NTNU – Trondheim
Norwegian University of
Science and Technology

Application of CFD to seakeeping

Steinar Malin

Marine Technology (2-year)
Submission date: June 2013
Supervisor: Sverre Steen, IMT

Norwegian University of Science and Technology
Department of Marine Technology

NTNU
Norwegian University of Science and Technology
Department of Marine Technology

MASTER THESIS

Title:	**Delivered:**
Application of CFD to sea keeping	17.06.2013
	Availability:
	OPEN
Student:	**Number of pages:**
Steinar Malin	76

Abstract:

Normally hull resistance is something that is estimated in calm water situations, so with no waves present.
This is a state that is an exception out in the real sea. There will almost always be some sort of waves present.
The study of ways to estimate added resistance is thus very important in optimizing hulls and making fuel consumptions lower. But to be able to do this kind of optimization, one has to have reliable and cost efficient methods of predicting this extra resistance due to incoming waves. This is where CFD has an ability to shine,
and there has been some research on this subject for a while now. But there still a lot to learn about such methods.

This thesis was meant to do a study on the reliability and accuracy of two different commercial CFD codes and make comparisons of the results to eachother, and to already done model test results.
The MOERI tanker KVLCC2 with a scale ratio of 58 was used in the simulations. This is a benchmark hull that has been used in several projects earlier to estimate accuracies of different CFD codes.

Calm water simulations were done in both FINE/Marine and Star-CCM+, and these results were then compared to results achieved in simulations that included different kinds of waves. And in this way the added resistance due to waves can be observed as an additional resistance appearing due to the incoming

Keyword:

Computational fluid dynamics
Added resistance

Advisor:

Sverre Steen

Address:
NTNU
Department of Marine Technology
N-7491 Trondheim

Location
Marinteknisk Senter
O. Nielsens vei 10

Tel. +47 73 595501
Fax +47 73 595697

Scope of work

Underneath is the problem text from the project thesis. My master thesis consists of continuing on the work done in the project.

PROJECT THESIS IN MARINE TECHNOLOGY

AUTUMN 2012

FOR

Steinar Malin

Application of CFD to Seakeeping

The capability of commercial CFD codes is increasing, and so is the interest of applying such codes to different problems in marine hydrodynamics. Especially problems that can hardly be addressed by the conventional potential flow solvers are interesting to explore with CFD, which in this context means RANS type solvers. One such problem is seakeeping, and especially added resistance due to waves. The objective of the combined project and master thesis is to investigate the capabilities of two commercial CFD codes, Fine/Marine and Star CCM+ with respect to prediction of added resistance due to waves. For this purpose, comparison with (existing) model tests is required. It is foreseen to perform the computations on the benchmark hullform KVLCC2, where both model test data and previous CFD calculation data exist, so that a solid basis for comparison is available.

For the project thesis, learning the CFD codes will be a formidable challenge, and not too much in the way of computations shall be expected. As a suggested scope of the project, the following is proposed:

- Give an overview of the CFD codes applied, with capabilities, typical applications and limitations.
- Perform calm water resistance computations with KVLCC2 and compare the results with model test results, and possibly published CFD results.

If time allows;

- Perform simulation of regular waves without hull present in the computations
- Do some preliminary simulation of the hull in waves.

In the thesis the candidate shall present his personal contribution to the resolution of problem within the scope of the thesis work.

Theories and conclusions should be based on mathematical derivations and/or logic reasoning identifying the various steps in the deduction.

The candidate should utilize the existing possibilities for obtaining relevant literature.

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

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

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

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

The thesis shall be submitted in two copies:

- Signed by the candidate
- The text defining the scope included
- In bound volume(s)
- Drawings and/or computer prints that cannot be bound should be organized in a separate folder.
- The bound volume shall be accompanied by a CD or DVD containing the written thesis in Word or PDF format. In case computer programs have been made as part of the thesis work, the source code shall be included. In case of experimental work, the experimental results shall be included in a suitable electronic format.

Supervisor : Professor Sverre Steen

Advisors	: Anders Östman and Eloise Croonenborghs
Start	: 01.09.2012
Deadline	: 14.12.2012

Trondheim, 05.09.2012

Sverre Steen
Supervisor

Preface

This is the final thesis of my master degree in Marine Technology at Norwegian Institute of Science and Technology (NTNU) in Trondheim, Norway.

Originally this thesis was supposed to be about the capabilities of the two CFD codes used in FINE/Marine and Star-CCM+. Comparisons to each other, model tests and earlier CFD results where to be done.

Due to problems with getting licenses to work, and also getting resources to do the calculations, time just ran out in the end to do the proper simulations.

So this paper consists of the results of the simulations I did manage to do in time and also a study on the subject of CFD and seakeeping.

Trondheim 17-06-2013

Steinar Malin

Contents

Figure list

Table list

Appendix list

Nomenclature

AP	Aft Perpendicular
CFD	Computational Fluid Dynamics
DOF	Degrees of Freedom
HEXPRESS	Mesh generator in the FINE/marine package
FINE/marine	CFD code from Numeca
FVM	Finite Volume Method
DNS	Direct Numerical Simulation
LES	Large Eddy Simulation
RSM	Reynolds Stress Model
Star-CCM+	CFD code from CD-adapco

Symbol	SI unit	
B	[m]	Ship beam
Caw	[-]	Added resistance coefficient, $C_{aw} = \frac{R_{aw}}{\rho g \zeta^2 B^2 / L}$
Cb	[-]	Block coefficient
Cw	[-]	Water plane coefficient
D	[m]	Ship depth
Fn	[-]	Froude number, $F_n = \frac{v}{\sqrt{gL}}$
Fx	[N]	Resistance in x-direction
g	[m/s^2]	Acceleration of gravity, 9.81
H	[m]	Wave height
I	[kg m^2]	Moment of inertia
K	[J]	Turbulent kinetic energy
KG	[m]	Vertical center of gravity over keel
L	[m]	Length scale
Lpp	[m]	Length between perpendiculars
Lwl	[m]	Length on water line

LCF	[m]	Longitudinal center of floatation
LCG	[m]	Longitudinal center of gravity
Raw	[N]	Added resistance due to waves
T	[m]	Ship draft
T	[s]	Wave period
u_t	[m/s]	Friction velocity
V	[m^3]	Volume displacement of ship
V	[m/s]	Velocity scale
y_{wall}	[m]	Distance from wall to first cell
y^+	[-]	Dimensionless wall distance
λ	[m]	Wave length
ρ	[kg/m^3]	Density of water
ω		Specific dissipation
ε		Turbulent dissipation

Summary

Simulations of the KVLCC2 hull in calm water were done in both Star-CCM+ and FINE/Marine with a Froude number of 0.142. Then the same hull was simulated while going forward in head waves.
In FINE/Marine this was only done for one wave length, 5.062 meters. But in Star-CCM+ simulations were done for 7 different wave lengths. The added resistance was then found for each case, and a plot was made giving the relation between wavelength and added resistance coefficient. The shape of this plot was similar to the typical added resistance plots like this. But each single result is not expected to be very accurate because of the big variations of results through time.

When comparing the calm water resistance calculated in FINE/Marine with Star-CCM+ and model tests, we see that FINE/Marine is 0.65% away from the results from model tests, and Star-CCM+ is 0.44% wrong. So these results are good, and close to what is expected.

The motions on the other hand are differing a little more. Pitch motion in Star-CCM+ is 16.67% off target, while FINE/Marine is 12.45% off. The big difference is however in heave motion, where Star-CCM+ misses with 4458% and FINE/Marine with 25.86%.

When comparing the added resistance found from the one wave in FINE/Marine, with the same wave in Star-CCM+. It is seen that FINE/Marine is over-predicting experimental model tests with 73% and Star-CCM+ over-predicts with 40%.

These results does in no way represent the accuracy provided by the CFD codes, but rather the quality in which simulations have been set up.

The reason for not having done more wave cases in FINE/Marine was due to a lot of problems with getting the license of FINE/Marine to work. And when it finally worked, the only computer resource I had for that program was too small for the time left. There was only time left to have one simulation done. There were also problems with acquiring computer resources for the Star-CCM+ program. Simulations were done for all the wave cases, but these simulations had to be tuned more to get more accurate and reliable results.

1 Basis of CFD

CFD is a way of using numerical methods and different algorithms to solve and analyze problems related to flow of fluids, heat transfer or other phenomena such as chemical reactions.
The technique is very powerful and has a huge area of uses both in industrial and non-industrial applications. The closest to us is hydrodynamics of ships and marine engineering with respect to loads on offshore structures. But the applications go far beyond that.

- Aerodynamics
- Combustion in turbines
- Turbo machinery
- Electronic engineering
- Chemical process engineering
- Environmental engineering (pollutants)
- Hydrology and oceanography
- Meteorology
- Biomedical engineering (blood flows through veins etc.)

The first uses of CFD were by the aerospace industry in the 1960s and out, where it was integrated in their routines for manufacturing and research jet engines.

Even though the governing equations of CFD are old, and have been known for over a century. The mainstream computer has only existed for less than 30 years. This means that CFD still is an incomplete tool which is nowhere near reaching its full potential yet.
There has been an immense progress over the last few years, and surely there will be even more progress in the years to come. Mainly due to the increase in computational power that is showing each year.

The fact is that solving the Navier-Stokes equations proves to be difficult, because there is very seldom an analytical solution to it. So numerical methods are necessary, and these methods need a lot of computer resources to be solved correctly and efficient, especially if there is turbulence involved or other complex phenomena such as transonic flows.
So even for the super computers we have today the time dependent, three dimensional flows of turbulence is very consuming and not straight forward.

These problems are under ongoing research and developers are coming up with new software that is able to improve the speed and accuracy of such complex simulations at a frequent basis.

So why CFD? There are several big advantages with using CFD over experiment based approaches.
There is a greatly reduced cost and time related to testing of new designs.
The opportunity to study systems that otherwise would be difficult to study, like very big systems that would be difficult to model indoors, or systems that could prove to be dangerous. Testing of dangerous systems would be important in accident scenarios for example.
Also optimization of performance of products will be a lot cheaper because of there are no limitations for doing parametric studies.

The way we solve problems when there is no analytical solution, is to make simplifications and assumptions that may pollute the result in some ways.
We make simplified boundary conditions and simplified dynamic equations for the conservation of mass, momentum and energy. But with the increasing power of CFD we are able to solve more complete forms of these equations and therefor get more accurate results.
There are also problems which are impossible to solve analytically in any way, and is in need of being solved numerically through CFD to get any reasonable results at all.
Thus with the help of CFD we can develop a more deep understanding of flow characteristics in certain processes and will therefore be able to further increase the knowledge used in the design process.

1.1 How does it work

The commercial software provided today uses numerical algorithms to solve the fluid problems,
but to make the products user friendly and easy to use, they are integrated into nice graphical user interface that makes input of parameters easy also provides the tools for examining the results in the end. This means that most CFD software contains three main things, which are:

- The pre-processor
- The solver
- The post-processor

These three items together makes the CFD code of most commercial software out there today. The pre-processor contains the option to feed in all the input needed for defining a flow problem, which is easily done by means of the graphical user interface provided. The input includes:

- The definition of the geometry of the problem and the region of interest. This is called the computational domain.
- Division of the computation domain into smaller, non-overlapping sub domains. This is referred to as a mesh of cells or elements.
- Definition of the fluid properties like density, viscosity, velocities etc.
- Selection of the appropriate physical models to be simulated, like the decision of laminar or turbulent flow. Which turbulence model to choose. The addition of waves. Gravity. Steady or un-steady flow. Etc.
- And at last the definition of the needed boundary conditions, often in sea keeping analysis it is only needed to define the ones that touch the domain boundaries.

