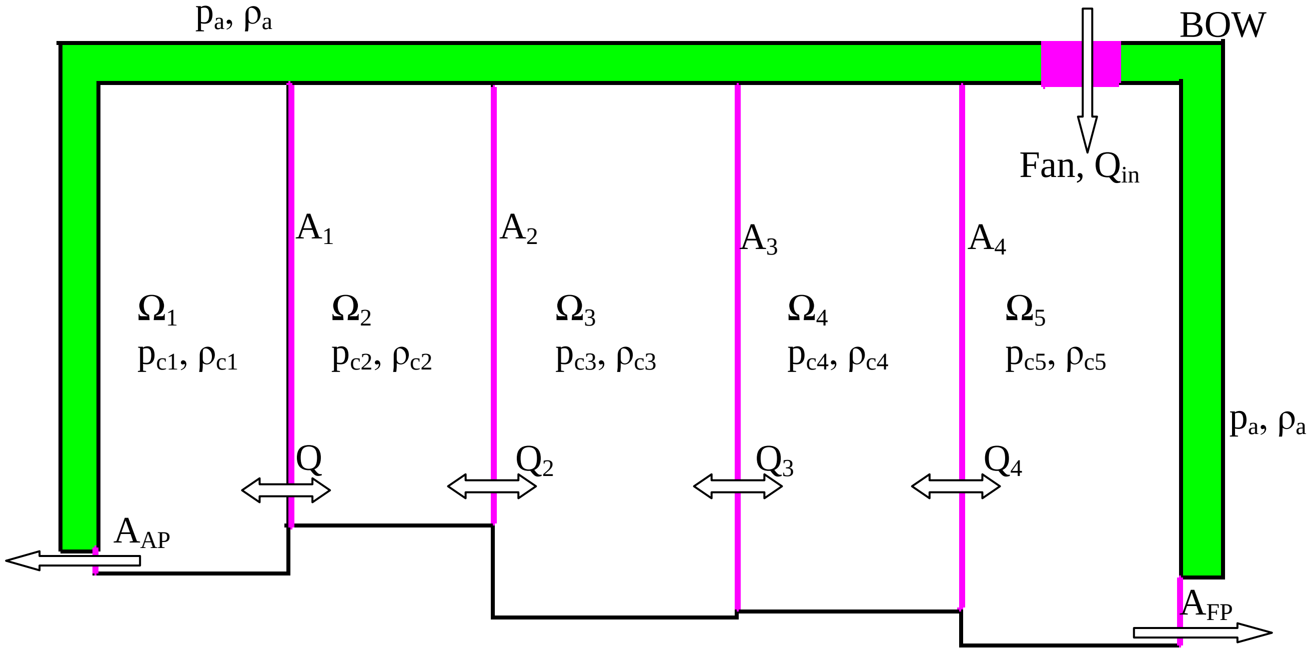


Figure 81: Schematic element model used to calculate spatial air cushion pressure

Fans are modeled by a point source. An x-coordinate must be specified to decide which element the fan's air enters. All air flows are calculated using the actual leakage area, A_i, and the air cushion pressure calculated previous time step. In the figure, A_{AP} and A_{FP} is leakage areas under respectively aft and fore seal.

Air flows are calculated using Bernoulli:

$$\frac{1}{2}\rho_i\left(\frac{Q_i}{c_n A_i}\right)^2 = \Delta p_i \qquad (.0)$$

$$Q_i = c_n A_i \sqrt{\frac{2\Delta p_i}{\rho_i}} \qquad (.1)$$

Each element must satisfy mass balance:

$$\sum_{in} \rho Q - \sum_{out} \rho Q = \frac{d}{dt}(\rho_c \Omega)_i = \frac{d\rho_{ci}}{dt}\Omega_\mathrm{i} + \rho_{ci}\frac{d\Omega_\mathrm{i}}{dt} \tag{.2}$$

The density in the air flows are taken in the area that the air flows from. The time dependent change in air cushion element is simply:

$$\left(\frac{d\Omega}{dt}\right)_n = \frac{\Omega_\mathrm{n} - \Omega_\mathrm{n-1}}{dt} \tag{.3}$$

The mass density is found using adiabatic relationship between air density and air pressure:

$$\frac{p}{p_0 + p_a} = \frac{p_a + p_c(t)}{p_0 + p_a} = \left(\frac{\rho_c}{\rho_a}\right)^\gamma \tag{.4}$$

Taylor expanding, this gives an expression for the mass density. This can be differentiated with respect to time.

$$\frac{\rho_c}{\rho_a} = \left(1 + \frac{\mu_u(t)p_0}{p_0 + p_a}\right)^{\frac{1}{\gamma}} \approx 1 + \frac{1}{\gamma}\frac{\mu_u(t)p_0}{p_0 + p_a} \tag{.5}$$

$$\frac{d\rho_c}{dt} = \frac{\rho_a}{\gamma}\frac{p_0}{p_0 + p_a}\frac{d\mu_\mathrm{u}}{dt} \tag{.6}$$

Equation for the mass density is solved using the air cushion pressure from previous time step when used in air flow calculations. When used in the final differential equation, current air cushion pressure is used.

Combine everything and solve differential equation for air cushion pressure.

$$\underbrace{\sum_{in} \rho Q - \sum_{out} \rho Q}_{\substack{A_L\, from\, current\, time\, step \\ \rho_c\, from\, last\, time\, step}} = \frac{\rho_a}{\gamma}\frac{p_0}{p_0 + p_a}\frac{d\mu_\mathrm{ui}}{dt}\Omega_\mathrm{i} + \left[\rho_a + \frac{\rho_a}{\gamma}\frac{\mu_{ui}(t)p_0}{p_0 + p_a}\right]\frac{d\Omega_\mathrm{i}}{dt} \tag{.7}$$

Appendix A.3: How to set up the simulator with the SES federate

Setting up a simulator model (with visualization) involves the following steps:

A. Make a vessel model. Decide vessel data as mass, radii of gyrations, LCG, air cushion data as air cushion pressure etc.
B. Import vessel model to ShipX. Feed vessel data to the different plug-ins needed for the simulator setup. Press “Start Simulator” and “Start Visualization”
C. Alter and add VeSim settings manually. That is: Add air cushion settings

A. Make vessel model and decide data

ShipX can take several different types of geometry files as input. Here AutoCad *.dxf files have been used. The files are on the attached CD-ROM. The geometry is defined by section curves and two longitudinal profile lines, one showing the shape of wet deck along CL, and one beneath the demi hull. Review the file for details.

B. ShipX

When file has been imported to ShipX, a VERES run is needed. Start VERES and press “Generate input to Ship Hydrodynamics for Vessel Simulator”. Choose max vessel speed in the simulator. Go to the “Edit input” tab, and then return to “Settings”. VERES has now been setup for a simulator run. However, these settings must be altered. Uncheck “Generate input to Ship Hydrodynamics for Vessel Simulator” and go back to “Edit input”.

On first page, choose “Multihull”. Press “Calculation Options” and check choose “Ignore mass balance check”. Now calculations will be done with actual vessel weight instead of displacement weight. Go back.

In condition information, add more wave periods and velocities if needed. Go back. In vessel description, add data for vessel.

You are now finished with the VERES settings. Press “Data Check” and control data. Then do the “Full Calculation”.

Now make a panel generation run with the vessel. This is straight forward. Some problem may occur though with the profile lines. The end of these must not have the same longitudinal coordinate as the end section (easily avoided by moving the profile lines 1[cm] in longitudinal direction).

Make a maneuvering run. This cannot be done using the SES model, as the plug-in is not designed for a catamaran shaped vessel. In this work, a dummy vessel was used. The dummy used for the MCMV is on the attached CD-ROM. It is important that the dummy vessel has the same lpp as the SES. The resistance curve must also be given as input to the maneuvering run. Under the tab “Resistance”, choose polynomial and type the different terms.

A “Ship Model for Vessel Simulator” run is now next. Choose the prepared runs and chose propulsors, sensors etc. In the cases here, the simple-controller is used. This can be chosen under “Miscellaneous”. Under Visualisation, check “Apply geometry file from Panel Generator”. Press “Generate Simulator Input File”.

Last step is to make a “Setup for the Vessel Simulator”. Choose wave conditions etc. If you chose “Regular long-crested waves” you need to alter the start positions for the vessel. Otherwise it will appear outside the visualized wave-field in SimVis.

Start the simulation and visualization.

B. Alter and add manually VeSim settings
The crossing from ShipX to VeSim is today date not made streamlined. Many manual settings must therefore be added.

First open the input folder to the “Setup for the Vessel Simulator”. All necessary settings are done here.

The two *.xml files defining the VeSim model need to be altered. This is further explained in appendix B.1 and B.2.

The maneuvering file needs to be altered. The file is placed in the “ShipModel.zip” file, and need to be altered there. As the maneuvering data are not part of this thesis work, most terms have been set zero. LCG and the mass should however be correct. These numbers are dimensionless. In the equations shown bellow, * means dimensionless.

$$LCG^* = \frac{\left(LCG - \frac{lpp}{2}\right)}{lpp} \tag{.8}$$

$$M^* = \frac{M}{0.5 \cdot \rho \cdot lpp^3} \tag{.9}$$

Last settings are for SimVis, file being in the “simvis” folder. The simonline.svp file does not have the be altered, but it is convenient to set initialViewPoint to the vessel’s initial position. The force vectors can be visualized by adding a “simvis.conf” file. The one that has been used is on the CD-ROM.

Appendix A.4: VERES data checks, case 1. Input to simulator and linear SES calculations

MARINTEK SINTEF	DATA-CHECK PROPERTIES	ENCL. (a)
		REPORT
		DATE 2008-06-09
		REF

Run name: input to simulator

```
Ship name: Case1
Loading condition description: Design waterline

ShipX exported data
Main dimensions (from input):
_______________________________________________________________

Length between perpendiculars    (m)     30.000
Breadth                          (m)     10.000
Draught, midship                 (m)      1.500
Sinkage                          (m)      0.000
Trim, + = aft                  (deg)      0.000
_______________________________________________________________

Number of hulls:   2  -> Catamaran

Coefficients for data check etc.:
Type                                  Specified    Calculated
_______________________________________________________________

Displacement                (tonnes)     279.00*        73.80
Vertical center of bouyancy,     KB                      0.750*
Vertical center of gravity,      VCG       3.000*
Longitudinal center of bouyancy, LCB                    15.000*
Longitudinal center of gravity,  LCG      15.000*       15.000
Longitudinal metacentric height, GMl                    47.858*
Transverse metacentric height,   GMt                    11.892*
Roll radius of gyration,         r44       3.500*
Pitch radius of gyration,        r55       7.500*
Yaw radius of gyration,          r66       7.500*
Roll-yaw radius of gyration,     r46       0.000*
_______________________________________________________________

* - Applied in the hydrodynamic calculations
```

ShipX - 06.06.2008 - 21:23:01 - Licensed to: Trygve E Halvorsen (NTNU)

MARINTEK SINTEF	DATA-CHECK PROPERTIES	ENCL. (b)
		REPORT
		DATE 2008-06-09
		REF

Run name: Linear SES calculations

```
Ship name: Case1
Loading condition description: Design waterline

ShipX exported data
Main dimensions (from input):
_______________________________________________________________

Length between perpendiculars    (m)     30.000
Breadth                          (m)     10.000
Draught, midship                 (m)      1.500
Sinkage                          (m)      0.000
Trim, + = aft                  (deg)      0.000
_______________________________________________________________

Number of hulls:   2  -> Catamaran

Data for air cushion no.  1:
Static cushion pressure                        (kPa)      8.000
Leakage                                     (m**3/s)    150.008
Inverse slope of fan characteristics                     -0.028
Volume of air cushion                         (m**3)    743.400
Surface area of air cushion                   (m**2)    252.000
Longitudinal volume center from AP               (m)     15.000
Bow seal type: Finger seal
Longitudinal position of bow seal                (m)     30.000
Stern seal type: Lobe bag
Excess membrane pressure                       (kPa)      1.200
Longitudinal position of stern seal              (m)      0.000
Ride control parameter                           ( )      0.000

Coefficients for data check etc.:
Type                                      Specified    Calculated
_______________________________________________________________

Displacement                 (tonnes)     279.00*         73.80
Lift from air cushion        (tonnes)                   205.503*
Vertical center of bouyancy,     KB                       0.750*
Vertical center of gravity,      VCG        3.000*
Longitudinal center of bouyancy, LCB                     15.000*
Longitudinal center of gravity,  LCG       15.000*       15.000
Longitudinal center of lift,     LCL                     15.000*
Longitudinal metacentric height, GMl                      9.864*
Transverse metacentric height,   GMt                      0.350*
Roll radius of gyration,         r44        3.500*
Pitch radius of gyration,        r55        7.500*
Yaw radius of gyration,          r66        7.500*
Roll-yaw radius of gyration,     r46        0.000*
_______________________________________________________________
 * - Applied in the hydrodynamic calculations
```

ShipX - 06.06.2008 - 21:19:48 - Licensed to: Trygve E Halvorsen (NTNU)

Appendix A.5: VERES data checks, case 1.1. Input to simulator and linear SES calculations