The grid with all the cells or elements is the baseline for the solution of the fluid problem. The flow problem's solution is solved and defined at nodes in each of these cells. The most optimal grids are also non-uniform which means that the cells are generally smaller in areas where there are big movements, high velocities or big changes in values between one cell to another.

More cells give more nodes which in term give a more accurate solution. So the solutions accuracy is highly dependent of the number of cells in the grid. But more isn't always better since the increase in cells will also greatly increase the resources needed to solve the problem. So the idea is to find an area of optimal cell count which will give the desired accuracy in the appropriate amount of time.

The final step of the pre-processor is then to convert all the information in a way that the solver is able to handle and use it in an efficient matter.

The solver makes use of the underlying equations that is needed to solve a flow problem, and generates a result based on the input given by the pre-processor. An overview of the parts included in the work flow of the solver includes an approximation of the variables that are unknown by the means of simple functions. Then there will be a discretization by substituting the approximations into the governing flow equations. This will then lead to a solution of the algebraic equations.

One can talk about mainly three different numerical techniques used in the solving of flow problems.
These techniques are: finite difference, finite element and spectral methods.
The difference in these techniques can be observed in the in the way the initial approximation is done and also with the discretization process.
The most used in recent CFD codes is the finite volume method. This method is the most know method and also the most thoroughly validated CFD technique. It is also the one used in both Star-CCM+ and FINE/Marine.

The steps of FVM are; first a formal integration of the governing equations of the flow problem over the whole computational domain. Then the discretization makes us of some finite difference type of approximations for the terms in the integrated equation, which represents some flow properties like convection, sources and diffusion. This will turn the integral equations into a set of algebraic equations.
The last step is to solve these algebraic equations by the means of iterative methods.

It is the first step that makes this method so attractive and also much simpler to understand. This first step, control volume integration, makes use of the conservation of relevant properties in each of the sub domains. One can express a flow variable e.g. enthalpy inside a finite control volume as a balance between different processes that will change the variable, in a decrease or increase due to convection or diffusion in the control volume. This is the transport equation explained in words.

The post-processor is the last step in the CFD workflow, and is also a step that has been more and more usual in later years due to increase in computer resources and graphics capabilities of workstations today, and therefore most of the commercial CFD codes today include an extensive package of data visualization tools.

Some of the things provided by a post-processor are different plots, like vector plots and tube plots, contour plots, surface plots and particle tracing. There are also possibilities to look at geometry and grid, and manipulate views. In some dynamic simulations the option to make animations can also be very useful. As for example in sea keeping analysis you can make a movie that will be represent the ship advancing in head waves, while you can see the diffraction waves, kelvin pattern and the motions of the ship as it hits the waves.

There are also options to export results for further processing in other types of software.

1.2 Governing equations

The core of computational fluid dynamics consists of a couple of very important equations. These are the fundamental governing equations of fluid dynamics, which is the continuity, momentum and energy equations. They are mathematical statements of three physical principles that all fluid dynamics are based upon.

F=ma (Newton's second law), mass conservation and energy conservation.

And it is this that FINE/marine's flow solver is based upon. It uses the incompressible unsteady RANSE, which stands for Reynolds Averaged Navier-Stokes Equations. The velocity field is obtained by the momentum equations, and the pressure field is given by the continuity equation, which is transformed into a pressure equation.

The mass conservation says that no mass can be destroyed or created in the flow field, and will lead to the equation:

$$\frac{\partial \rho}{\partial t}+\frac{\partial}{\partial x_j}(\rho u_j)=0 \qquad (1.1)$$

The energy equation starts out at the first law of thermodynamics, which states that

Change in total energy = Work by external forces + Heat supplied to the system

This implies that the energy of a closed system is constant, which will lead to the energy conservation equation.

$$\rho\frac{DE}{Dt}=\frac{\partial}{\partial x_j}(\sigma_{ij}u_i)+\rho f_i u_i-\frac{\partial}{\partial x_j}\left(-K\frac{\partial T}{\partial x_i}\right) \qquad (1.2)$$

Newton's second law will in turn lead to the famous Navier-Stokes equation, and together with mass and energy conservation they govern a vast majority of any fluid flow in a laminar flow field, and it is also the starting point of any approach to give a description of a turbulent flow.

$$\rho\frac{Du_i}{Dt}=-\frac{\partial p}{\partial x_i}+\left[\mu\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}-\frac{2}{3}\delta_{ij}\frac{\partial u_k}{\partial x_k}\right)\right]+\rho f_i \qquad (1.3)$$

In case of incompressible flow which is a property met in ship hydrodynamics, the equation reduces to a more simplified form shown underneath. Even though it looks much simpler, it is still one of the more complicated differential equations in mathematical physics. (White 2006)

$$\rho \frac{Du_i}{Dt} = -\frac{\partial p}{\partial x_i} + \mu \nabla^2 u_i + \rho f_i \qquad (1.4)$$

The equation lead out from newton’s second law is the real Navier-Stokes equation, but it is increasingly more common in modern CFD literature to call the collection of mass, energy and momentum equations the full solution of the Navier-Stokes equation. (Wendt 2010)

1.3 Turbulence modeling

One concept that will further complicate the solving of flow problems is that flows are not always stable; in fact will all flow problems become unstable when a certain Reynolds number is achieved within the flow. This is true for both complex and simple flows no matter what the nature of the flow problem is. Most engineering types of flow problems are turbulent, so this is a concept that is very important.

$$Rn = \frac{V \cdot L \cdot \rho}{\mu} \qquad (1.5)$$

Reynolds number is a dimensionless number which expresses the relation between inertial and viscous forces in a fluid. When the Reynolds number is over about 4000, it is usually defined as a turbulent flow. In contrary the flow is laminar if the value is beneath 2000.

For the case of the KVLCC2 hull in this thesis the value would be calculated like this:

$$Rn = \frac{1.045 \cdot 5.5172 \cdot 1000}{0.0010987} = 5247542$$

Which is above the breaking point for turbulent flow, and it can then be said that our problem is in the turbulent regime. This means that there is a chaotic, random state of motion in the fluid, which means that the velocity and pressure in the fluid will fluctuate continuously with time throughout the fluid region. These velocity fluctuations will create additional stresses on the fluid and body in the fluid. These stresses are called the Reynolds stresses.

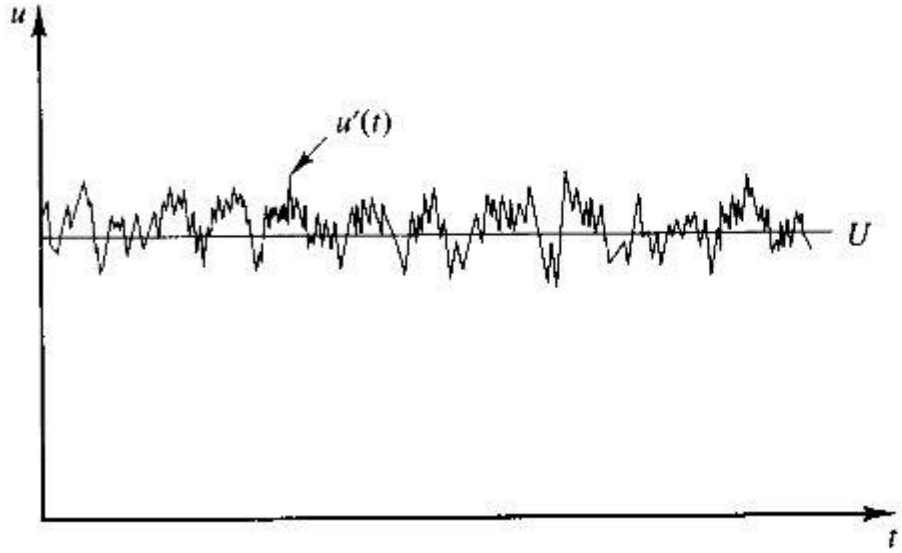


Figure 1: A typical point velocity in turbulent flow (Blazek 2001)

There is a lot that can be said about the transition between laminar and turbulent flow, but general purpose CFD today generally skips the transition part. The common thing is just to define a problem as either fully laminar or fully turbulent.

1.4 Different turbulence models

The solving of the extra Reynolds stresses that appears because of turbulence is what complicates things when it comes to solving the governing equations. The methods with highest accuracy and complexity are represented by direct numerical simulation (DNS) and the large eddy simulation (LES). These method are however not very applicable in engineering problems at this point. These methods are only applicable for simple cases where the Reynolds number is low. It is still an important tool when it comes do understanding turbulent structures, and is also used in verification, development and calibration of new turbulence models. On the other hand in engineering practices it is more convenient to take a more approximate approach to modeling the turbulence.

LES is the next step in terms of approximation, which resolves the larger eddies accurately, and only makes approximations for the small eddies. This is a method which needs much less grid points than the DNS, and thus is applicable to flows with higher Reynolds number than the DNS.

It is still a very demanding method, and accordingly is not a very viable method to use on engineering problems.

RANS equations are the next step in approximations. Reynolds decomposition is the backbone of this method, which means that one splits a quantity in a time steady part and one fluctuating part. The Navier-Stokes equation can then be simplified by taking the mean value of the steady and fluctuating part, and then substitute in the equation.
This will lead to the Reynolds stresses mentioned earlier, which is the result of the turbulence. The approximation of using time averaged values opposed to using exact values for each point and every eddy in the flow is considered exact enough for most uses, but is also necessary to be able to solve turbulent problems in more complex flows with the computer resources available today.
This method is able to handle substantially coarser grids than the LES and DNS, and stationary mean solution can be assumes. Both these properties will reduce the computational time a huge amount, and thus is this method very popular in engineering applications. The averaging done in RANS is a good approximation, but will not provide a detailed picture of the turbulence. But this is something that is not too important in most problems.

There are also several different ways of solving the RANS equations, which have different grades of approximations and accuracy.

- Algebraic
- One-equation
- Multiple-equations
- Reynolds-stress models (RSM)

The three first models on the list are called the first-order closures, and RSM is called a second-order closure. Where the three first are based on the boussinesq approximations. These are the most commonly used models.

Boussinesq approximations use an algebraic equation, which includes determining turbulent viscosity and solving the transport equations to be able to determine the turbulent kinetic energy and dissipation.
The Boussinesq approximation also assumes that the turbulent shear stress is related linearly to the mean strain rate and this proportionality factor is the eddy viscosity. Thus it is only needed one variable, the eddy viscosity to solve the equation.

The most known and used first-order closures are:

- Spalart-Allmaras one-equation model
- K-ε two-equation model
- K-ω two-equation model

1.4.1 Zero-equation models

The zero-equation models assumes that the velocity scale V, and the length scale L are related by algebraic equations to the local flow properties. As an example if you have a wake or a free shear layer, V is going to be proportional to the velocity difference across the flow, and L will be proportional to the width of the layer. And in a boundary layer, L would relate to the wall distance and V by the relation $L \cdot \frac{\partial u}{\partial y}$.

Algebraic models like this have the advantage and disadvantage of being very simple, and can thus be used for problems such as simple shear flows, such as jets and wakes.