MARINTEK SINTEF	DATA-CHECK PROPERTIES	ENCL. (c)
		REPORT
		DATE 2008-06-09
		REF

Run name: input to simulator

```
Ship name: Case1.1
Loading condition description: Design waterline

ShipX exported data
Main dimensions (from input):
_______________________________________________________________

Length between perpendiculars    (m)      30.000
Breadth                          (m)      10.000
Draught, midship                 (m)       1.200
Sinkage                          (m)       0.000
Trim, + = aft                  (deg)       0.000
_______________________________________________________________

Number of hulls:   2  -> Catamaran

Coefficients for data check etc.:
Type                                 Specified    Calculated
_______________________________________________________________

Displacement                (tonnes)     140.20*        43.01
Vertical center of bouyancy,     KB                      0.791*
Vertical center of gravity,      VCG       3.000*
Longitudinal center of bouyancy, LCB                    15.000*
Longitudinal center of gravity,  LCG      15.000*       15.000
Longitudinal metacentric height, GMl                   105.276*
Transverse metacentric height,   GMt                    26.868*
Roll radius of gyration,         r44       3.500*
Pitch radius of gyration,        r55       7.500*
Yaw radius of gyration,          r66       7.500*
Roll-yaw radius of gyration,     r46       0.000*
_______________________________________________________________

* - Applied in the hydrodynamic calculations
```

ShipX - 06.06.2008 - 21:03:22 - Licensed to: Trygve E Halvorsen (NTNU)

MARINTEK SINTEF	DATA-CHECK PROPERTIES	ENCL. (d)
		REPORT
		DATE 2008-06-09
		REF

Run name: Linear SES calculations

```
Ship name: Case1.1
Loading condition description: Design waterline

ShipX exported data
Main dimensions (from input):
______________________________________________________________

Length between perpendiculars     (m)      30.000
Breadth                           (m)      10.000
Draught, midship                  (m)       1.200
Sinkage                           (m)       0.000
Trim, + = aft                   (deg)       0.000
______________________________________________________________

Number of hulls:   2  -> Catamaran

Data for air cushion no.  1:
Static cushion pressure                       (kPa)      4.000
Leakage                                    (m**3/s)     75.084
Inverse slope of fan characteristics                    -0.021
Volume of air cushion                        (m**3)    542.800
Surface area of air cushion                  (m**2)    240.000
Longitudinal volume center from AP              (m)     15.000
Bow seal type: Finger seal
Longitudinal position of bow seal               (m)     30.000
Stern seal type: Lobe bag
Excess membrane pressure                      (kPa)      0.600
Longitudinal position of stern seal             (m)      0.000
Ride control parameter                          ( )      0.000

Coefficients for data check etc.:
Type                                  Specified    Calculated
______________________________________________________________

Displacement                  (tonnes)     140.20*        43.01
Lift from air cushion         (tonnes)                   97.859*
Vertical center of bouyancy,      KB                      0.790*
Vertical center of gravity,       VCG       3.000*
Longitudinal center of bouyancy,  LCB                    15.000*
Longitudinal center of gravity,   LCG      15.000*       15.000
Longitudinal center of lift,      LCL                    15.000*
Longitudinal metacentric height,  GMl                    29.503*
Transverse metacentric height,    GMt                     5.451*
Roll radius of gyration,          r44       3.500*
Pitch radius of gyration,         r55       7.500*
Yaw radius of gyration,           r66       7.500*
Roll-yaw radius of gyration,      r46       0.000*
______________________________________________________________
 * - Applied in the hydrodynamic calculations
```

ShipX - 06.06.2008 - 21:03:59 - Licensed to: Trygve E Halvorsen (NTNU)

Appendix A.6: VERES data checks, case 2. Input to simulator and linear SES calculations

MARINTEK SINTEF	DATA-CHECK PROPERTIES	ENCL. (e)
		REPORT
		DATE 2008-06-09
		REF

Run name: input to simulator

```
Ship name: Case2
Loading condition description: Design waterline

ShipX exported data
Main dimensions (from input):
______________________________________________________________

Length between perpendiculars      (m)      30.000
Breadth                            (m)      10.000
Draught, midship                   (m)       1.500
Sinkage                            (m)       0.000
Trim, + = aft                    (deg)       0.000
______________________________________________________________

Number of hulls:   2  -> Catamaran

Coefficients for data check etc.:
Type                                     Specified    Calculated
______________________________________________________________

Displacement                  (tonnes)     277.00*        77.64
Vertical center of bouyancy,      KB                       0.896*
Vertical center of gravity,       VCG        3.000*
Longitudinal center of bouyancy, LCB                      15.000*
Longitudinal center of gravity,  LCG       15.000*        15.000
Longitudinal metacentric height, GMl                      68.697*
Transverse metacentric height,   GMt                      16.288*
Roll radius of gyration,         r44        3.500*
Pitch radius of gyration,        r55        7.500*
Yaw radius of gyration,          r66        7.500*
Roll-yaw radius of gyration,     r46        0.000*
______________________________________________________________

* - Applied in the hydrodynamic calculations
```

ShipX - 04.06.2008 - 21:19:29 - Licensed to: Trygve E Halvorsen (NTNU)

MARINTEK SINTEF	DATA-CHECK PROPERTIES	ENCL. (f)
		REPORT
		DATE 2008-06-09
		REF

Run name: linear SES calculation

```
Ship name: Case2
Loading condition description: Design waterline

ShipX exported data
Main dimensions (from input):
_______________________________________________________________

Length between perpendiculars    (m)     30.000
Breadth                          (m)     10.000
Draught, midship                 (m)      1.500
Sinkage                          (m)      0.000
Trim, + = aft                  (deg)      0.000
_______________________________________________________________

Number of hulls:   2  -> Catamaran

Data for air cushion no.  1:
Static cushion pressure                        (kPa)      8.000
Leakage                                     (m**3/s)    150.008
Inverse slope of fan characteristics                     -0.028
Volume of air cushion                         (m**3)    568.672
Surface area of air cushion                   (m**2)    228.649
Longitudinal volume center from AP               (m)     15.000
Bow seal type: Finger seal
Longitudinal position of bow seal                (m)     30.000
Stern seal type: Lobe bag
Excess membrane pressure                       (kPa)      1.200
Longitudinal position of stern seal              (m)      0.000
Ride control parameter                           ( )      0.000

Coefficients for data check etc.:
Type                                      Specified    Calculated
_______________________________________________________________

Displacement                   (tonnes)     277.00*        77.64
Lift from air cushion          (tonnes)                   186.461*
Vertical center of bouyancy,      KB                        0.896*
Vertical center of gravity,       VCG      3.000*
Longitudinal center of bouyancy,  LCB                      15.000*
Longitudinal center of gravity,   LCG     15.000*          15.000
Longitudinal center of lift,      LCL                      15.000*
Longitudinal metacentric height,  GMl                      16.700*
Transverse metacentric height,    GMt                       2.010*
Roll radius of gyration,          r44      3.500*
Pitch radius of gyration,         r55      7.500*
Yaw radius of gyration,           r66      7.500*
Roll-yaw radius of gyration,      r46      0.000*
_______________________________________________________________
* - Applied in the hydrodynamic calculations
```

ShipX - 04.06.2008 - 21:18:18 - Licensed to: Trygve E Halvorsen (NTNU)

Appendix B.1: process.xml

process.xml governs the startup and termination of the simulation. It tells which federates to start (federates can also be started manually later), and can also start optional postprocessors after termination. Each federate has some lines starting with “processBuilder”, giving initiating arguments to the federate. The example given bellow is for case 1.

```
<processBuilder command="java" name="aircushion">
        <args>-cp</args>
        <args>..\lib\vesim.jar;..\lib\toolsmk3.jar;..\lib\csi.jar;..\lib\jdom.jar;D:\Skole\
        Java\SESFederate\dist\SESFederate.jar</args>
        <args>-Xincgc</args>
        <args>no.marintek.simulator.ses.SESFederate</args>
        <args>D:\\Skole\\Diplom\\ShipX\\root\\Fle71D3F37A\\XShip4D3E66D0\\Loa
        CE7F6599\\runs\\Run545550D8\\input\\federation.xml
        </args>
        <args>aircushion</args>
        <args>csi</args>
</processBuilder>
```

The lines above must be copied into the process.xml file. In addition the following line must be copied into the end of the file:

```
<process name="aircushion" client="localhost" wait="500" />
```

The last line here tells the simulator to start the aircushion process. “Wait” is time in milliseconds the simulator waits before it proceeds to the next federate in the “start processes” list.

If the forced position tester is to be used, this must also be added manually. The lines are:

```
<processBuilder command="java" name="forcedpositiontester">
        <args>-cp</args>
        <args>..\lib\vesim.jar;..\lib\toolsmk3.jar;..\lib\csi.jar;..\lib\jdom.jar
        </args>
        <args>-Xincgc</args>
        <args>no.marintek.simulator.controller.ForcedPositionTester</args>
        <args>D:\\Skole\\Diplom\\ShipX\\root\\Fle71D3F37A\\XShip4D3E66D0\\Loa
        CE7F6599\\runs\\Run545550D8\\input\\federation.xml
        </args>
        <args>forcedpositiontester</args>
        <args>csi</args>
</processBuilder>

<process name="forcedpositiontester" client="localhost" wait="500" />
```

No other changes are needed in process.xml.

Appendix B.2: federation.xml

federation.xml states what parameters and attributes each federate uses. It also tells how the federates interact with each other. Fairly many changes are needed in this file.

The vessel federate needs to be told to subscribe to the air cushion federate. The subscribe lines are:

```
<subscribe name="Case1.AirCushion" type="AirCushion">
    <attribute name="force_surge"/>
    <attribute name="force_sway"/>
    <attribute name="force_heave"/>
    <attribute name="moment_roll"/>
    <attribute name="moment_pitch"/>
    <attribute name="moment_yaw"/>
    <attribute name="x_rel_ap"/>
    <attribute name="y_rel_cl"/>
    <attribute name="z_rel_bl"/>
</subscribe>
```

One more alteration is needed for the vessel to subscribe correct. Under “model properties”, “AirCushion” must be added to the list of “forceclasses”.

The air cushion federate must be added as a whole. The lines are:

```
<!-- START FEDERATE -->
<federate name="aircushion" lookahead="0.05" simbus="csi">
  <!-- SUBSCRIPTIONS -->
  <subscribe name="Case1_1.Vessel" type="Vessel">
  <attribute name="cg_x_rel_ap" />
  <attribute name="cg_y_rel_cl" />
  <attribute name="cg_z_rel_bl" />
  <attribute name="lpp" />
  <attribute name="cg_north" />
  <attribute name="cg_east" />
  <attribute name="cg_down" />
  <attribute name="cg_surge_vel" />
  <attribute name="cg_sway_vel" />
  <attribute name="cg_heave_vel" />
  <attribute name="cg_surge_acc" />
  <attribute name="cg_sway_acc" />
  <attribute name="cg_heave_acc" />
  <attribute name="roll" />
  <attribute name="roll_vel" />
  <attribute name="roll_acc" />
  <attribute name="pitch" />
  <attribute name="pitch_vel" />
  <attribute name="pitch_acc" />
  <attribute name="yaw" />
  <attribute name="yaw_vel" />
  <attribute name="yaw_acc" />
  <attribute name="loaded_draught_at_amidships" type="double" numdec="2" />
  <attribute name="cur_surge_vel" />
    </subscribe>
  <subscribe name="Case1_1.Wave" type="Environment">
  <attribute name="length" />
  <attribute name="n_active_HF" />
  <attribute name="n_active_LF" />
  <attribute name="n_fft" />
  <attribute name="peak_period" />
  <attribute name="peakedness" />
  <attribute name="seastate_changed" />
```

```
<attribute name="seastate_ramp_time" />
<attribute name="spectrum" />
<attribute name="spreading_angle" />
<attribute name="swell_direction" />
<attribute name="tile" />
<attribute name="wave_direction" />
<attribute name="wave_height" />
<attribute name="tsw_file" type="String" />
<attribute name="cb_wave_vel_x" />
<attribute name="cb_wave_vel_y" />
<attribute name="cb_wave_vel_z" />
    </subscribe>
<subscribe name="SimulationTime" type="Time">
<attribute name="time" />
    </subscribe>
     <!-- REGISTRATION -->
<register name="Case1_1.AirCushion" type="AirCushion" parent="Case1_1.Vessel">
  <!-- ATTRIBUTES -->
<attribute name="force_surge" description="Total Air Cushion Federate surge force" unit="N"
     type="double" value="0.0" numdec="0" logable="true" />
<attribute name="force_sway" description="Total Air Cushion Federate sway force" unit="N" type="double"
     value="0.0" numdec="0" logable="true" />
<attribute name="force_heave" description="Total Air Cushion Federate heave force" unit="N"
     type="double" value="0.0" numdec="0" logable="true" />
<attribute name="moment_roll" description="Total Air Cushion Federate roll moment" unit="Nm"
     type="double" value="0.0" numdec="0" logable="true" />
<attribute name="moment_pitch" description="Total Air Cushion Federate pitch moment" unit="Nm"
     type="double" value="0.0" numdec="0" logable="true" />
<attribute name="moment_yaw" description="Total Air Cushion Federate yaw moment" unit="Nm"
     type="double" value="0.0" numdec="0" logable="true" />
<attribute name="x_rel_ap" description="Point of attack rel AP x-dir" unit="m" type="double" value="0.0"
     numdec="2" logable="false" />
<attribute name="y_rel_cl" description="Point of attack rel CL y-dir" unit="m" type="double" value="0.0"
     numdec="2" logable="false" />
<attribute name="z_rel_bl" description="Point of attack rel BL z-dir" unit="m" type="double" value="0.0"
     numdec="2" logable="false" />
<attribute name="cushion_force_surge" description="Air Cushion surge force" unit="N" type="double"
     value="0.0" numdec="0" logable="true" />
<attribute name="cushion_force_sway" description="Air Cushion sway force" unit="N" type="double"
     value="0.0" numdec="0" logable="true" />
<attribute name="cushion_force_heave" description="Air Cushion heave force" unit="N" type="double"
     value="0.0" numdec="0" logable="true" />
<attribute name="cushion_moment_roll" description="Air Cushion roll moment" unit="Nm" type="double"
     value="0.0" numdec="0" logable="true" />
<attribute name="cushion_moment_pitch" description="Air Cushion pitch moment" unit="Nm"
     type="double" value="0.0" numdec="0" logable="true" />
<attribute name="cushion_moment_yaw" description="Air Cushion yaw moment" unit="Nm" type="double"
     value="0.0" numdec="0" logable="true" />
<attribute name="viscous_force_surge" description="Total viscous surge force" unit="N" type="double"
     value="0.0" numdec="0" logable="true" />
<attribute name="viscous_calm_force_surge" description="Viscous surge force calm water" unit="N"
     type="double" value="0.0" numdec="0" logable="true" />
<attribute name="viscous_added_force_surge" description="Added viscous surge force rel calm water"
     unit="N" type="double" value="0.0" numdec="0" logable="true" />
<attribute name="viscous_moment_pitch" description="Total viscous pitch moment" unit="Nm"
     type="double" value="0.0" numdec="0" logable="true" />
<attribute name="viscous_calm_moment_pitch" description="Viscous pitch moment calm water"
     unit="Nm" type="double" value="0.0" numdec="0" logable="true" />
<attribute name="viscous_added_moment_pitch" description="Added viscous pitch moment rel calm
     water" unit="Nm" type="double" value="0.0" numdec="0" logable="true" />
<attribute name="p_c" description="Overpressure inside air cushion" unit="Pa" type="double" value="0.0"
     numdec="1" logable="true" />
<attribute name="cushion_volume" description="Enclosed air cushion volume" unit="m3" type="double"
     value="0.0" numdec="3" logable="true" />
<attribute name="wet_surface" description="Wet surface" unit="m2" type="double" value="0.0"
     numdec="1" logable="true" />
<attribute name="wet_surface_0" description="Static wet surface" unit="m2" type="double" value="0.0"
     numdec="1" logable="true" />
```

```xml
<attribute name="cushion_volume_0" description="Static enclosed air cushion volume" unit="m3"
    type="double" value="0.0" numdec="3" logable="true" />
<attribute name="Q_0" description="Static air flow" unit="m3/s" type="double" value="0.0" numdec="3"
    logable="true" />
<attribute name="Q_in" description="Air flow due to fans" unit="m3/s" type="double" value="0.0"
    numdec="3" logable="true" />
<attribute name="Q_out" description="Leakage from air cushion" unit="m3/s" type="double" value="0.0"
    numdec="3" logable="true" />
<attribute name="rho_a" description="Mass density of air" unit="kg/m3" type="double" value="1.23"
    numdec="2" logable="true" />
<attribute name="gamma" description="Adiabatic constant" unit="" type="double" value="1.4" numdec="1"
    logable="true" />
<attribute name="p_a" description="Atmospheric pressure" unit="Pa" type="double" value="101325"
    numdec="0" logable="true" />
<attribute name="c_n" description="Contraction of air flow" unit="" type="double" value="0.65" numdec="2"
    logable="true" />
<attribute name="H" description="Roughness factor" unit="E-6m" type="double" value="75.0" numdec="0"
    logable="true" />
<attribute name="k" description="Form factor" unit="" type="double" value="0.0" numdec="1"
    logable="true"/>
<attribute name="v" description="Kinematic viscosity" unit="m2/s" type="double" value="0.0000012"
    numdec="7" logable="true" />
<attribute name="lift_ratio_0" description="Static air cushion lift/total mass ratio" unit="%"
    type="double" value="0.0" numdec="1" logable="true" />
<attribute name="lift_ratio" description="Air cushion lift/total mass ratio" unit="%" type="double"
    value="0.0" numdec="1" logable="true" />
<attribute name="seal_stern_x_rel_ap" description="Longitudinal position of stern seal rel AP" unit="m"
    type="double" value="0.0" numdec="2" logable="true" />
<attribute name="seal_stern_z_rel_bl" description="Vertical position of bottom of stern seal rel BL"
    unit="m" type="double" value="0.15" numdec="2" logable="true"/>
<attribute name="seal_bow_x_rel_ap" description="Longitudinal position of bow seal rel AP" unit="m"
    type="double" value="30.0" numdec="2" logable="true" />
<attribute name="seal_bow_z_rel_bl" description="Vertical position of bottom of bow seal rel BL"
    unit="m" type="double" value="0.10" numdec="2" logable="true"/>
    <!-- PARAMETERS -->
<parameter name="cushion_active" description="Air cushion active" unit="" type="boolean" value="true"
    logable="true" />
<parameter name="update_air_cushion" description="Reset/update air cushion based on current
    settings" type="boolean" value="false" logable="true" />
<parameter name="p_0" description="Static over pressure" unit="Pa" type="double" value="4000.0"
    numdec="0" logable="true" />
<parameter name="on_cushion" description="Condition on cushion. False: Off cushion" unit=""
    type="boolean" value="true" logable="true" />
<parameter name="forces_p_c" description="Calculate air cushion forces using p_c (false: use p_0)"
    unit="" type="boolean" value="true" logable="true" />
<parameter name="viscous_forces" description="Include added viscous resistance" unit="" type="boolean"
    value="true" logable="true" />
<parameter name="fan_and_leak_active" description="Calculate air flow in/out from air cushion" unit=""
    type="boolean" value="true" logable="true" />
<parameter name="fan_number" description="Number of fans active" unit="" type="integer" value="2"
    numdec="0" logable="true" />
   </register>
 <!-- MODEL PROPERTIES -->
<model
    unzip_folder="D:\Skole\Diplom\ShipX\root\Fle71D3F37A\XShip4D3E66D0\LoaCE7F6599\runs\Ru
    n545550D8\input\temp" mgf_path="D:\Skole\Java\case1_1.mgf"
    fan_path="D:\Skole\Java\case1_1.fan"/>
   </federate>
 <!-- END FEDERATE -->
```

Some data in the federate list is vessel specific, and must be altered. These are:

- Water, air and hull properties: Mass densities, adiabatic constant, atmospheric pressure, contraction of air flow, hull roughness factor, form factor and kinematic viscosity (for the water)
- Positions of bow and stern seals, both vertical and longitudinal

- Everywhere Case1 appears, this must be altered to name of actual simulator run
- Name of parent, this is the name of the vessel federate
- unzip_folder, the path to where the ship model archive is unzipped. This path is given several other places in the federation.xml file and can be copied from here
- *.mgf, path to the file containing the *.mgf geometry file
- *.fan, path to the file containing the *.fan fan characteristic

If the simple heading controller is to be used as propulsion, this will probably have to be altered. The max force from this is decided by three attribute values giving the maximum acceleration. These values must probably be increased in order to reach high enough vessel speed.

If the vessel position tester is to be used, all federation data must be added, as with the air cushion federate. These lines are:

```xml
  <!-- START FEDERATE -->
<federate name="forcedpositiontester" lookahead="0.05" simbus="csi">
  <!--  SUBSCRIPTIONS -->
  <!--  REGISTRATION -->
<register name="ForcedPositionTester" type="VesselMotionSolver">
   <!--  ATTRIBUTES -->
  <attribute name="cg_north" description="NED north position of CG" unit="m" type="double" value="0.0"
      numdec="3" logable="true" />
  <attribute name="cg_east" description="NED east position of CG" unit="m" type="double" value="0.0"
      numdec="3" logable="true" />
  <attribute name="cg_down" description="NED down position of CG" unit="m" type="double" value="0.0"
      numdec="3" logable="true" />
  <attribute name="cg_surge_vel" description="Surge velocity of CG" unit="m/s" type="double" value="0.0"
      numdec="3" logable="true" />
  <attribute name="cg_sway_vel" description="Sway velocity of CG" unit="m/s" type="double" value="0.0"
      numdec="3" logable="true" />
  <attribute name="cg_heave_vel" description="Heave velocity of CG" unit="m/s" type="double" value="0.0"
      numdec="3" logable="true" />
  <attribute name="cg_surge_acc" description="Surge acceleration of CG" unit="m/s^2" type="double"
      value="0.0" numdec="4" logable="true" />
  <attribute name="cg_sway_acc" description="Sway acceleration of CG" unit="m/s^2" type="double"
      value="0.0" numdec="4" logable="true" />
  <attribute name="cg_heave_acc" description="Heave acceleration of CG" unit="m/s^2" type="double"
      value="0.0" numdec="4" logable="true" />
  <attribute name="roll" description="Roll angle of vessel" unit="deg" type="double" value="0.0" numdec="3"
      logable="true" />
  <attribute name="roll_vel" description="Roll velocity of vessel" unit="deg/s" type="double" value="0.0"
      numdec="3" logable="true" />
  <attribute name="roll_acc" description="Roll acceleration of vessel" unit="deg/s^2" type="double"
      value="0.0" numdec="4" logable="true" />
  <attribute name="pitch" description="Pitch angle of vessel" unit="deg" type="double" value="0.0"
      numdec="3" logable="true" />
  <attribute name="pitch_vel" description="Pitch velocity of vessel" unit="deg/s" type="double" value="0.0"
      numdec="3" logable="true" />
  <attribute name="pitch_acc" description="Pitch acceleration of vessel" unit="deg/s^2" type="double"
      value="0.0" numdec="4" logable="true" />
  <attribute name="yaw" description="Yaw angle of vessel" unit="deg" type="double" value="0.0"
      numdec="3" logable="true" />
  <attribute name="yaw_vel" description="Yaw velocity of vessel" unit="deg/s" type="double" value="0.0"
      numdec="3" logable="true" />
  <attribute name="yaw_acc" description="Yaw acceleration of vessel" unit="deg/s^2" type="double"
      value="0.0" numdec="4" logable="true" />
   <!--  PARAMETERS -->
  <parameter name="cg_north" description="NED north position of CG" unit="m" type="double"
      value="640.0" numdec="3" logable="true" />
  <parameter name="cg_east" description="NED east position of CG" unit="m" type="double" value="640.0"
      numdec="3" logable="true" />
```

```xml
<parameter name="cg_down" description="NED down position of CG" unit="m" type="double" value="-
    3.15" numdec="3" max="50.0" min="-50.0" logable="true" />
<parameter name="roll" description="Roll angle of vessel" unit="deg" type="double" value="0.0"
    numdec="3" max="90.0" min="-90.0" logable="true" />
<parameter name="pitch" description="Pitch angle of vessel" unit="deg" type="double" value="0.0"
    numdec="3" max="90.0" min="-90.0" logable="true" />
<parameter name="yaw" description="Yaw angle of vessel" unit="deg" type="double" value="0.0"
    numdec="3" max="360.0" min="0.0" logable="true" />
<parameter name="cg_surge_vel" description="Surge velocity of CG" unit="m/s" type="double" value="0.0"
    numdec="3" logable="true" />
<parameter name="cg_sway_vel" description="Sway velocity of CG" unit="m/s" type="double" value="0.0"
    numdec="3" logable="true" />
<parameter name="cg_heave_vel" description="Heave velocity of CG" unit="m/s" type="double"
    value="0.0" numdec="3" logable="true" />
<parameter name="roll_vel" description="Roll velocity of vessel" unit="deg/s" type="double" value="0.0"
    numdec="3" logable="true" />
<parameter name="pitch_vel" description="Pitch velocity of vessel" unit="deg/s" type="double"
    value="0.0" numdec="3" logable="true" />
<parameter name="yaw_vel" description="Yaw velocity of vessel" unit="deg/s" type="double" value="0.0"
    numdec="3" logable="true" />
  </register>
  </federate>
 <!-- END FEDERATE -->
```

Nothing in the text for forced position tester needs to be altered.

Appendix B.3: *.mgf file

The *.mgf file contains the geometry of the vessel. They file type is the same as used internally in ShipX. It contains the lpp of the vessel and the geometry of the hull section wise. The coordinates is for the starboard side of the vessel, meaning that y-direction is positive in starboard direction.

Excerpt from example file is given below (case 1). Geometry of the first two sections is given.

```
******
* Four first lines are reserved for user
* Can contain any information
******                                   lpp
30.0
32                                       Section
-15.00000191                             nr
6                                  Number of
0.00000000   6.00000048            coordinates in
5.00000000   6.00000048            section            Data for section 32
5.00000000   0.00000000
4.20000029   0.00000000
4.20000029   4.50000000
0.00000000   4.50000000
31
-14.50000191                      x-coordinate
6                                 of section
0.00000000   6.00000048
5.00000000   6.00000048                               Data for section 31
5.00000000   0.00000000
4.20000029   0.00000000
4.20000029   4.50000000
0.00000000   4.50000000
0.00000000   4.50000000
…
…                                      Pairs of y- and
…                                      z-coordinates
```

Appendix B.4: *.fan

*.fan files contain the fan characteristics. As an example, the fan characteristic used in case 1 and 2 is given here.

```
'Fan characteristic curve
'T-craft
'
'Number of points on curve:
2
'Q0_fan [m^3/s], P_fan [mmWc] (values for a single fan)
0.      1323.7
376.    0.
```

The four first lines are only comments. The fifth line is the number of points on curve (here two points, a linear characteristic). The sixth line is a comment, and the rest is operation points. First number is air flow in m^3/s and the second is air pressure in mm water column.

NetBeans IDE 6.0.1 has been used as editor. The compiler was Sun JDK (Java Development Kit) v1.6, which is integrated in NetBeans.

The java code is built up by several java classes. Classes are run as instants in the program execution. These instants are called objects. This means that a class can have several instants running at the same time as objects. E.g. the seal class is used to make two objects, one bow seal and one stern seal. Also all fan characteristics are included in separate fan objects.

The following appendixes explain briefly the build up and function of the different classes. It first list the main assumptions carried out in that class. Then it describes the input given to the class and the out data it provides. Finally it briefly explains how the class operates, its principal and how it is configured. For more in depth knowledge, the code must be reviewed. All code is placed on the attached CD-ROM and is filled with comments.