1.4.2 One-equation models

The one equation models are better than the zero equation models (algebraic) in the way that they try to implement an eddy viscosity that is no longer dependent only on the local flow conditions, but also the flow history. Most approaches define V and L separately and then find the eddy viscosity based of those values. V is mostly defined as the root of the kinetic energy per unit mass of fluid generated by the fluctuating in velocity. A transport equation can then be made from the Navier-Stokes equations to determine this kinetic energy. This is the one transport equation present in the one-equation model.

Spalart-Allmaras is a popular one-equation model which works with a transport equation for an eddy-viscosity variable only.

1.4.3 Two-equation models

In general, it is advisable to solve two different transport equations for V and L, which is the corner stone of the two-equation-models. In addition to the transport equation for kinetic energy, K, one has to solve an equation for a quantity that can determine L.

Two-equation-models are very popular in industrial use because of it is the simplest closure that does not need geometry or flow regime dependent input.

The K-ε-model is the most popular model, where epsilon is the rate at which turbulent energy is dissipated by the viscosity action on the smallest eddies. A transport equation is solved to determine epsilon and is then used to find the length scale, L, by the relation:

$$L = \frac{C_\mu K^{\frac{3}{2}}}{\varepsilon} \qquad (1.6).$$

The K-ω-model is another highly used model, where omega is the frequency of the large eddies.
The rest is the same as before and L is decided by the relation:

$$L = \frac{K^{\frac{1}{2}}}{\omega} \qquad (1.7)$$

These two models both are good at different fields, and problems.
The K-ω-model is as an example very good close to walls in the boundary layer flows, and especially when large pressure gradients are present. One disadvantage of this method is that it is very sensitive to the free stream value of ω. Thus one has to be very careful when setting this value, because very inaccurate results may appear in both free shear flows and boundary flows.

The K-ε-model on the other hand is much more capable of bringing good results when less care has been taken when setting the free stream values. But as a disadvantage this model was generally not so good at handling large pressure gradients.
But this is something that has been improved in other models, such as the SST-Menter model which takes the best of both these models. It will behave as the K-ω-model close to walls, and shift more and more to a K-ε-model when dealing with the free stream.

1.5 Reynolds stress model

RSM is trying to be more accurate and actually solve the transport equations for each of the stress components, and will thus have to deal with 6 more coupled equations. This will of course lead to a much more expensive problem to solve in case of computer power.

1.6 Boundary layers

Initially the theories of turbulence were examined in the thin shear layers where there is a turbulent structure present. These are regions where the velocity experiences large changes in a small cross stream area.

Mathematically explained: $\frac{\partial}{\partial x} \ll \frac{\partial}{\partial y}$.

This is something that will present itself when there is a flow over a wall or a solid body, in terms of a boundary layer.

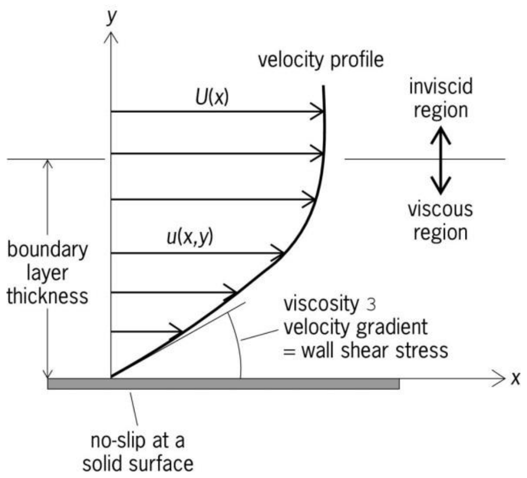


Figure 2: Velocity profile near wall. (McGraw-Hill Concise Encyclopedia of Physics , 2002)

In this boundary layer the effects of viscosity and friction are significant and are creating a retardation of the flow as seen in the velocity profile on the picture. Also the flow close to the wall does not depend on any free stream parameters, but only depends on the distance to the wall, the density of the fluid, viscosity and the wall shear stress(τ_w).

$$U = f(y, \rho, \mu, \tau_w) \qquad (1.8)$$

Dimensional analysis shows that:

$$u^+ = \frac{U}{u_t} = f\left(\frac{\rho u_t y}{\mu}\right) = f(y^+) \qquad (1.9)$$

Where $u_t = \left(\frac{t_w}{\rho}\right)^{\frac{1}{2}}$ is the friction velocity.

This formula is called the “Law of the wall”, and gives the definition of two very important parameters in boundary layer theory, u^+ and y^+.

As known and seen from the picture above, the fluid is stationary at the wall surface and that also means that the turbulent eddies in the flow also has stopped appearing very close to the wall. Without these eddies the Reynolds stresses are non-existent and the fluid is mainly affected by viscous shear stresses, and not the wall shear stress itself. This region is in a region that is very thin, at most the values of y^+ would be about 5. The shear stress can be assumed to be constant trough the layer and also equal to the wall shear stress.

$$\tau(y) = \mu\frac{\partial U}{\partial y} \cong \tau_w \rightarrow U = \frac{\tau_w y}{\mu} \qquad (1.10)$$

This can then be used to show the following relation by using algebra and the definition of u^+ and y^+.

$u^+ = y^+$ This relation holds true in the so-called linear sub-layer where $y^+<5$.

The next layer is the turbulent region a little further away from the wall and usually presents itself in the region of: $30 < y^+ < 500$.

This is the region where both turbulence and viscous effects are present, and very important but it is the Reynolds stresses that dominates.
The relationship between u^+ and y^+ is given by the log law, which is given as:

$$u^+ = \frac{1}{\kappa}\ln y^+ + C \qquad (1.11)$$

Values for the constants are found from measurements, and as an example the values for a smooth wall would be k=0.4 and C=5.5. These are values that are true for all kinds of turbulent streams flowing past a smooth wall at high Reynolds number.

Even further away from the wall the outer layer is present and is a region dominated by inertia forces.

In this region the law of the wake is explaining the relationship between velocity and wall distance.

These regions all contribute a lot to the resistance of the body moving in the fluid, and especially the turbulent layer (log-law layer) and the inner layer, so in the case of CFD calculations it is important to be able to capture the effects of these layers. This is done by having small enough cells close to wall when meshing, such that the changes in the velocity profile are modeled correctly.

It is desirable to have a sufficient number of grid points inside the boundary layer.
To estimate an appropriate initial cell size, y_{wall}, for Navier-Stokes simulations with turbulence included, the local Reynolds number based on a wall variable y^+ has to be calculated.

First the y^+ associated with the first node is calculated: $y^+ = \dfrac{\rho u_\tau y_{wall}}{\mu}$ (1.12)

Where $u_\tau = \sqrt{\dfrac{\tau_{wall}}{\rho}} = \sqrt{\dfrac{1}{2}(V_{ref})^2 C_f}$ (1.13)

The log law relationship is also given in the graph in figure 3.

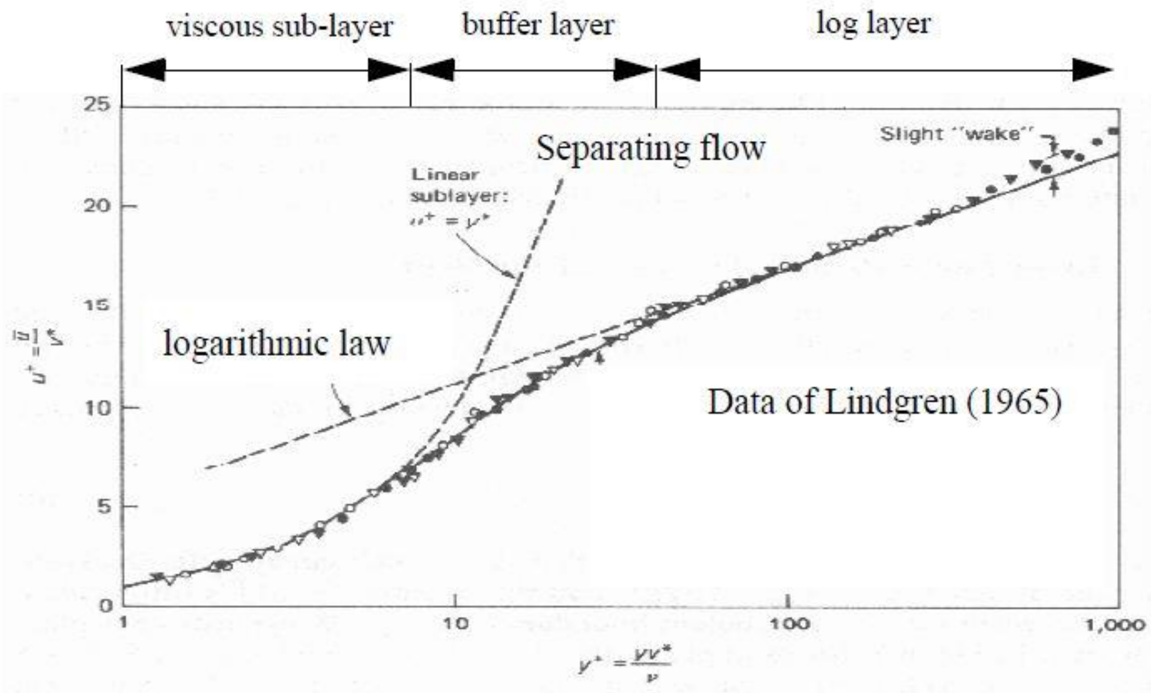


Figure 3: Law of the wall. Where $C^+ = 5$ and k=0.41 as given by experiments. (Numeca user manual)

By the graph it is shown that the log law does not apply for cases with large cases of separation or for y+ values beneath 5. Under y+=5 the relation is approximate u+=y+

One can see that all is related to the value of y^+ and this is a value that has to be chosen for different cases and sizes. But for a model scaled ship it is usual to use a value of about 50 as said in the manuals of FINE/marine and information provided by Eloise Croonenborghs.

1.7 Grid generation

As mentioned before a vital step in the CFD process is the discretization of the model, this is something that has to be done before the numerical solving of the equations.
The whole flow domain and all the surfaces on the boundaries have to be discretized to generate a volume mesh inside. There are two main types of grids used, and these are called structured and unstructured grids. The difference between these types can be seen in the picture below.

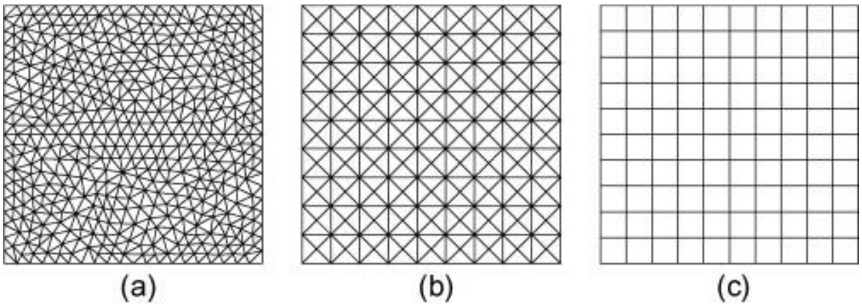


Figure 4: (a) is an unstructured grid, while (b) and (c) is both structured. (Blazek 2001)

As one can see the structured grids has cells with equal size, and are aligned very cleanly. The grid cells are quadrilaterals in 2D and hexahedra in 3D. Each grid point can be identified by the indices i, j, k and the corresponding Cartesian coordinates x_{ijk}, y_{ijk}, and z_{ijk}.
While the unstructured grid looks more random and the cells vary in shape and size.
Here the grid cells and grid points have no particular ordering, so cells next to each other cannot be identified by their indices. The unstructured grids usually consist of a mix of different cell shapes like both quadrilaterals and triangles in 2D, and hexahedra, tetrahedral, prisms and pyramids in 3D, so the boundary conditions can be modeled properly.
There are advantages and disadvantages with both of these methods.