In addition to what code is shown here, Marintek developed code is used. This code is collected in libraries or packages. Each package contains one or more classes, and the code itself cannot be viewed. Dariusz Fathi [FATHI 08] has helped out a lot telling what classes and packages that can be utilized for what. Some conversion between axis system and decomposing of forces are for instance done using Marintek packages. Also the global position of hull and wave elevation is obtained using Marintek packages.

The air cushion federate is also collected in a package, called SESFederate.

Address for the air cushion federate package is:
no.marintek.simulator.ses

Addresses for Marintek packages used are:
no.marintek.csi.datadesk.IObjectStorage
no.marintek.simulator.environment.WavesModel
no.marintek.simulator.generic.GenericFederate
no.marintek.simulator.vessel.ShipMotionConversion
no.marintek.tools.dataobjects.Motion
no.marintek.tools.dataobjects.Ned
no.marintek.tools.math.CoordinateConversions
no.marintek.tools.dataobjects.Linear

Addresses for common Java packages used are:
java.io.*
java.util.*

Appendix C.1: SESFederate

The SESFederate class is the main class of the air cushion federate. It is started by the VeSim process.xml file. It is the federate that communicates with the CSI server, sending and receiving attributes and parameters.

The main goal of the SESFederate is to provide the forces that the air cushion inflicts on the vessel, as well as added viscous resistance.

Assumptions

Uniform air cushion pressure

Enclosed air cushion volume and leakages calculated using p_0. All other calculations needing air cushion pressure use p_c

Point of attack of added viscous resistance: (x,y,z) = (Lpp/2, 0, Static draught/2) relative AP, CL, BL

When the SESFederate is inactive, the vessel floats at user-given draught. This equals static air cushion forces

In data

Time step and running time

The input files specified in appendix B

Wave elevation

Global position of vessel

Parameters from the CSI server:

Name	*Unit*	*Description*
cushion_active	-	Air cushion active. True or false
on_cushion	-	True: Calculate air cushion forces from the air cushion. False: Off cushion
fan_and_leak_active	-	Calculate air flow in/out from air cushion. True or false
fan_number	-	Number of active fans
forces_p_c	-	Calculate air cushion forces using p_c (false: use p_0). True or false
p_0	Pa	Static over pressure
update_air_cushion	-	Reset/update air cushion based on current settings. True or false
viscous_forces	-	Include added viscous resistance. True or false

Attributes from the federation.xml file:

Name	*Unit*	*Description*
c_n	-	Contraction of air flow out from the air cushion
H	μm	Hull roughness factor
k	-	Hull form factor
p_a	Pa	Atmospheric pressure
seal_bow_x_rel_ap	m	Longitudinal position of bow seal relative aft perpendicular
seal_bow_z_rel_bl	m	Vertical position of bow seal relative baseline
seal_stern_x_rel_ap	m	Longitudinal position of stern seal relative aft perpendicular
seal_stern_z_rel_bl	m	Vertical position of stern seal relative aft baseline
rho_a	kg/m^3	Mass density of air at atmospheric pressure
v	m^2/s	Kinematic viscosity

Out data

Attributes to the CSI server:

Name	*Unit*	*Description*
Q_0	m^3/s	Static air flow from fans
Q_in	m^3/s	Air flow from fans

Q_out	m^3/s	Leakages from air cushion
cushion_force_*	N	Force due to air cushion over pressure
cushion_moment_*	Nm	Moment due to air cushion over pressure
p_c	Pa	Air cushion overpressure
cushion_volume	m^3	Enclosed air cushion volume
cushion_volume_0	m^3	Static enclosed air cushion volume
force_*	Pa	Total force from the air cushion federate
lift_ratio	-	Air cushion lift/total mass. Ratio
lift_ratio_0	-	Static air cushion lift/total mass. Ratio
moment_*	Nm	Total moment from the air cushion federate
viscous_added_force_surge	N	Added viscous surge force
viscous_added_moment_pitch	Nm	Added viscous pitch moment
viscous_calm_force_surge	N	Viscous surge force for calm water
viscous_calm_moment_pitch	Nm	Viscous pitch moment for calm water
viscous_force_surge	N	Total viscous surge force
viscous_moment_pitch	Nm	Total viscous pitch moment
wet_surface	m^2	Wet surface
wet_surface_0	m^2	Static wet surface
x_rel_ap	m	Point of attack (of force) relative AP x-dir
y_rel_cl	m	Point of attack (of force) relative CL y-dir
z_rel_bl	m	Point of attack (of force) relative BL z-dir

Manner of operation and configuration

When started up, the SESFederate reads all parameters. These decide what kind of calculations the SESFederate should do, if any at all. Next step is to check wave attributes. The code for this was supplied by Dariusz Fathi [FATHI 08]. No focus will be given to this, but it means that the SESFederate initiate its own wave data. The wave settings are controlled for each time step.

If the air_cushion_active parameter is set to “true”, the SESFederate will run its main method, called “calculateAirCushionForces”. If the parameter is set to “false”, the federate will simply set all forces to zero.

The federation.xml file is the main input files. This gives input data as attributes. It also gives the file path to the three other input files the federate uses. These are the geometry file, the fan characteristic and one file telling the displacement mass of the vessel. The objects initialized now are: constants, fanCharacteristic, airCushion, stern and bow Seal, SESHull and SESMass. All these classes are reinitialized if the “update_air_cushion” parameter is set to true. This includes rereading all input files. This is done so the user easily can alter all settings without having to restart the simulator.

Next step is now to obtain new motion characteristics of center of gravity. This information is used to find global coordinates.

If the federate object has just been started, or update_air_cushion has been set to true, static and initial calculations will be carried out. This includes making an initalSESData object. Static calculations are used to make initial values for the pressure calculations.

Next step for the federate is to find global coordinates for the air cushion and hull geometry. This is done by the airCushion and SESHull objects. The airCushion object is then used by the cushionVolume to calculate enclosed air cushion volume. wetHull uses both the airCushion and the SESHull object to calculate wet surface. Next steps depend on what calculations method has been chosen: Leakages and fan included or excluded.

If the leakages and fans are not included, the air cushion pressure is calculated by the equation given in chapter 2.6.1 If they are included, air cushion pressure is calculated using the equations in chapter 2.6.2. This involves using the cushionLeakage class to calculate leakages from the air cushion and the fanCharacteristic object to get air flow from the fan(s). The leakage and volume calculations use both static air cushion pressure p_0 as input. All other calculations needing air cushion pressure use the air cushion pressure from last time step. p_0 is used while using pc allows for very unphysical situation, with e.g. a water column raising many meters above the mean water level.

Solving the differential equation gives new air cushion pressure. The integration variables A, B and p in equations 3.35-3.37 has initial values calculated using values from initalSESData. After that, they use values from last time step.

Having obtained air cushion pressure, forces can be calculated using the SESForces class. This depends on the "on_cushion" parameter. The SES is assumed on cushion when no forces are fed from this federate, meaning that the SES federate supply dynamic forces. This is defined as instantaneous forces subtracted static forces (from the initalSESData object). If "on_cushion" is "true", that is how forces are calculated. Dynamic forces are given as output. If it is set to "false", forces from the SES federate equals negative static forces, rendering the vessel off cushion. If the restoring forces are linear, this might give a small surge force if the vessel obtains trim (the negative heave force from the SES federate gets a surge component).

If "forces_p_c" is set to "false", instantaneous air cushion forces will be calculated using p_0. Lift ratio is calculated using instantaneous heave force and mass from SESMass object.

When "viscous_forces" is set "true", added viscous resistance will be calculated. This is calculated using the equations in chapter 3. The velocity is defined as relative velocity between the vessel and the wave particle at center of gravity. This means that wave particle velocity and current velocity is taken into account. Point of attack of added viscous resistance is (x,y,z) = (Lpp/2, 0, Static draught/2) relative AP, CL, BL. As forces are given relative center of the air cushion axis system, the added viscous resistance needs to be decomposed.

Last step of the air cushion federate run is to set attributes. These are data that is sent to the CSI server. Finally the air cushion pressure and the air cushion volume are stored until the calculations at the next time step.

The federate repeats the operations for each time step. The time step is set in the federation.xml file, and should be rather low. In this thesis 0.05 [s] has been used. This is the lowest possible time step (same as the vessel federate), and is chosen because the air cushion forces are the main force contributor (in case 1.1, 70% of the lift is supplied by the air cushion). Results seem to suggest that this is a reasonable time step.

Appendix C.2: constants

Contains the different air, water and hull attributes used in the classes. This class is used to unite all attributes in the same object instead of using e.g. separate variables.

In data

The constants. These are collected from the CSI server as air cushion federate attributes.

Out data

rho, mass density of sea water
g, acceleration of gravity
rho_a, mass density of air at atmospheric pressure
gamma, adiabatic constant
p_a, atmospheric pressure
c_n, contraction of air flow
H, hull roughness factor
υ, kinematic viscosity
k, form factor

Manner of operation and configuration

The class gets the various air, water and hull attributes as input. By including the constants object when initializing a new object, all necessary wave, water and hull attributes are imported.

Appendix C.3: SESHull

This class stores all geometric data from the *.mgf file. This includes the vessel geometry (local and global coordinate system and element information) and the positions where the different hull sections cross the water surface on the outside hull side (see Figure 84).

In data

Hull geometry, *.mgf file
Draught, needed to calculate static conditions
Global position of vessel
Wave elevation
Simulation time

Out data

Lpp, length of vessel between perpendiculars
Vessel geometry (in local or global coordinates) in node coordinate and element information
Crossing points between water and the hull, outside the cushion

Manner of operation and configuration

The class reads hull geometry from the .mgf file. The data includes length between perpendiculars and section data. Section data includes x coordinate of the section as well as several points (given by y and z coordinates) describing the section. Due to symmetry, the coordinates is only for starboard side. The method of numbering is shown in Figure 82.

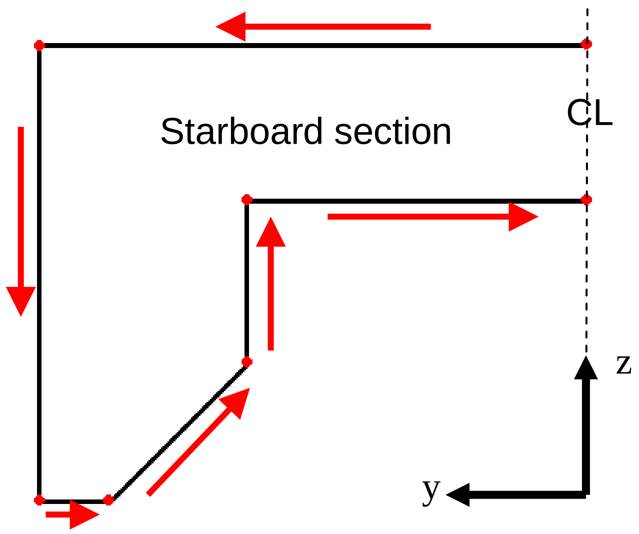


Figure 82: Definition of hull geometry in mgf file. Red arrows shows the numbering of coordinates

The geometry is stored as node coordinates. These follow the *.mgf coordinate system, see Figure 38. Element information is also stored with information about length, normal vector and moment arm. See Figure 83. Red numbers are node numbering and black are element numbering. The coordinates shown in the figure are what is stored in the local coordinate system. For the global axis system, coordinates on both starboard and aport side are stored. The global axis system is the global NED system shown in Figure 39.

Global coordinates are updated for each time step, as well as the crossing point between hull and water surface, see Figure 84.

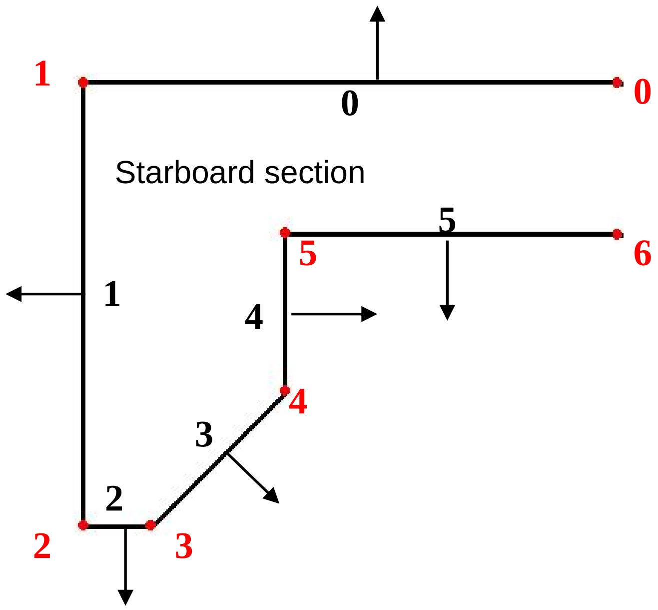


Figure 83: Geometry build up. Elements, nodes and normal vectors

The SESHull class and the airCushion class deals with different parts of the vessel geometry. The coordinate dividing the two is the lowest point on the geometry. If several coordinates are equally low, the closest on to the center line is used. In Figure 83, this point is number 3. Figure 85 shows what geometry is stored in the airCushion object. Compare with Figure 83 to see the difference.

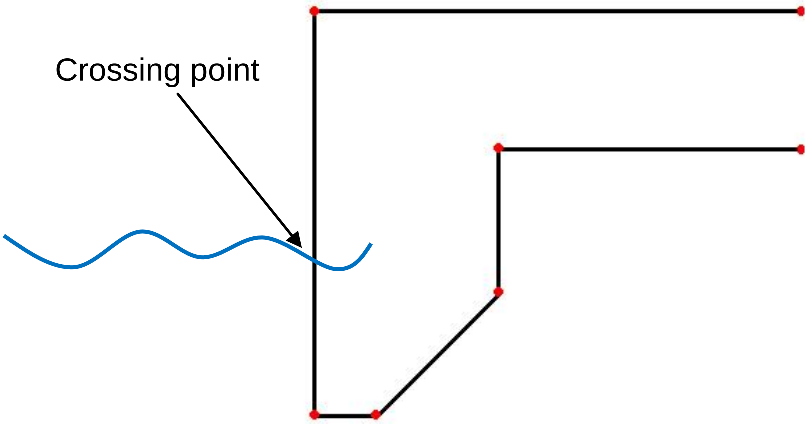


Figure 84: Definition of the crossing point between hull and water surface that is stored in the SESHull object

The crossing point that is stored in the SESHull object is shown in Figure 84. The crossing point inside the air cushion is stored in the airCushion object. The crossing point is stored in local coordinate both for starboard and aport side.