So the main advantage of the structured grid is the simplicity it can provide to a problem because of the direct correlation between the Cartesian coordinate and the indices.
This means that it is very easy for the computer to access the neighboring grid point, just by adding or subtracting a value from the indices. So if the computer is working at cell (i, j, k) it can move to the next cell by just adding a value to one index, e.g. (i+1,j,k). This will greatly simplify the evaluation of the gradients and treatment of boundary conditions.

The downside of using structured grids is that it will be difficult to model complex structures with this method. This is because of the difficulty to model all kinds of geometry with the use of one sized and one geometry boxes. A solution to this is to divide the domain of interest into several smaller parts, which is less complex in topology and that can be meshed more easily. But this is something that will make the problem more complex, and some more computational time will be necessary.

So for the case of very complex geometries it is usually best to take an approach with the use of unstructured mesh. This will offer huge flexibility when it comes to model geometry. This is mainly due to the fact that triangular grids can be generated automatically, and independent of the complexity of the geometry. It can still be wise to apply rectangular or prismatic cells close to boundaries so the solution will be as accurate as possible. This is for example something that is done close to solid walls, so the effects of the boundary layer can be modeled correctly.

Another important advantage of the unstructured grids is the reduction of grid edges, faces and also grid points.

1.8 Spatial discretization

After the initial discretization of the domain, one has to apply this to be able to solve the flow problem given. This is done by soling the governing equations by the use of different numerical techniques, as mentioned earlier in chapter 1.1.

The three main methods are the finite difference method, finite volume method and the finite element method (FEM). In CFD today it is the finite volume method (FVM) that is mostly used.
The finite difference method is used in some cases, as for example for the direct numerical simulation of turbulence (DNS). But this method requires a structured mesh, so the applications are limited due to the fact that the geometries have to be rather simple.

Also FEM can be used to solve flow problems, but is mostly known for its application in structural analysis only.

So it is the FVM which is the most recognized method to use.
This method will utilize the conservation laws directly, and will discretize the governing equations by first dividing the domain into several arbitrary polyhedral control volumes, and

then achieves consistent expresses for the fluxes for different flow properties through the cell faces of adjacent control volumes. These control volumes also has different ways to be defined in regards to the grid.

The two main approaches are cell-centered scheme, and cell-vertex scheme.

For the cell-centered scheme all the flow properties are stored in the middle of the grid cell, and conveniently the control volumes are thus identical to the grid cells.

While for the cell-vertex scheme the flow properties are stored at the grid points, but additional volumes needs to be created. But the advantage is that boundary conditions are more easily applied since all the values at the boundary are known.

These two options are illustrated on the next picture.

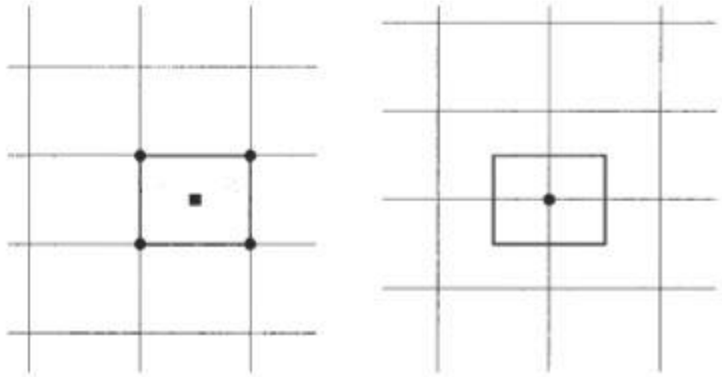


Figure 5: To the left is the cell-centered scheme, and to the right is the cell-vertex scheme. (Blazek 2001)

As for the FVM the main advantage of this method is that the discretization is being done in the physical space. There is no need for any transformations between different coordinate systems, which would be an issue in other methods like for example the finite difference method.

The FVM is also very flexible and can easily be adapted to both structured and unstructured grids, and is thus very good to use on complex problems.

There also exists several more discretization method, which can be used for special problems in very specific situations where they can outshine the more commonly used methods, but these are generally not used very frequently. Some examples would be the spectral element method and the grid-less method.

Underneath these basic forms of spatial discretization there exist various other numerical schemes to actually perform the discretization. More specific it is the different ways that it will handle the convective fluxes. This can be divided into two main categories, central schemes and upwind schemes.

Central scheme is as it name reflects, solely based on central averaging. The method involves taking an average of variable at the left and right side, so that the flux can be evaluated at the desired side of a control volume. While in the upwind scheme, the spatial discretization is more advanced, and consideration between upstream values and downstream effects has to be taken. This is something that is important when dealing with i.e. wave propagation.

There are several advantages and disadvantages but a main thing worth mentioning is the difference is computer resources required. The central scheme requires much less CPU time than the upwind scheme.

1.9 Temporal discretization

Since a separate discretization of time and space is the main way to go, the next thing to look at is the discretization in time. A separation of these two provides a large flexibility when it comes to the solution of the N-S equation, because different levels of approximations can be used which suits the problem at hand the best. There also exists methods where time and space is coupled, but it does not provide the same kind of flexibility.

When this method is used together with the governing equations it ends up with a system of coupled ordinary differential equations in time.

$$\frac{d(\Omega M\vec{W})}{dt} = -\vec{R} \qquad (1.15)$$

Ω is the volume of the control volume, $\vec{R}$ is the complete spatial discretization, including the residual term, which is a non-linear function of the conservative variable $\vec{W}$. Lastly, M represents the mass matrix.

By assuming a static grid, and by taking the volume and mass matrix outside of the time derivative, an approximation can be done of the time derivative by the use of this linear scheme (Blazek 2001):

$$\frac{\Omega M}{\Delta t}\Delta\vec{W}^{n} = -\frac{\beta}{1+\omega}\vec{R}^{n+1} - \frac{1-\beta}{1+\omega}\vec{R}^{n} + \frac{\omega\Omega \mathrm{M}}{(1+\omega)\Delta t}\Delta\vec{W}^{n-1} \qquad (1.16)$$

Δt represents the time step, and this time scheme is second order accurate if the condition

$$\beta = \frac{1}{2} + \omega \qquad (1.17)$$

is satisfied. If not, the time accuracy is only first-order accurate.

Numerical solutions schemes are usually called either explicit or implicit. The difference is that when a direct computation of the dependent variables can be made in terms of known quantities, the scheme is called explicit. While when the dependent variables are defined by sets of equations, and also a iterative technique is needed to obtain the solution, the method is implicit.

1.9.1 Explicit schemes

The value of β and ω decides if the scheme is explicit or implicit.

By setting both these values to 0, a basic explicit time-integration scheme is achieved. Here the time derivative is approximated by a forward difference, and the residual is evaluated at only the current time level.

1.9.2 Implicit scheme

Implicit time integration is achieved when setting $\beta \neq 0$, and an example of a much used scheme is

the three point implicit backward-difference scheme with $\beta = 1$ and $\omega = \frac{1}{2}$, which also qualifies for being second order accurate in time. This scheme is mostly used together with a dual time-stepping approach, which means that a steady state problem is would be solved at fictitious time steps at each actual physical time step, which will give certain advantages such as convergence acceleration and possibility to use local time stepping.

The main advantage of implicit time integration in comparison to explicit is that much larger time steps can be used, without damaging the stability of the time integration process.

The implicit scheme will, as the time step goes towards infinity, transform into a standard Newton's method, which allows quadratic convergence.

Implicit schemes are just much more robust and has a superior convergence speed in certain problems, i.e. turbulence modeling.

Also in CFD the governing equations are nonlinear, and the number of unknown variables is usually very large. This means that implicitly formulated equations like this are always solved using iterative techniques such as in the implicit scheme.

1.10 Added resistance

The calm water resistance of a ship in waves is lower than the mean resistance of a ship in waves.
The difference between the two is the added resistance due to waves. This resistance part is created by the loss of energy related to both reflections of the incident wave from the hull and the waves that are produced by the movement of the ship. The distribution of these two components is dependent on the parameter, λ/L, which is a ration between wave length and ship length. As one can see in figure 6, for wave lengths up to half of ship length, the main part of added resistance is due to the reflection of incident waves on the bow. While when ship length and wave length are about the same, the main contributor is energy loss due to ship motions. Added resistance also has its peak here. For long waves the added resistance will go towards zero. Since ship motions due to waves will be small.

It is second-order effects of the wave-body interaction that will create a mean drift force working against the forward motion of the ship. This force is the cause of the added resistance that a ship will face in the interaction with waves. For calculations of this resistance Faltinsen has developed a formula that can be used for small wave lengths, which corresponds to a wavelength under half of the ships length. The formula is only viable when the added resistance is dominated by diffraction forces on the hull. When the wavelengths get longer, the added resistance is mainly dominated by the relative motion between the ship and waves. The Faltinsen formula is shown below:

$$\overline{F_1} = \frac{\rho g}{2}\zeta_a^{\,2}\left(1+\frac{2\omega U}{g}\right)\int_{L_1}\left[\sin(\theta)\right]^2 n_1 dl \qquad (1.18)$$

This formula is only sensitive to the bluntness of the bow of the vessel, and will increase the more blunt the bow is, which is related to greater wave reflections. (Faltinsen 1990)

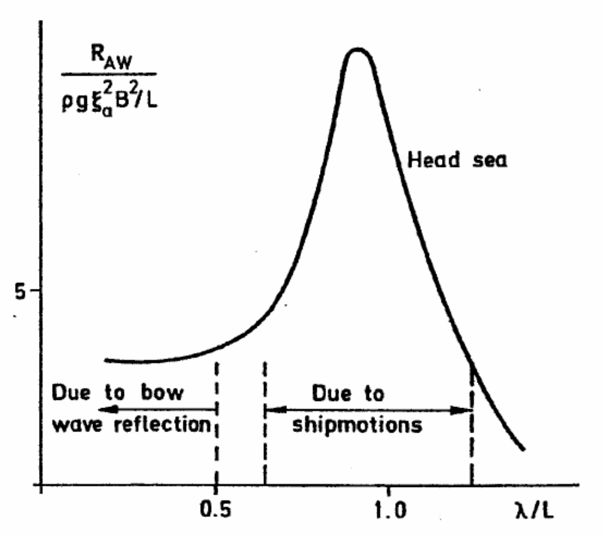


Figure 6: Variation of added resistance coefficient, Caw, dependent on wave length at a moderate Froude number (Faltinsen 1990)

Added resistance is important to know about since most of the times a ship will be facing waves, and thus the power needed by the ship is larger than for calm water cases. It is important to take into account this extra resistance. The best thing would be if one had a reliable, effective and accurate way of predicting these forces instead of using estimates, which is common today. This is the reasoning behind the studies around added resistance today. To be able to know more about the conditions a ship will face, before it is specified and build.

There are different tools to obtain the added resistance, and these tools are mainly: experiments, computational fluid dynamics or empirical formulas.
Experiments have been commonly used and seen as providing trustworthy results to a limit. There are several sources of error, and small mistakes in setup or scaling can contaminate results.
Experiments also prove to be expensive both in time and costs.
In comparison the empirical formulas are both cheap and fast, but there are a large amount of restrictions on the use. They are very limited to the assumptions that they are based on, and often the accuracy can be very poor. Another issue arises because one only uses the main dimensions of a ship, so detail analysis of design is not possible.