Appendix C.4: airCushion

This class stores all geometric data for the air cushion. This includes the air cushion geometry (local and global coordinate system and element information) and the positions where the different hull sections cross the water surface.

In data

Hull geometry, *.mgf file
Information about longitudinal extension of cushion, that is: position of seals
Global position of vessel
Wave elevation
A constant object
Draught, needed to find static dry point of hull surface
Air cushion pressure
Simulation time

Out data

Air cushion geometry (in local or global coordinates) in node coordinate and element information
Crossing points between water and the hull
Heading vector, telling the global orientation of the vessel

Manner of operation and configuration

The class reads hull geometry from the .mgf file. The data includes length between perpendiculars and section data. Section data includes x coordinate of the section as well as several points (given by y and z coordinates) describing the section. Due to symmetry, the coordinates is only for starboard side. The method of numbering is shown in Figure 85.

The geometry is stored as node coordinates. These follow the air cushion coordinate system, with center of the axis system at longitudinal center of the air cushion, CL and BL. X is positive forward, y towards starboard and z up (see also Figure 41). Also, only necessary geometry to define the air cushion is stored. That is defined at the lowest point on the hull, and inwards towards CL. If several coordinates are equally low, the closest on to the center line is used.

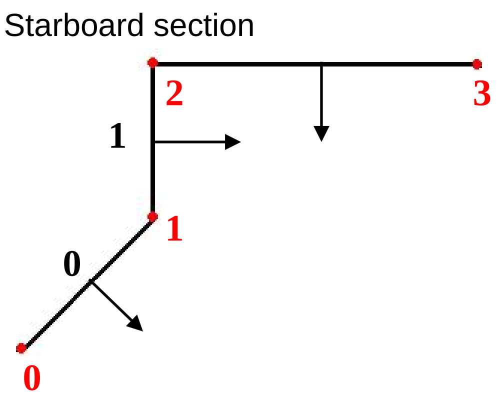


Figure 85: Geometry used to define the air cushion. Compare with Figure 83 to see what coordinates are kept from the original .mgf file.

Figure 85 shows what coordinates are stored in the local coordinate system. For the global axis system, coordinates on both starboard and aport side are stored.

Element information is also stored. This includes information about element length, normal vector and moment arm. The numbering format is shown in Figure 85.

The class has a method called “updateGlobal”. This updates the global coordinates based on the position of the vessel. It also calculates where on the section the water crosses. This information is needed in order to calculate enclosed air cushion volume, leakage under the hull and wet surface. It is also needed in order to find dry area of the section, as it is this area that gives air cushion forces (pressure integrated over the dry area). See Figure 86.

.

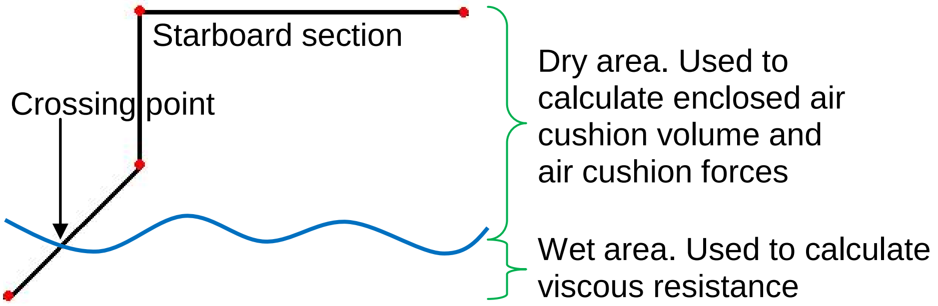


Figure 86: Definition of the crossing point between hull and water surface that is stored in the airCushion object

One last thing the object does is to calculate a direction vector. This is done under the “updateGlobal” method. The vector is used when calculating volume and leakage. As information about direction of sections are lost during the volume and leakage calculations (using absolute values for distances), this vector tells whether the area is positive or negative. See Figure 89.

Appendix C.5: seal

The seal class holds the geometry of a seal.

Assumptions

A seal can be described by a rigid vertical wall
No dynamic of the seal (this assumption fits large sea states best)
No space between seal and hull (depending on the static calculation, see Appendix C.12)

In data

An airCushion object
X and z position of bottom of seal
Type of seal (stern or bow. Stern has value 0 and bow value 1)
Global position of vessel

Out data

Geometry of seal in local and global coordinates

Manner of operation and configuration

The class reads in geometry of the air cushion at the seal. This combined with the height of the bottom of the seal gives the total seal geometry. The geometry of the hull is also included in the case of non-vertical side hulls and large leakage areas (heavy seas). Both starboard and aport side are included in the geometry. See Figure 87.

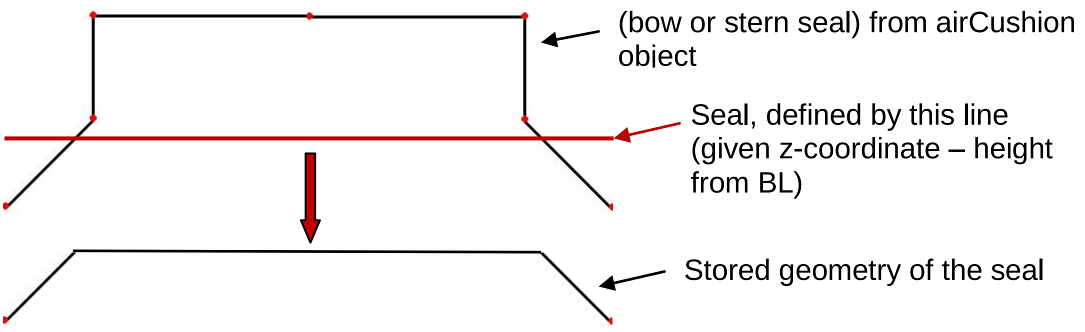


Figure 87: Build-up of the seal geometry

The global coordinates must be updated for each time step by running the "updateGlobal" method. This uses the global position of the vessel as in data.

The seal profile given above is divided into nodes. The distance between the nodes is maximum 1 [m]. This distance can be set to a smaller or larger number. A local (private) variable gives this number.

If the z-coordinate defining the bottom of the seal is too high (higher than wet-deck on section) or too low (lower than the lowest point on cushion section), the class will try with a section on step closer to amidships. If not this either complies, a warning message will be written to the command window. The federate will not stop but run on with either the entire section open (if position to high) or with the seal being a vertical wall going beneath the hull.

Appendix C.6: fanCharacteristics

A class that reads an input file containing fan characteristic. By giving air cushion overpressure as input, it returns the air flow from the fan system.

Assumptions

Fan curves have to be 1-to-1 with pressure and air flow, so that air flow is strictly decreasing with increasing air cushion pressure. A real fan curve has a 2-to-1 relation between air flow and pressure when pressure is higher than a certain value (see Figure 22)

In data

*.fan, file containing fan characteristics
p_c, air cushion overpressure
g, acceleration of gravity
rho, mass density of sea water

Out data

Output from the fan system (air flow in [m^3])

Manner of operation and configuration

When the object is initialized, it reads a *.fan file and stores the data given in this.

It has a method called “getQ_in”. This interpolates the output from the fan system for given air cushion overpressure. It also need g and rho as input, as the *.fan system operates with pressure as “millimeter water column [mmWC]”.

If the pressure is larger than the largest air cushion pressure in the characteristic, output air flow equals zero. If the pressure is lower than what given in the characteristic curve, out air flow is the one corresponding to lowest pressure in the characteristic.

Appendix C.7: SESMass

A class that stores the vessel mass (displacement, air cushion lifted, total)

In data

Path to the simulation temp folder. This contains the vessel model used by VeSim
massDisplacement, total weight of the vessel
massAirCushion, the weight lifted by the air cushion

Out data

massTotal, total weight of the vessel
massDisplacement, vessel's displacement weight
massAirCushion, the weight lifted by the air cushion

Manner of operation and configuration

The object is given the path to the simulation temp folder when initialized. This folder is where the simulator unpacks the vessel model zip file. Among other, this contains a file called "retfun.re10". Among other things, this file contains the mass matrix of the vessel. The (1,1) position in the matrix gives vessel displacement. This value is read and stored.

The default option in ShipX is to use displacement mass. An option in VERES can override this, forcing the user-input vessel mass.

The massDisplacement and massAirCushion must be saved to the SESMass object afterwards.

This object is only used to calculate lift ratio, i.e. ratio between air cushion lifted mass and total vessel mass.

Appendix C.8: cushionVolume

cushionVolume is a class that calculates the enclosed air cushion volume. It can be used to calculate both initial condition (no motion and waves) and instantaneous situation (with motion and waves). These two conditions require different input data.

Assumptions

The air cushion pressure will never give a positive contribution to water level (underpressure relative to atmospheric pressure inside the air cushion will not give a contribution to water level)

Water level inside the air cushion is only function of wave elevation and static air cushion pressure, not e.g. diffraction and radiation waves

The density of geometry (sections from airCushion object) is good enough to obtain accurate calculations

In data

An airCushion object

p_c, air cushion overpressure

draught, if doing static calculation

wave elevation, if doing instantaneous calculation

time, simulation time (if doing instantaneous calculation)

A constant object

Out data

Enclosed air cushion volume

Manner of operation and configuration

There are two different constructors in this class. One is designed for initial calculation with no motion and waves, and one for the instantaneous calculation with motion and waves. This means that the object needs to be initialized for each new calculation. The area that is interpreted as enclosed air cushion volume is shown in Figure 88 and Figure 91.

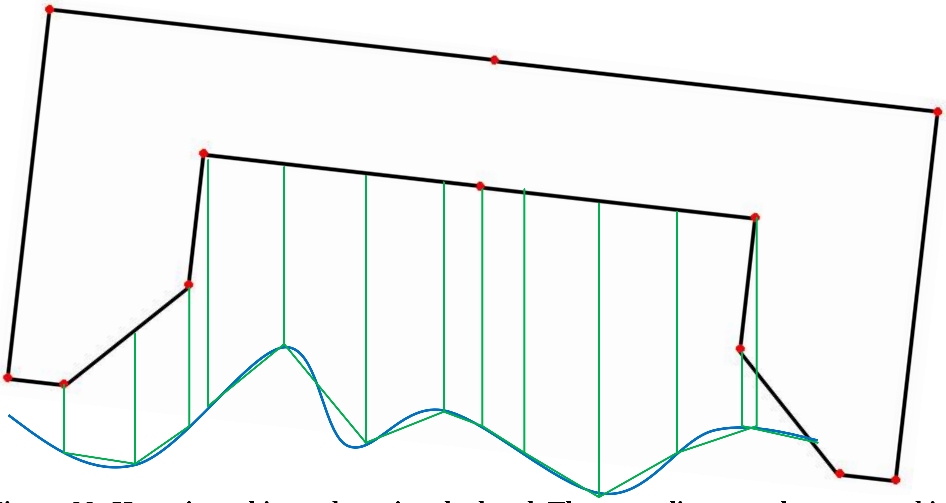


Figure 88: How air cushion volume is calculated. The green lines are the area used in the calculation

The calculations are done twice, once for starboard and once for aport. The geometry is equal but not the wave elevation (if there are waves and motion present).

The cushion area is divided into squares as shown in Figure 88. The top corners of the square areas are the nodes defined in the hull geometry file. If the horizontal distance between the nodes is larger than one meter, more squares are used (with horizontal distance less than one meter). As is the case with seal geometry, this distance may be easily altered in the class. It is defined as an own variable in the class.

In order to cope with positive and negative direction in global coordinates (possibly giving positive and negative square areas), the square areas are calculated using absolute values. If looking close at the right side of Figure 91, one see that this present a new problem with areas outside the cushion being included, adding more volume than is the case (see figure below). This is counteracted by using the vessel direction vector calculated in the airCushion object. Using this, the program decides whether the calculated square area should be added or subtracted to the air cushion volume. The principle is attempted explained below.

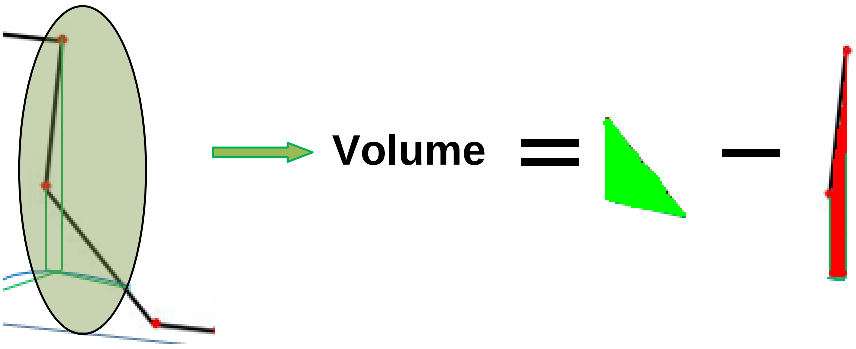


Figure 89: Negative and positive volume when calculating enclosed air cushion volume

Appendix C.9: cushionLeakage

The cushionLeakage class calculates leakage from the air cushion. It is only the geometrical leakage it calculates, i.e. leakage area.

Assumptions

Water level inside the air cushion is only function of wave elevation and static air cushion pressure, not e.g. diffraction and radiation waves

Leakage happens only under the demi hull or under the seals

The density of geometry (sections from airCushion object) is good enough to obtain accurate calculations

In data

A SESHull object

An airCushion object

Two seal objects, stern and bow seal

p_c, air cushion overpressure

draught, if doing static calculation

wave elevation, if doing instantaneous calculation

time, simulation time (if doing instantaneous calculation)

A constant object

Out data

Four choices:

Leakage under stern seal

Leakage under bow seal

Leakage under hull

Total leakage

Manner of operation and configuration

The class has two different methods, one static and one instantaneous. There are two kinds of leakages: Leakage under hull and leakage under seals.

Leakage beneath the demi hulls

One point is checked for leakage. This is the lowest coordinate on the section. If several coordinates are equally low, the one closest to the center line is used. See Figure 90.

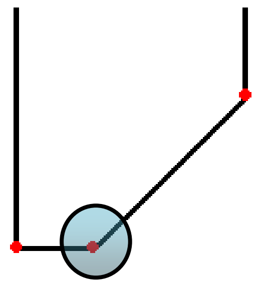


Figure 90: This point is used to check for leakage. See for instance Figure 91 for the total figure

If leakage is discovered, the leakage is calculated as shown in Figure 91. The shortest distance of a and b is used for leakage length. The section's contribution to leakage area is leakage length multiplied with dx, which is the width of the section in longitudinal direction.

The section's contribution to enclosed air cushion volume is the area indicated with green lines. This area times dx gives the section's enclosed air cushion volume.

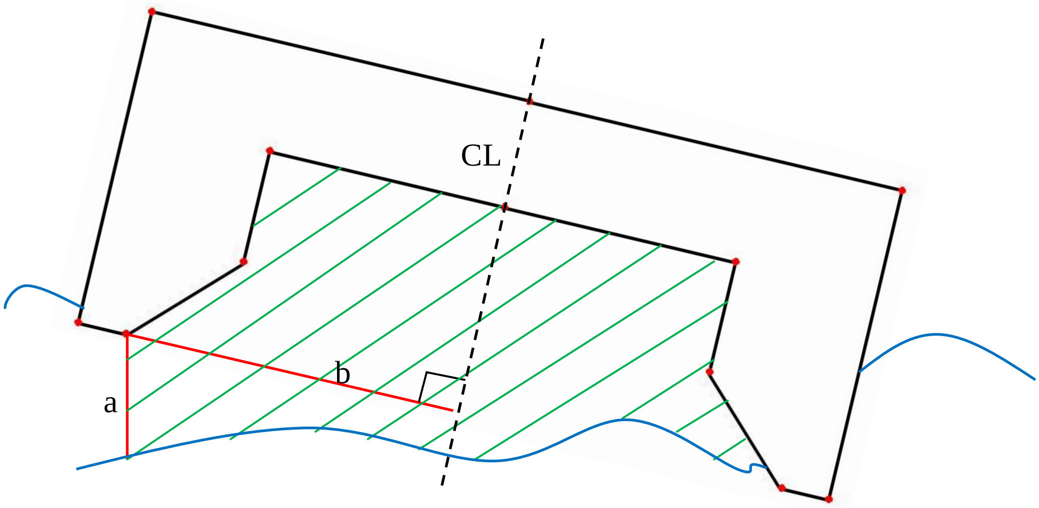


Figure 91: Illustration on how leakage and air cushion volume is interpreted in program

Leakage beneath the seals

This is quite straight forward for the static condition. The leakage area is quite simply the area between hull, seal and water surface. See Figure 92.

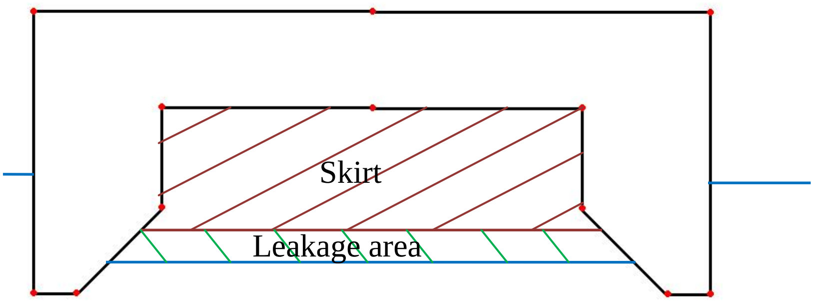


Figure 92: How static leakage under air cushion seal is modeled

The instantaneous case is somewhat more complex. See Figure 93. The leakage area is divided into squares in the calculations. The sizes of the squares are decided in the same manner as explained in the volume calculations.

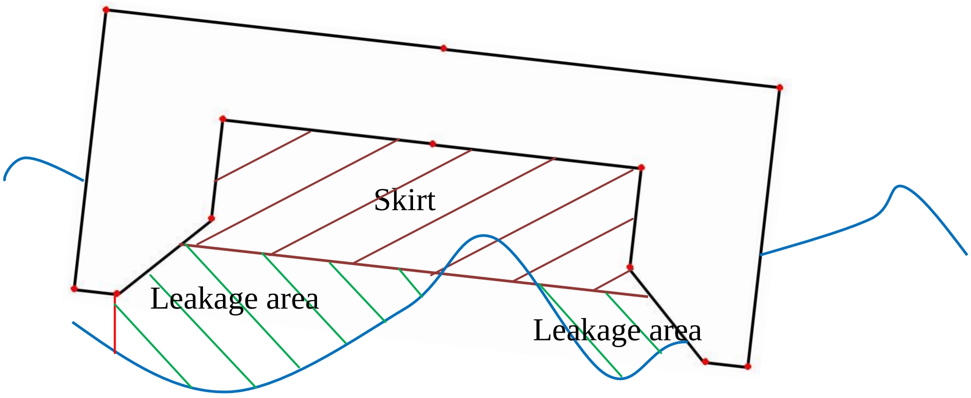


Figure 93: How dynamic leakage under air cushion seal is modeled

Appendix C.10: wetHull

This class calculates wet surface of the hull. This is needed for the viscous resistance calculations.

Assumptions

The sections defining the geometry is dense enough to give accurate results
Water level inside the air cushion is only function of wave elevation and static air cushion pressure, not e.g. diffraction and radiation waves

In data

A SESHull object
An airCushion object

Out data

Wet surface of hull

Manner of operation and configuration

The calculations are done in two step. One for the geometry defined in the airCushion object, and one for the geometry outside the air cushion. This geometry is obtained from the SESHull object. Figure 94 shows the difference between the two steps.

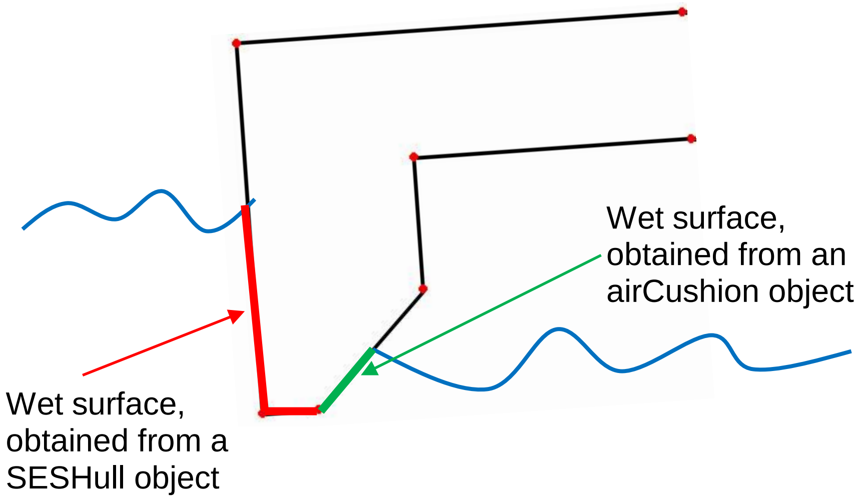


Figure 94: Definition of wet surface, and which objects that hold the geometry

The calculations are done separately for starboard and aport side.

Appendix C.11: SESForces

The SESForces class calculates forces due to the air cushion pressure. The forces are obtained by integrating air cushion pressure over the dry parts of the air cushion.

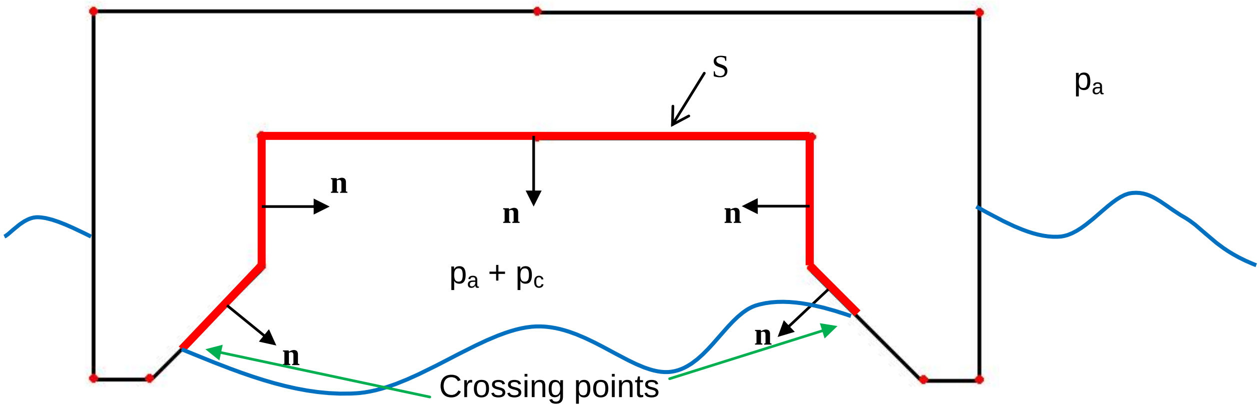


Figure 95: Hull surface used when calculating air cushion forces

Assumptions

Uniform air cushion pressure

The orientations of the hull surfaces (sections) are strictly longitudinal (no forces in local x-direction)

The sections defining the geometry is dense enough to give accurate results

In data

An airCushion object

p_c, air cushion overpressure

Out data

Forces in surge, sway and heave

Moments in roll, pitch and yaw

Manner of operation and configuration

The class reads the geometry of the vessel in local coordinates. It multiplies the air cushion overpressure with the length of the dry area, marked with red in Figure 95. This value is again multiplied with the sections length in x-direction, dx. All geometry data is obtained from the airCushion object.

Moments are calculated using the air cushion axis system. That is, all moments are around the origin of this axis system. This also means that the coordinate of this point must be given as input to the CSI server, in order for the vessel federate to know where the forces are working.

The class is used to calculate forces both in static condition and for every time step.

Appendix C.12: initialSESData

Initial data is here meant to be static conditions. That is the ship laying still (no ship motion) and zero wave elevation. Air cushion pressure is uniform and equal p_0. Air in due to fan is equal Q_0, which should also equal leakage from the air cushion.

In data

A SESHull object
An airCushion object
Two seal objects, stern and bow seal
p_0, static air cushion pressure
draught, the vessel's draught at amidships
nFan, number of fans used
A fanCharacteristic object
A SESMass object
A constant object

Out data

Q_0, static air flow from fans
wet_surface_0, static wet hull surface
cushion_volume_0, static enclosed air cushion volume
A_Leakage_Stat_Calc, see explanation below
A_Leakage_Stat, static leakage area due to static air flow from fans
lift_ratio_0, static ratio between air cushion lift and total vessel mass
Static forces and moments from the air cushion

Manner of operation and configuration

The build-up of initialSESData is very simple. First of all the cushion object is used to obtain the static crossing point between the hull and sea water, using the SESHull and airCushion objects. The constant object, p0, and the draught are used as input to these. It then calculates static data by making the appropriate objects giving these values.

The wet surface is calculated by a wetSurface object, the enclosed cushion volume by a cushionVolume object and the leakage by a cushionLeakage object. Q_0 is obtained by using the fanCharacteristic object with p_0 as input. Static forces are obtained by making a SESForces object with p_0 as input, and the lift ratio is calculated using mass from the SESMass object. The SESForces are only used to find static air cushion forces. The heave force here corresponds to air cushion lift and is stored in the SESMass object (also displacement mass is calculated and stored in SESMass object). The lift_ratio__0 is calculated using the SESMass object.

As the leakage area calculated by cushionLeakage is not necessarily correct according to the static air flow from fans, two variables store information about static leakage area. The A_Leakage_Stat is calculated from Q_0 and p_0. A_Leakage_Stat_Calc is needed to later be able to calculate correctly the dynamic part of the leakage area. It depends on what leakage area is calculated by cushionLeakage. There are two different outcomes:

1. No leakage is calculates by cushionLeakage, or the calculated leakage area is less than A_Leakage_Stat. The static leakage not included in results from cushionLeakage is then considered to be through something not calculated (e.g. louvers or between seals and hull).

2. The leakage calculated by cushionLeakage is larger than A_Leakage_Stat. This is due to incorrect user input (e.g. too high placement of seals). This means that the initial conditions are NOT static condition (as the air flow out from the cushion is larger than air flow into the cushion). A_Leakage_Stat_Calc is set to be equal A_Leakage_Stat.

Situation 2 may not be a critical error. It means that vessel will have some initial instability. This can be very small (if e.g. seals are place 1 [cm] too high), or very large. The calculated static enclosed air cushion volume will not be correct, but this does not play any part in the equations as time goes by.

Note that in situation 2 all leakage will happen under the seals (and under the hull) while in situation 1 some air will leak out from other places. These leakages are included in the static leakage. This means that in situation 2 zero leakage can occur. This is not the case in situation 1.

If case 2 occur, a warning message is sent to the command window.