There are not big amounts of literature concerning the subject of added resistance prediction with CFD, and especially not with proper validation of the accuracy of the results. There have been some CFD workshops in the past, but resistance in waves is something that has not been looked at until the Gothenburg workshop of 2010. Proper accuracy is hard to achieve because of many factors. Some of these factors that may create problems with accuracy in added resistance prediction with CFD are, different ways of modeling and generate waves, domain sizes, time step, grid refinement and much more. There are also big differences in results when it comes which turbulence models are chosen or which properties are used in the solvers.

2 Main dimensions KVLCC2 hull

	Units	Ship	Model
Scale		1:1	1:58
Lpp	[m]	320	5.5172
Lwl	[m]	325.5	5.6121
B	[m]	58	1
T	[m]	20.8	0.3586
D	[m]	30	0.5172
Cw	[-]	0.9077	0.9077
Cb	[-]	0.8098	0.8098
V	[m^3]	312622	1.6023

Table 1: Main dimensions of KVLCC2

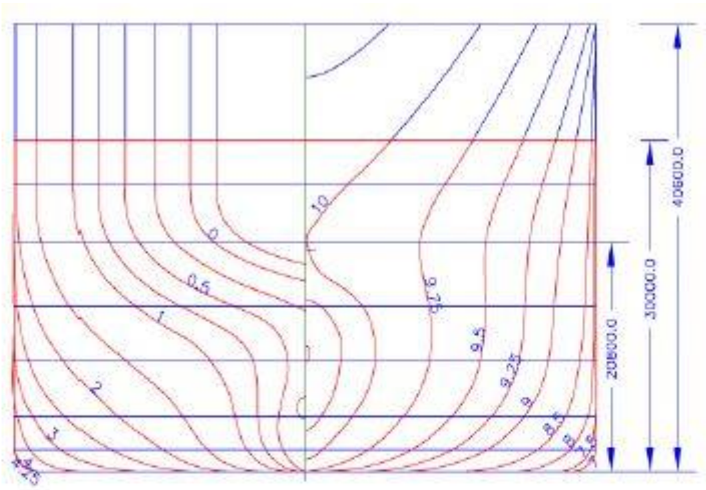


Figure 7: Lines plan of the hull, here with extra height drawn in.

3 Setup HEXPRESS

The first thing to do when setting up a CFD project is to create the mesh. In FINE/Marine this is done in the HEXPRESS package. The mesh generation is a crucial part in any CFD calculations, and will be the main reason for good or bad results. Settings in the mesh creation have to be chosen wisely.

3.1 Parasolid model

I have been provided a model of the ship in question, the KVLCC2 hull. This is a parasolid model, which is scaled 1:1000. I therefore made the mesh based on this hull size, and then at the end scaled the mesh up by a factor of 1000/58=17 to get the appropriate scaling, such that comparisons can be done with earlier model tests and CFD calculations.

The model is oriented with positive x being in the direction of the movement, and positive z upwards.

This model provided has already been cleaned by Eloise Croonenborghs, so that it is ready to be meshed without huge problems. There can be considerably much work regarding cleaning a cad model, so that it can be meshed.

3.2 Domain

A box around the model has to be created that will be the domain of the calculations. This is the basis of the fluid flow around the hull, and it should be chosen wisely. The size of the domain has a big effect on the reliability of the results.

There seems to be some standards of what is considered an appropriate domain size, and I have used the ones provided by the documentations that come with HEXPRESS from Numeca.

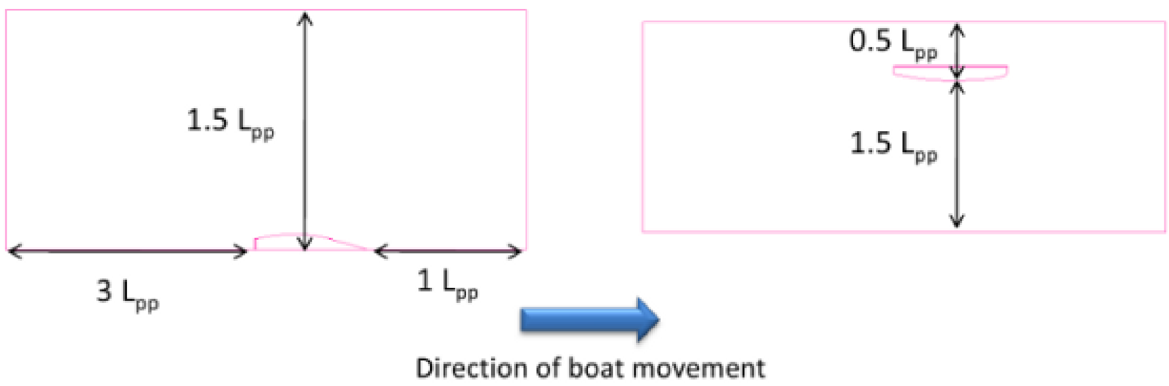


Figure 8: Domain size

To get a satisfactory mesh that will recognize the hull inside it, some faceting settings have to be applied. Here I also follow the tips that are given by Numeca documentations and tutorials.

Faceting settings	
Minimum length	LPP*10^-3
Maximum length	LPP
Curve chordal tolerance	10^-3
Surface plane tolerance	10^-3
Curve and surface resolution	3

Table 2: Domain settings

Before making the mesh, the domain can be modified by splitting or merging both patches and edges. It can be smart to merge and split patches so you have divided the hull such that you can set the appropriate mesh refinements at the right places. The deck doesn’t need the same refinement as for example the bottom of the hull.

The initial domain was split up in many inconvenient pieces, with some of them being very badly shaped and being very small, which will create problems with the mesh later.
Some work had to be done and the result was patches with the following names:

- Top (External)
- Bottom (External)
- Side (External)
- Mirror (Mirror)
- Inlet (External)
- Outlet (External)
- Upper edge
- Bottom hull
- Side hull
- Bow
- Stern
- Transom
- Deck

3.3 Free surface

An extra refinement of the mesh is needed around the free surface to capture the waves that are generated by the hull, or by the regular waves that is imposed in the domain.

One surface is made over the whole domain with cells with a high aspect ratio, since the waves a primarily going in one direction.

Another surface is made as a triangle which purpose is to cover the area where the kelvin waves will appear. This surface is made to have cells with a smaller aspect ratio, because the kelvin waves will propagate in both x and y direction.

3.4 Initial mesh

The basis of the mesh is first made, and the usual practice here is to divide a domain which corresponds to half of a ship hull in to about 1000 cells.
In my case this is 960 cells, which is 6x8x20.

3.5 Mesh Adaption

In the mesh adaption step one has to specify the needed refinements on the different patches on the hull. Here it is specified how big the cells should be, and how many times the initial cell should be divided. Also one can specify if extra curvature refinement is needed, and how big the aspect ratio should be. This is the most important step in the mesh generation, and will decide the size and quality of the mesh.

The front and aft of the ship:
Includes stern, bow and transom.

Refinement: 8 times
Diffusion: depth: 4 cells

Deck and upper edge:
Refinement: 3 times
Diffusion depth: 4 cells

Side hull:
Refinement: 6 times
Diffusion depth: 4 cells
Curve refinement

Bottom hull:
Refinement: 6 times
Diffusion depth: 4 cells

Free surface:
Maximum refinements: 7 times
Diffusion depth: 4 cells
Target cell size: (0.9 0.9 0.00032)
Aspect ratio: 128

Kelvin surface:

Maximum refinements: 8 times

Diffusion depth: 4 cells

Target cell size: (0.1 0.1 0.00032)

Aspect ratio 64

Edge refinement at transom:

Refinements: 8 times

Curve refinement

The flow will be more disrupted at the bow and at the stern, due to the sudden changes in geometry that the flow will encounter when it goes by. That is the reason for the smaller bigger refinement at bow and stern, compared to the middle of the ship at bottom and sides.

The deck and upper edge are not going to have any contact with water, which is clearly the main reason for resistance in comparison on to air. So these patches don't need any big refinements.

Due to the bilge at the side hull, some extra curvature refinement has been added at this patch, so the bilge will be captured properly.

Some extra edge refinement has also been set at the edge that is under water at the transom. Since this edge will abrupt the flow, and will contribute much distortions in the flow wake.

The idea behind the surface refinements is that the most important one is the surface closest to the ship, the kelvin surface. This will be the surface with most cells, and lowest aspect ratio. The target cell sizes are something that is listed as best practice tips in the Numeca documentation.

This is to take the cell size in z direction to be 1/1000 of the ships LPP. Mesh isn't scaled yet, so the LPP in question here is the one of the model of 1:1000.

3.6 Snapping and optimization

After the mesh adaption process, the mesh is coarse in relation to the ship geometry.
All the cells are cubes, which is not properly aligned with the ship hull, and therefore doesn't capture the hull geometry properly. The snapping process will take care of this. It will fit the mesh around the hull in the right manner.

After this process is done, a lot of the cells tend to distort in a bad way. An example is creating negative sized cells. This means that they are compressed so much that they fold in on themselves.
This will greatly reduce the robustness of the solution, so this can't be present in the final mesh.

The optimization process will take care of these problems. It will run an algorithm that repairs the mesh until no such bad cells are left. No user input was used in these processes.

3.7 Viscous layers

The final step of the meshing process is inserting the viscous layers.
The flow over a plate creates a boundary layer close to the wall, because of the no-slip condition.
It is needed to make a finer mesh close to the wall, such that the effect of the no-slip condition can be accounted for. This has a great effect on the final results.

The program will do the calculations for what size the cell closest to the wall should be, and how many cells should be inserted. One only has to give some input about the y+ and the Reynolds number. Here also the input is in the 1:1000 scaled, due to the scaling that will be done later.

First layer thickness	0.000495042
Number of layers	3

Table 3: Viscous layer settings

The only patches where viscous layers has been imposed is the ones that is going to be under water, so the deck and the upper edge is not in need of extra layers.

3.8 Importing mesh to FINE/Marine

Last step is to assign the appropriate boundary conditions (external, mirror and solid), and scale the mesh by 1000/58. The scale is now 1:58 and the calculation set up can be done. Before one goes further, one should just check if the mesh quality is satisfactory.

4 FINE/Marine calculation setup

The calculations done are going to be compared to model test results and other CFD results that have already been done on this very same hull, the KVLCC2. The datasheet from the previous model test are provided, and it contains data about the hull and other test conditions.

4.1 Calm water resistance

The first calculation done was the calm water resistance test, which is the hull going forward in a constant speed in water without any waves present. This was first done with only one degree of freedom, which is the imposed motion in the motion direction, and all other degrees of freedom constrained.
These results are without value, but were done to just test if the mesh would work properly.

The fact is that the sinkage and trim that the hull will be exposed to is contributing a great deal to the resistance of the hull. So the next simulation done, was one were the motion in pitch and heave were going to be solved. This will greatly affect the simulation time, but is crucial for a good result.

Main parameters that are being used for this calm water test are:

Steady computation

Multi fluid

Water: Dynamic viscosity = 0.0010987 Pas
Density = 998.82 kg/m^3

Air: Dynamic viscosity = 1.85*10^-5 Pas
Density = 1.2 kg/m^3

Turbulence model – K-omega (SST-menter)

Mass – 1600 kg

KG – 0.321 meter

LCG – 2.95 meter (from AP)

Time step – 0.1 seconds

4.2 Resistance in waves

After the calm water resistance computations were done, it was time to do some preliminary calculations with the hull in waves. The first step then is to be able to create waves in FINE/marine without any hull present. This is fairly easy done. The reason for doing this is so the mesh's ability to model the wave is good enough.