Appendix D.1: Seakeeping results for case 1. VERES RAO

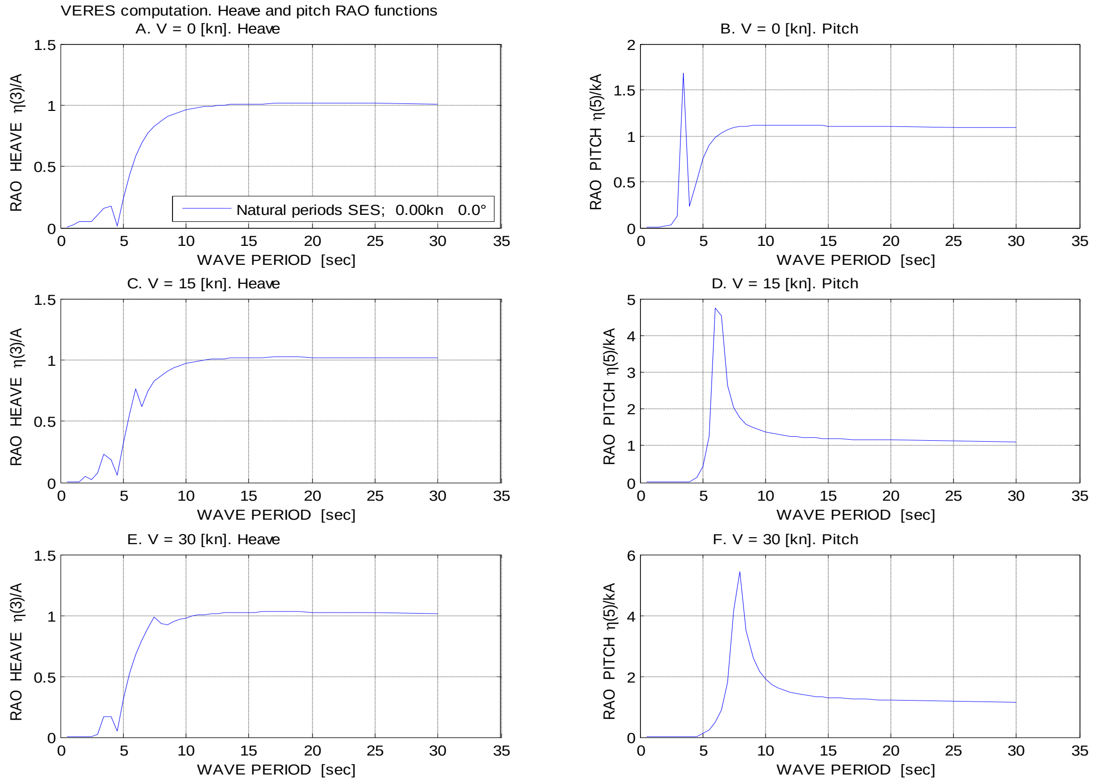


Figure 96: Case1. Heave and pitch RAO functions, zero heading. Results from VERES

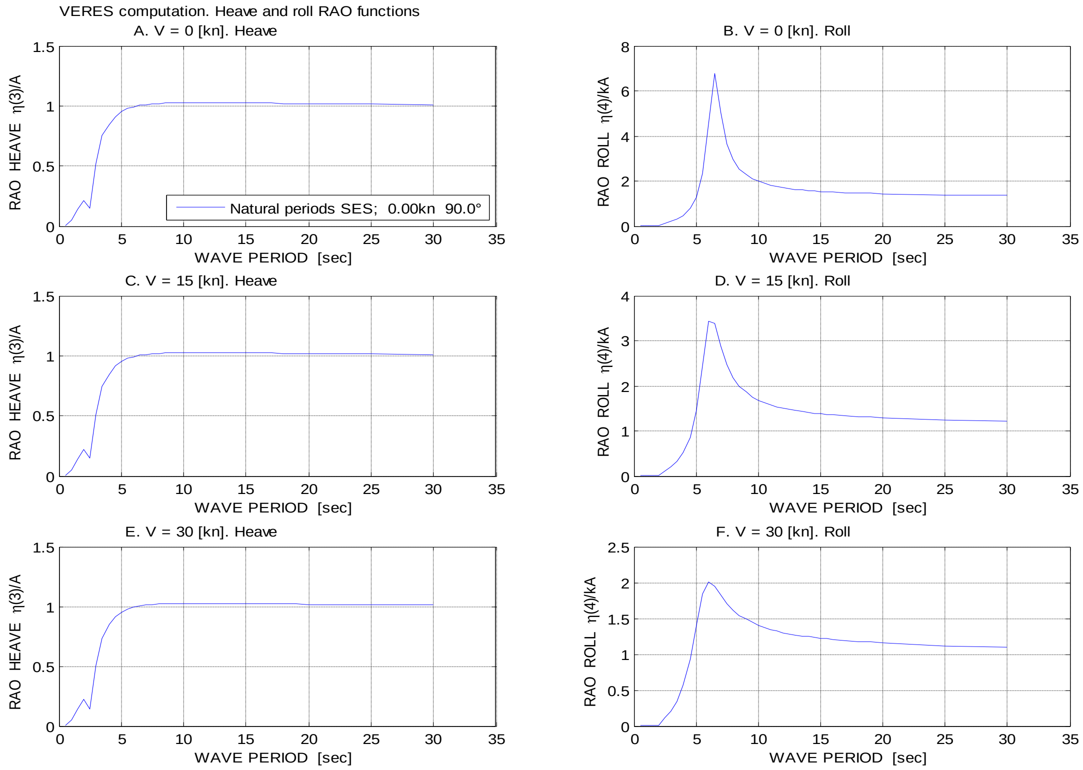


Figure 97: Case1. Heave and roll RAO functions, heading 90°. Results from VERES

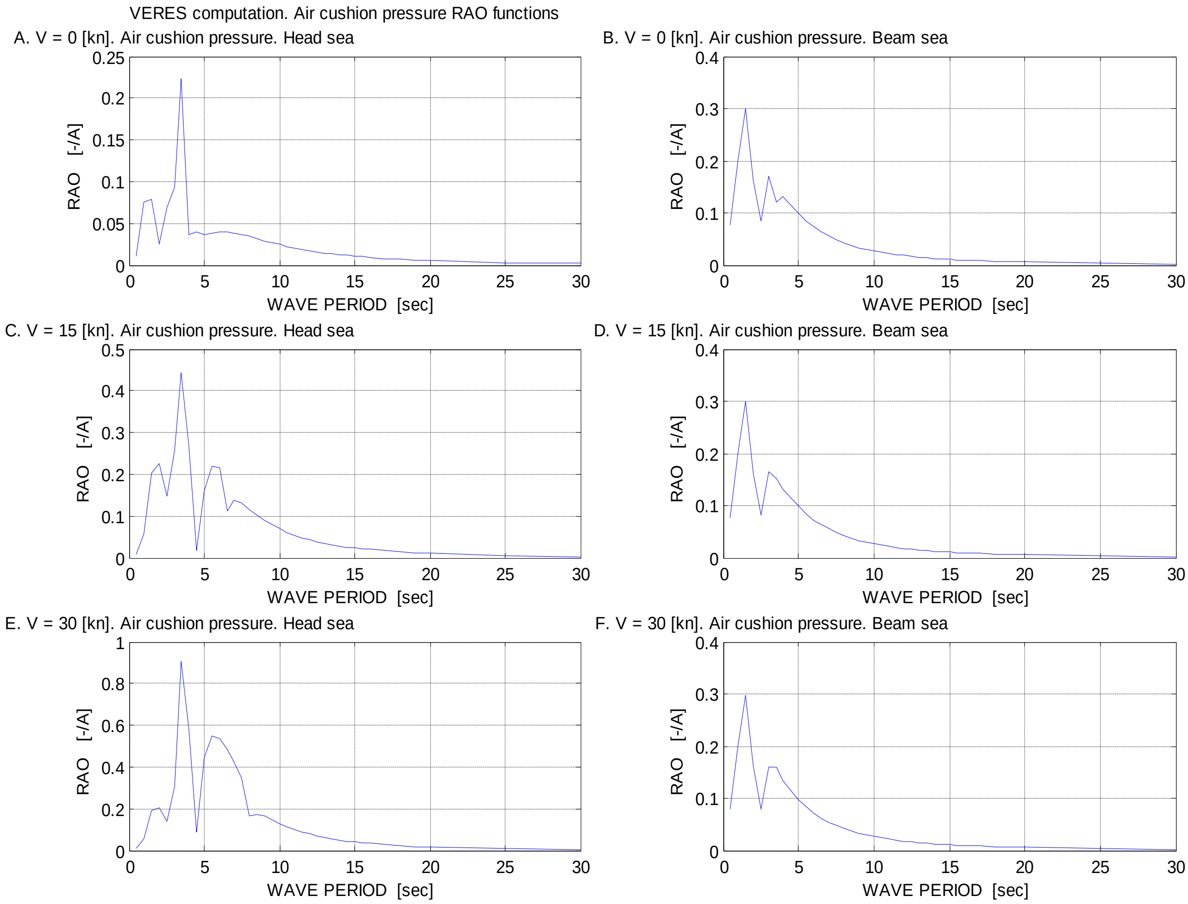


Figure 98: Case1. Air cushion pressure RAO functions, headings 0° and 90°. Results from VERES

Appendix D.2: Seakeeping results for case 1.1. VERES RAO

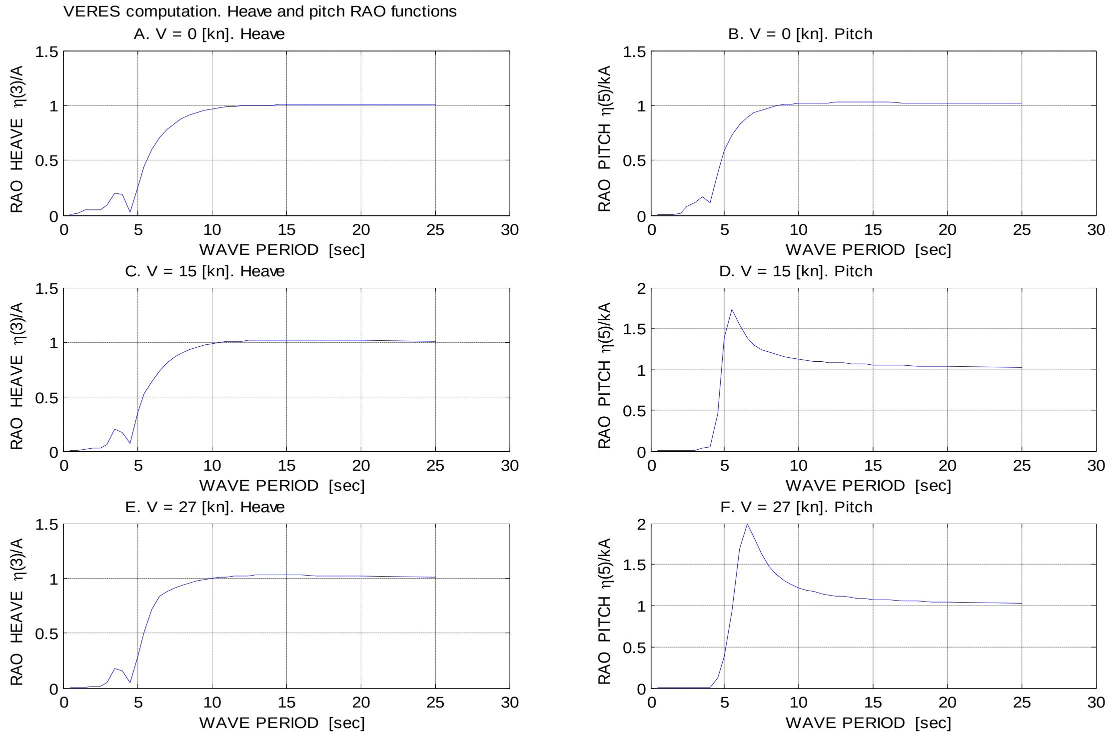


Figure 99: Case1.1. Heave and pitch RAO functions, zero heading. Results from VERES