The same domain was made without any hull, and the appropriate mesh refinements was done according to Numeca best practice OpenOffice calc sheet, appendix B.

Since computations are going to be compared to model tests and other CFD tests, the same wave data was used as in Guo Bingjie's thesis. Following wave data was used to create the stokes wave.

- **Wave period** – 1.8 second
- **Wave height** – 0.15 meter
- **Wave velocity** – 2.81 meters per second
- **Wave length** - 5.062 meter
- **Water depth** – 5.5172 meter

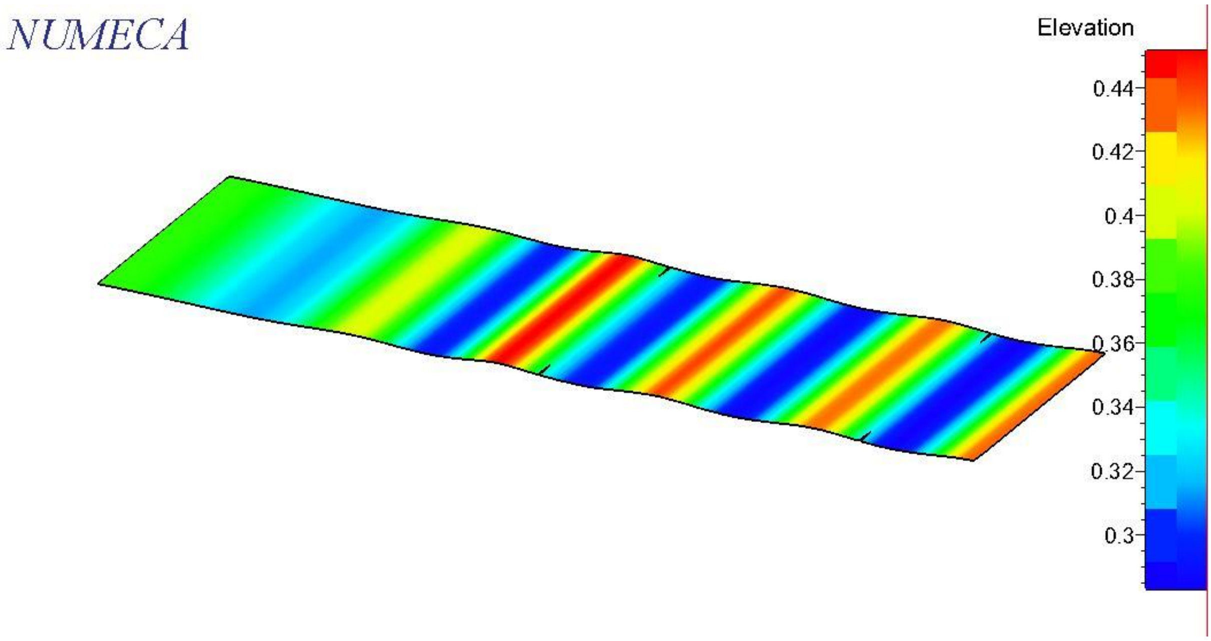


Figure 9: Stokes wave generated without hull present. T = 1.8 [s] H = 0.15 [m] λ = 5.062 [m]

When the mesh was good enough for modeling the waves properly, the hull was introduced in the domain. Most of the parameters are as before, with the wave settings, and other settings. The major difference from the calm water calculation is that it's an unsteady computation, and the time step is much smaller. Also, when dealing with pitch motion one needs to have information about the inertia matrix. The inertia for the pitch motion was approximated by the formula:

$$I = mr^2 = m \cdot (0.25 \cdot Lpp)^2 \qquad (4.1)$$

At first a calculation with no solved degrees of freedom was done, to test if it worked.
Second a calculation with one degree of freedom solved, heave.
Thirdly, a calculation with 2 solved motions, heave and pitch, was done.

The time steps used for these calculations are something that is based on a spreadsheet given by Numeca. They are based on the encounter frequency between ship and waves, and also best practices which they know from experience. The tip from Numeca is that for a ship without solved motions, one has to use 80 times steps per wave period. And when using solved motions, it has to be over 200 time steps per wave period.

	Time steps per period	**Delta t**
No solved motion	80	0.016 [s]
Solved motion	200	0.0065 [s]

Table 4: Time steps used

5 Results and comparisons

5.1 Calm water resistance results

While the calculations were running, one can monitor the results and look for the point where the desired results have converged, as seen in figure 10.

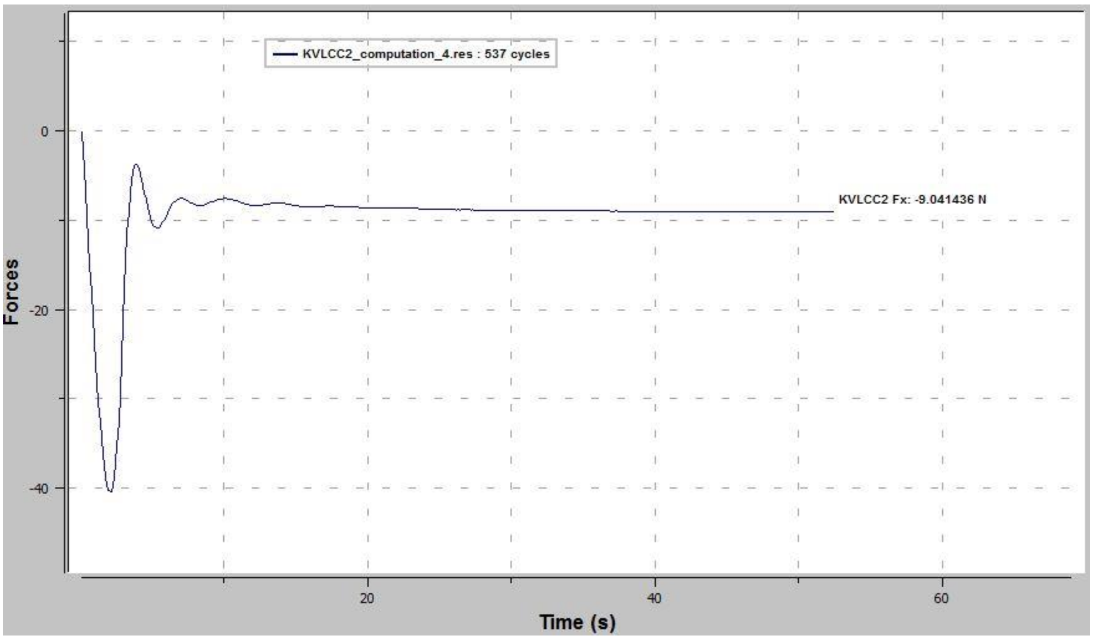


Figure 10: Graph showing the convergence of Fx

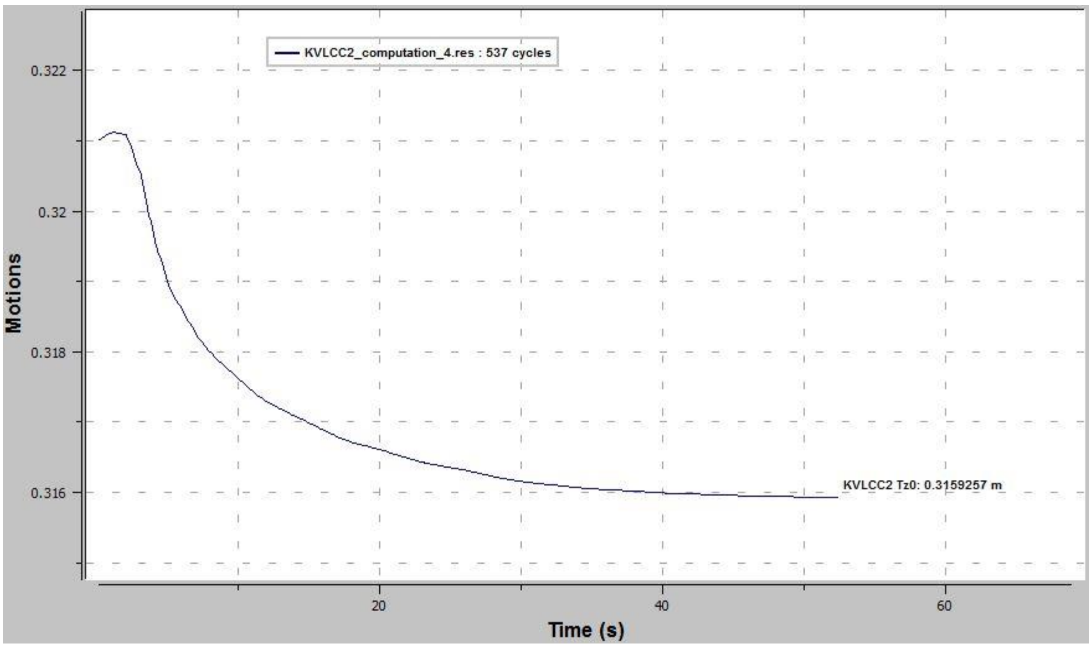


Figure 11: Graph showing convergence of heave motion

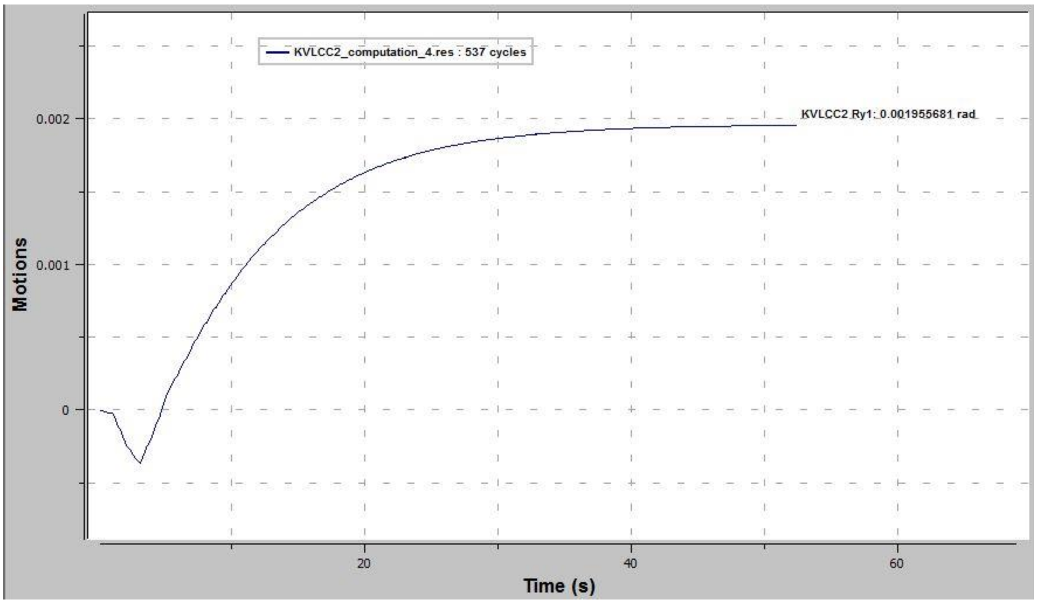


Figure 12: Graph showing convergence of pitch motion

The force in x-direction in the first picture is the force that is acting on half of the hull, so it needs to be multiplied by two. The value of heave motion is the new position of the gravity center.

The correct values are collected in the table below, together with the comparison done by model tests.

Values from model are taken from Guo Bingjie's PhD thesis.

Speed [m/s]	1.045			
Froude	0.1420			
Model tests		CFD calc		Difference
Sinkage [mm]	-6.419	Sinkage [mm]	-5.1	25.86 %
Pitch [degrees]	-0.126	Pitch [degrees]	-0.112052267	12.45 %
Resistance[N]	18.2	Resistance [N]	18.082	0.65 %

Table 5: Results from calculation of speed 1.045 m/s

As we seen here, the results are quite alike, especially the resistance, which is under 1% from the model test. When looking at the sinkage and trim, there is observed some bigger discrepancies, and there are some reasons that can explain those.

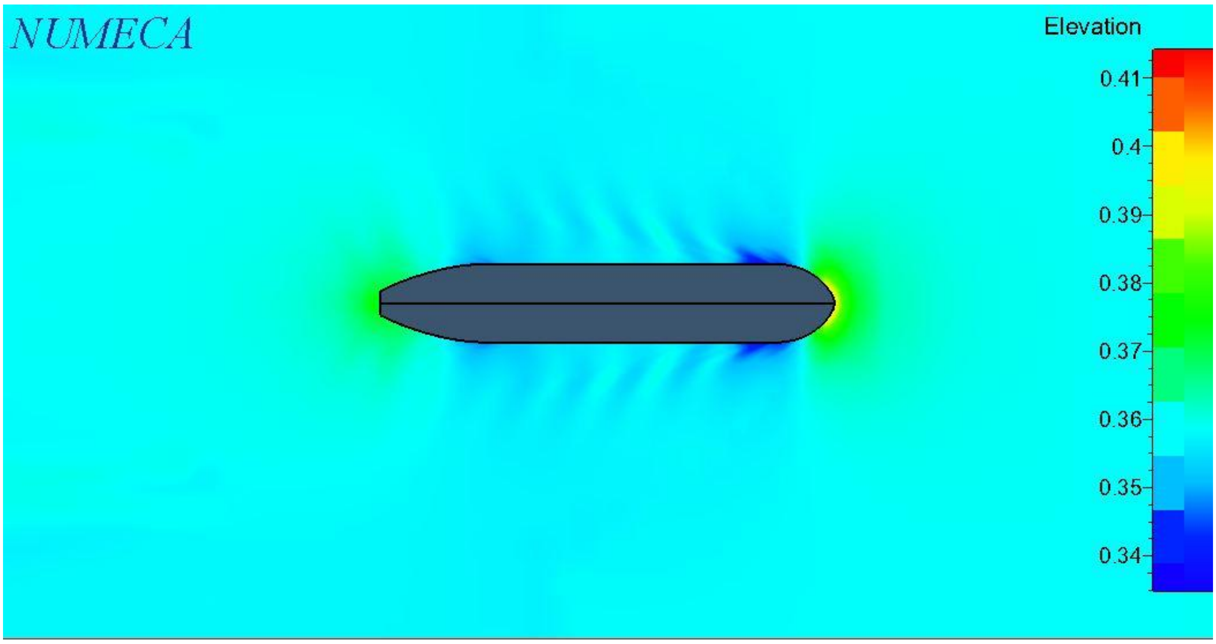


Figure 13: Wave s generated by hull in calm water, for Froude number 0.142

Figure 14 shows how the free surface is elevated along the hull. There is a peak just in front of the bow as expected and a drop right after that. The kelvin pattern is also very clear in this picture, and shows why there was a need for extra refinement in this area, as discussed earlier.

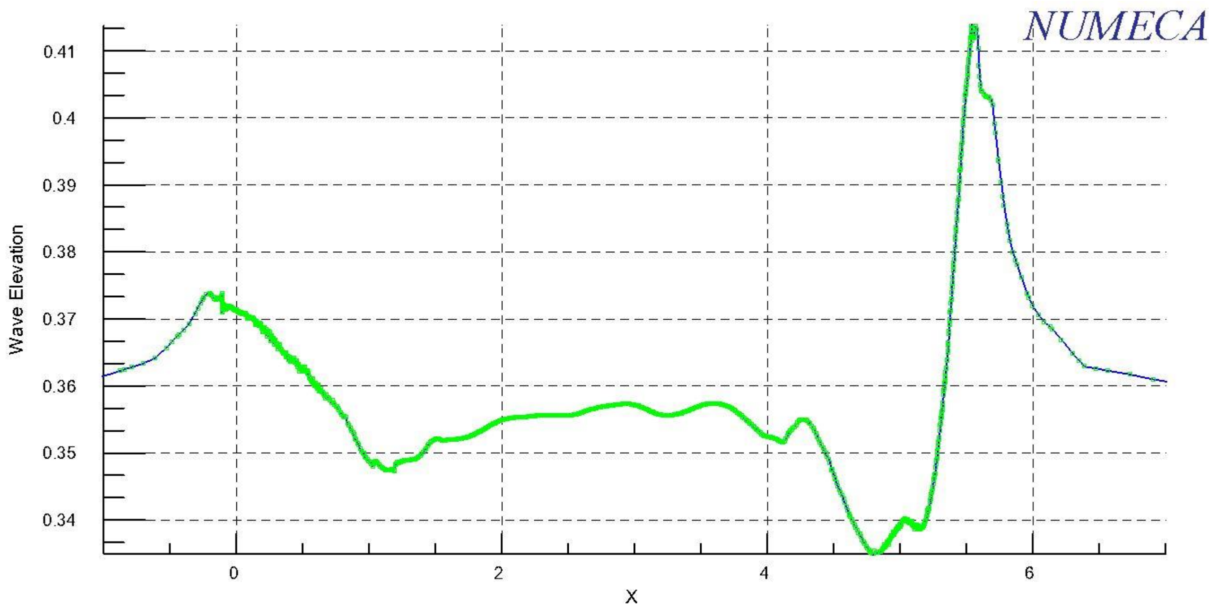


Figure 14: Plot of surface elevation along the hull, where 0 is the AP

5.2 Resistance in waves

Some graphs for showing the results for the simulations done with the hull in a wave with length of 5.062 meters.

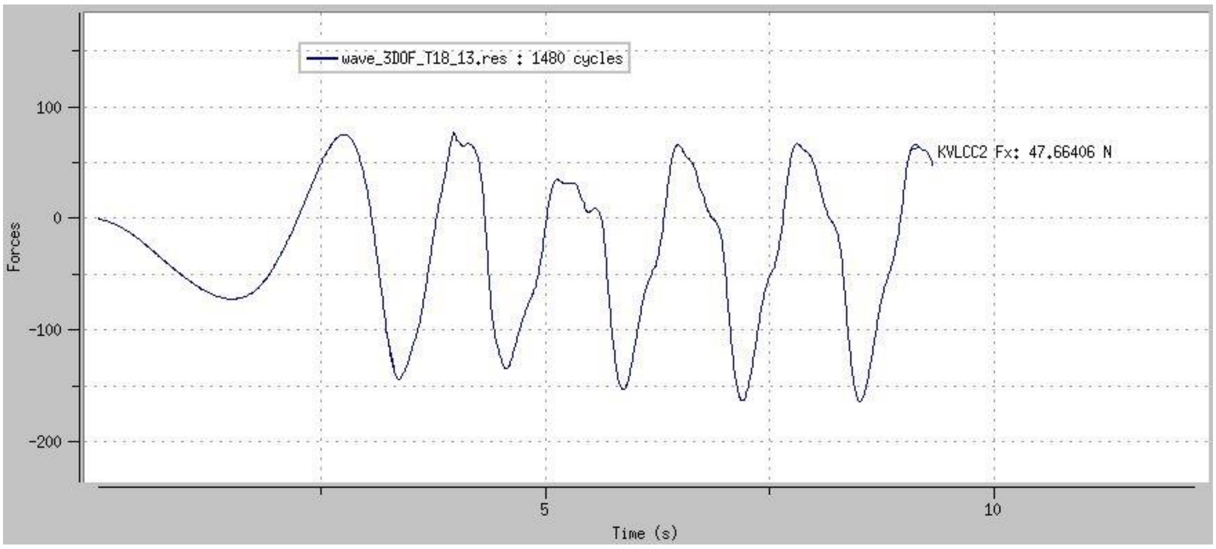


Figure 15: Evolution of the drag force through the time steps of the simulation. Wave length = 5.062 [m] and Wave period = 1.8 [s]

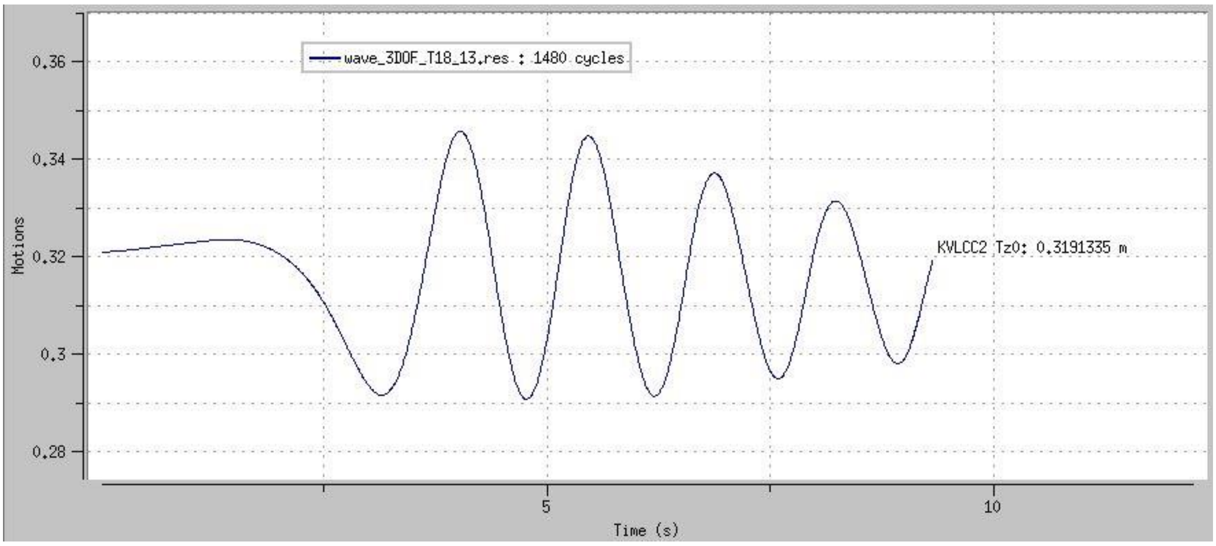


Figure 16: How mass center moved in z-direction with time.

As seen in the pictures the simulation only ran for a time corresponding to about 10 seconds of physical time. This is way too low, and accurate results cannot be expected. This is also seen in both the pictures; the oscillation has not stabilized and converged to a mean value. This is due to the time running out in the end, as discussed earlier.

There also was only time to run just one wave, and the wave chosen was the one with wavelength 5.062 meters. The preliminary results acquired from this short run are presented below.

		Calm water resistance	18.082	N	
λ/Lpp	**Wave amp. [m]**	**Wave period [s]**	**Mean resistance [N]**	**Raw [N]**	**Caw [-]**
0.917	0.075	1.8	97.414	79.332	7.941

Table 6: Results of the resistance test in a wave in FINE/Marine.

6 STAR-CCM+

To make use of the parasolid model of the ship provided, it had to first be opened in FINE/Marine, and then exported into a .STL file which Star-CCM+ was able to open. Again it had to be scaled to the appropriate model size, and then it was ready to be prepared for meshing.