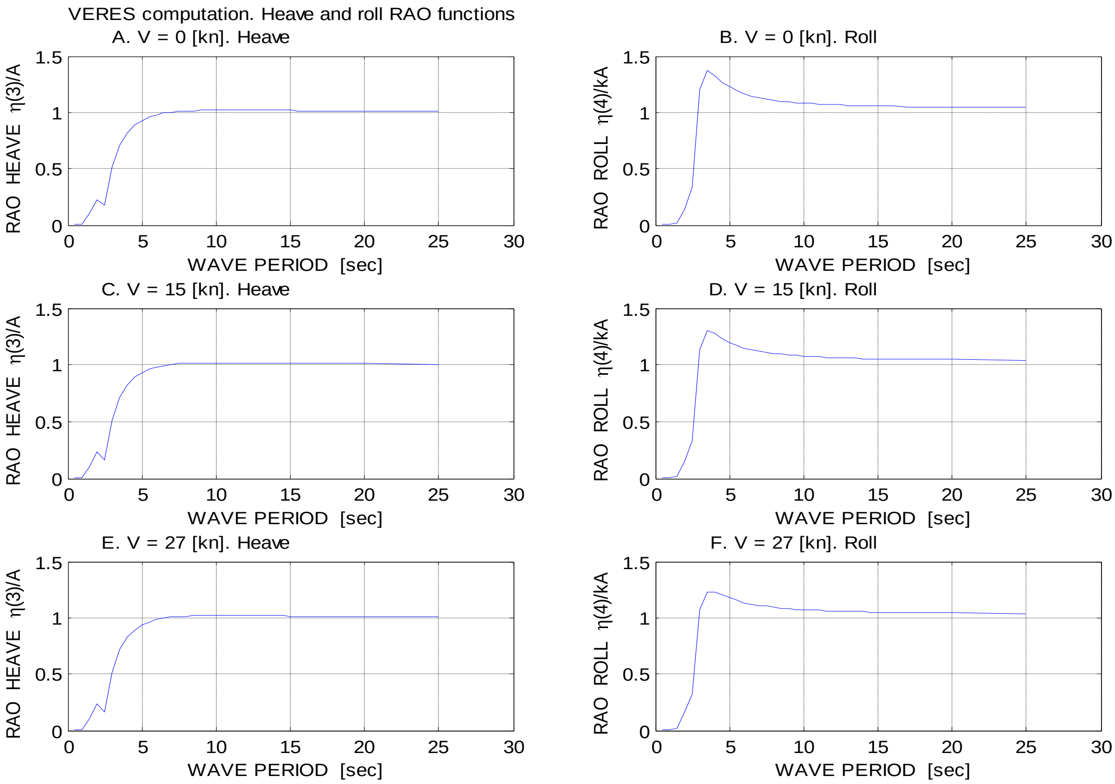


Figure 100: Case1. Heave and roll RAO functions, heading 90°. Results from VERES

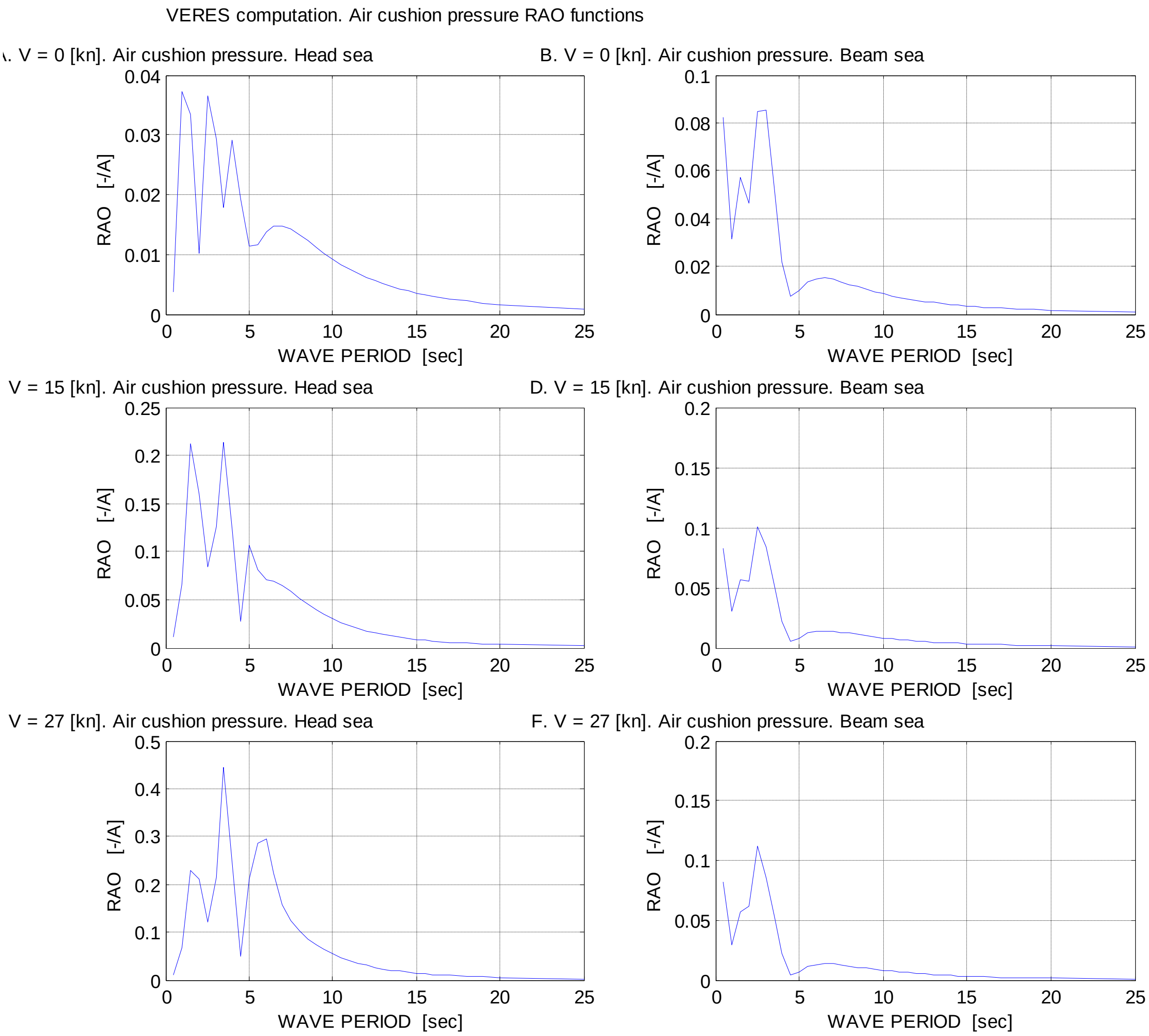


Figure 101: Case1.1. Air cushion pressure RAO functions, headings 0° and 90°. Results from VERES

Appendix D.3: Seakeeping results for case 2. VERES RAO

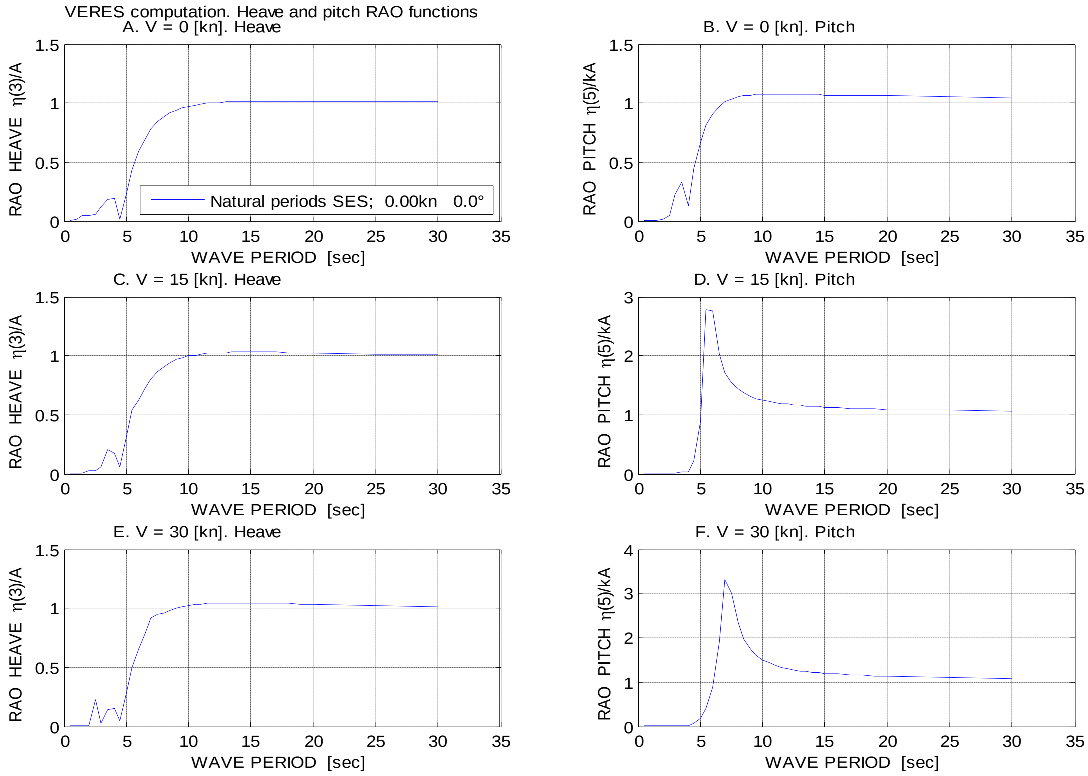


Figure 102: Case 2. Heave and pitch RAO functions, zero heading. Results from VERES

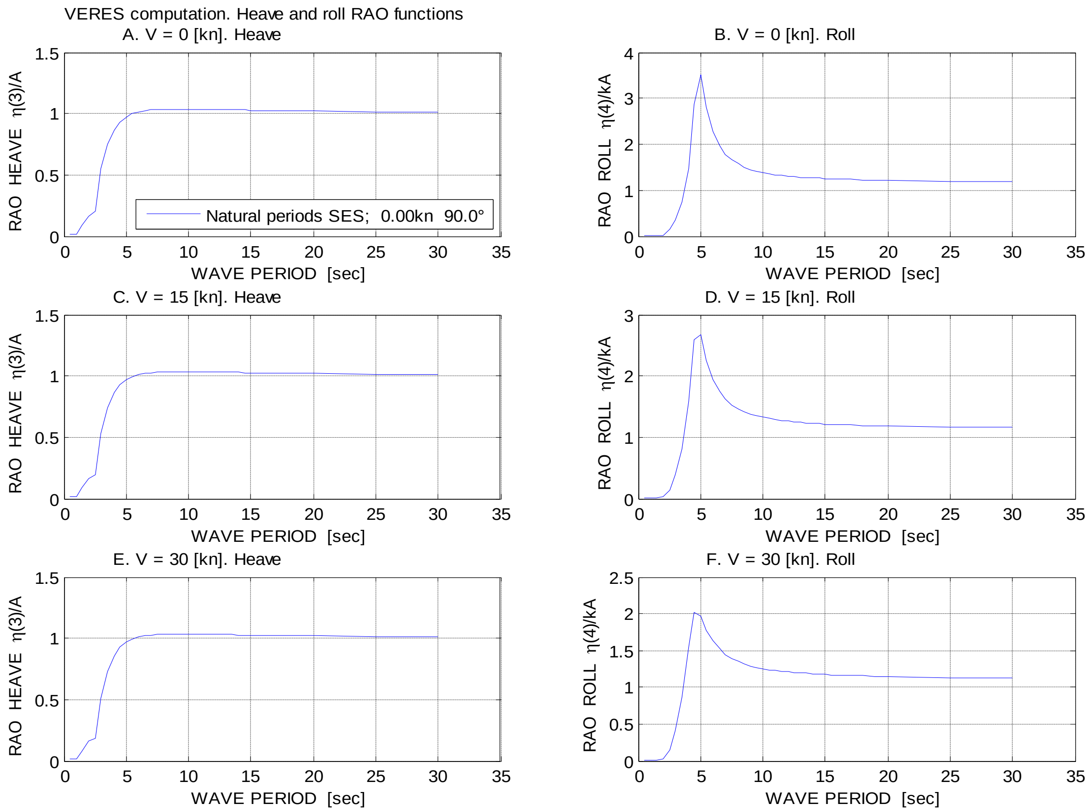


Figure 103: Case2. Heave and roll RAO functions, heading 90°. Results from VERES

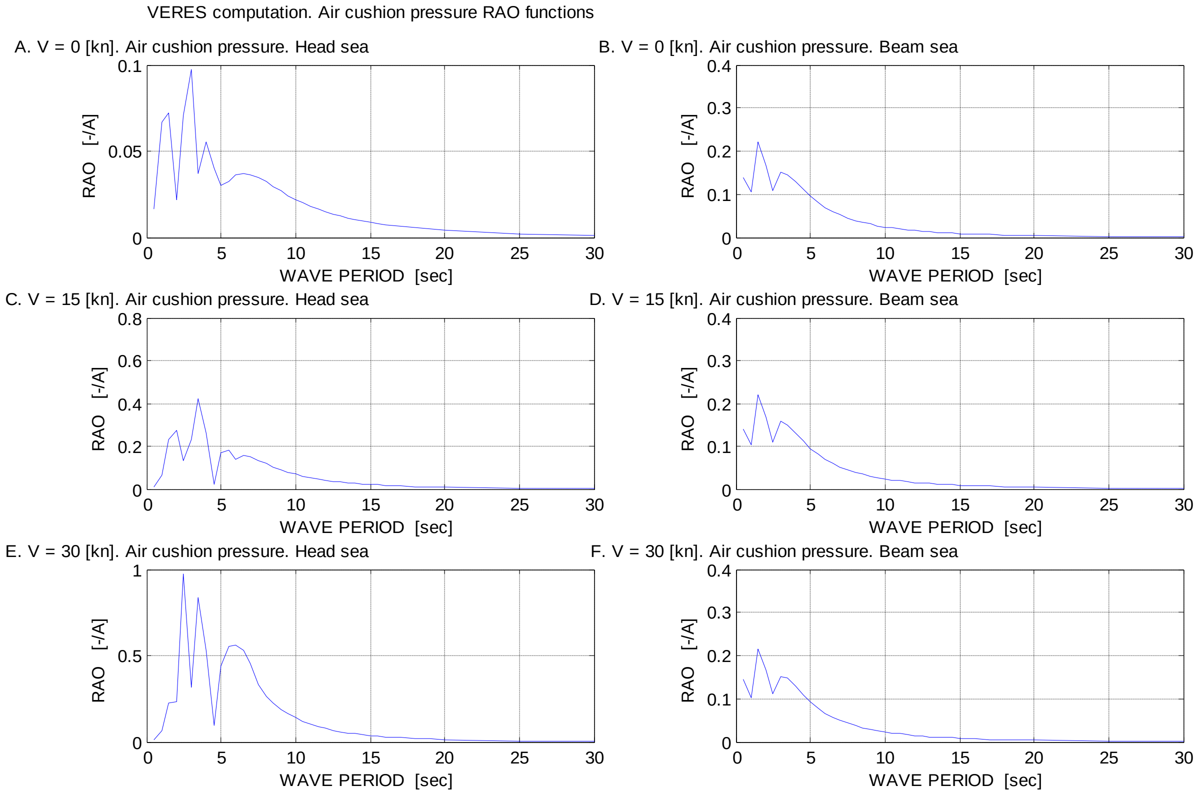


Figure 104: Case2. Air cushion pressure RAO functions, headings 0° and 90°. Results from VERES