At first the patches was completely random and excessive, so the first step was to organize the patches in a logical manner. Patches that was corresponding to certain areas on the hull was created, and were given the appropriate names. They were made as following.

- Bow
- Upper ledge (over waterline)
- Transom
- Stern
- Deck
- Bottom hull
- Side hull

Also patches for the boundaries were created, so that correct boundary conditions could be specified for each boundary.

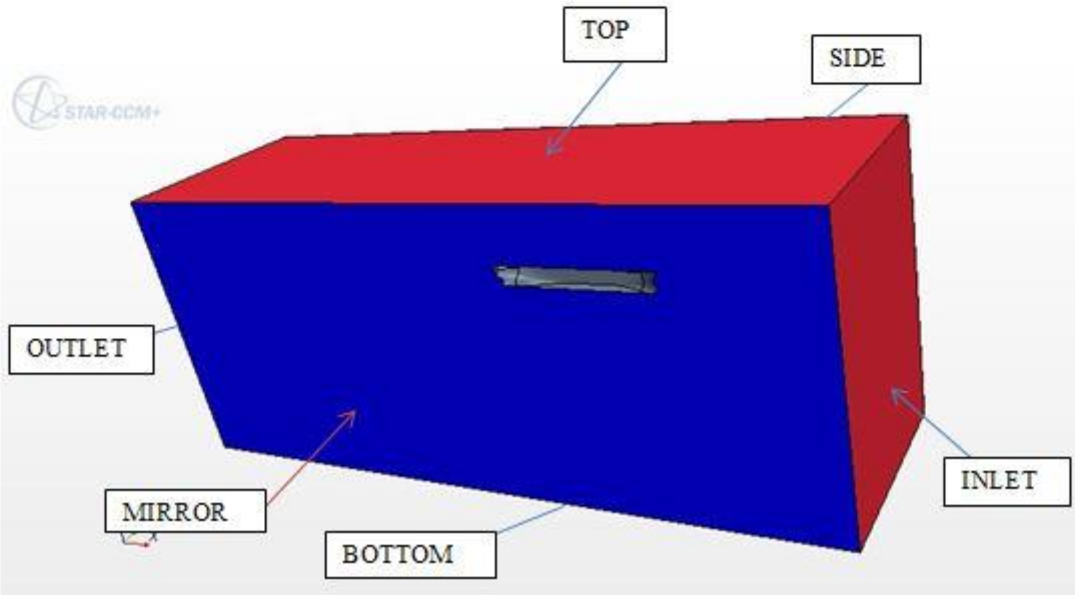


Figure 17: Boundary conditions

6.1 Calm water resistance

6.1.1 Mesh

When setting up the model in Star-CCM+ I followed some tutorials, given to me by Eloise Croonenborghs, which are provided by Star-CCM+ in their online courses.
These tutorials together with the guides provided by NUMECA, gave me tips on how to choose parameters and models, which would be suitable to resistance tests of ship hulls.

The first step was to make a proper mesh, which is quite easily done in Star-CCM+.
One just chooses the appropriate models, and after that basically just chooses the size of the cells in different directions. Extra refinement of cells was also applied around the free surface, and around the hull itself. The mesh refinement was chosen to be close to how it was done in FINE/Marine, but identically was impossible because of the completely different way the mesh specifications were done in these two codes.

The result of the mesh is shown in the picture below.

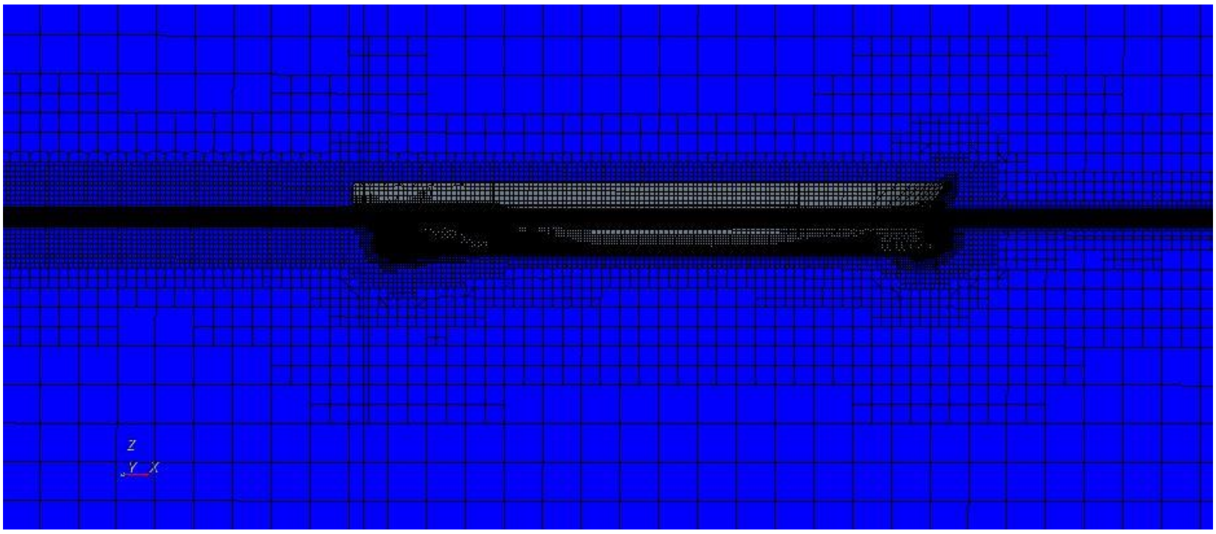


Figure 18: Mesh in calm water resistance test with 2 000 000 cells.

One can see in the picture that the mesh is very refined at the free surface compared to other places, and also around the hull and in the wake.
A model to create a prism layer mesh close to solid walls is also selected to be able to capture boundary layer effects better.

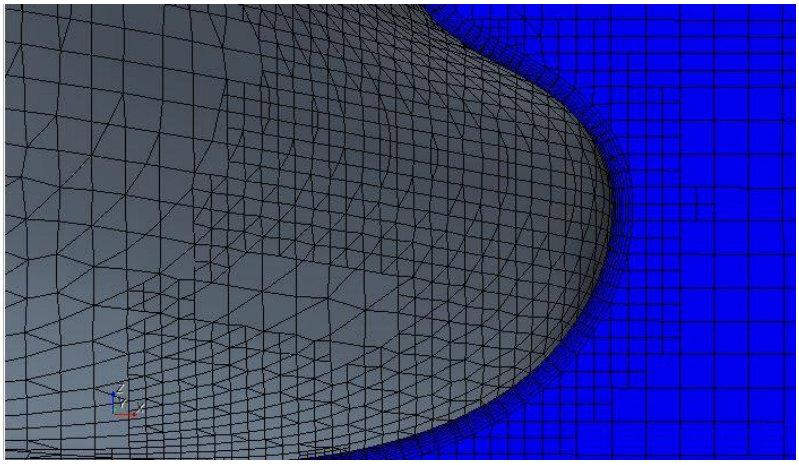


Figure 19: Prism layer mesh close to solid walls.

6.1.2 Physics models

The next step is to choose all the models that will create all the physical conditions that will be apart in modeling the problem. These are the models chosen:

- Three dimensional
- VOF waves
- Multi-phase Mixture
- Implicit unsteady
- Turbulent (K-omega model)
- Gravity
- Volume of Fluid (VOF)
- Segregated flow

Some of these models are very basic and logical to choose, i.e. gravity, three dimensional and multi-phase mixture.

VOF is an interface capturing method where the positioning of each fluid phase is decided by assigning a volume fraction of either 0 or 1 in each cell. The interface region is then recognized as the place where this volume fraction has a change. So to specify the free surface in the CFD codes, one has to specify the place where the volume fraction of the fluid is 0.5.

VOF waves are chosen even if there are no waves present, but the calm water is actually specified as a flat wave.

Segregated flow is the normal way to go in problems like this. The alternative would be a coupled flow, but this not something that is used other in flows where there are high density fluctuations, i.e. supersonic flows with shocks. Segregated is also a lot cheaper to solve in terms of time and computational power.

Other than this one has to specify fluid properties like densities, viscosities and also ships forward speed. These are all the same as specified earlier.

The correct initial conditions also has to be set, this is done in a way where you assign the pressure field, velocity field and the volume fraction to be decided from the wave specified. In this case this is a flat wave.

6.1.3 Boundary conditions

There are some conditions that needs be specified at each of the boundaries presented in the earlier picture. The mirror boundary needs to be specified as a mirror.
While the bottom, side, inlet and top boundary is going to be specified as velocity inlets. Further, it is necessary to specify that velocity and volume fractions at these boundaries have to follow the field functions specified by the flat wave. And care has to be taken to assigning the flow directions at all the velocity inlets so that there is no conflict, and a curved free surface will appear.

At last the outlet is going to be assigned as a pressure outlet. And again pressure and velocity has to be related to the field function assigned by the flat wave.

6.1.4 Time step

The last thing to do is to setup the solver. This involves choosing time step, stopping criteria and convergence criteria. A time step of 0.05 seconds was chosen and maximum physical time at 60 second.

6.1.5 Physical parameters of hull

At last the parameters of the hull have to be specified. These are things like center of gravity, mass of hull and the inertia matrix. All these parameters are the same as given for FINE/Marine.

6.1.6 Results

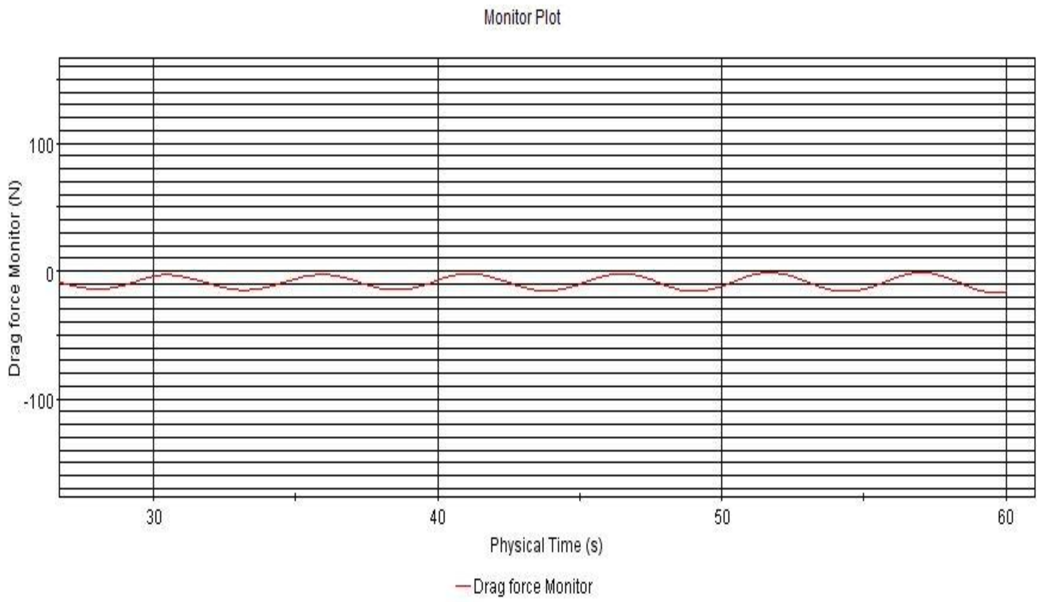


Figure 22: Drag force plot, from 25 seconds to 60 seconds.

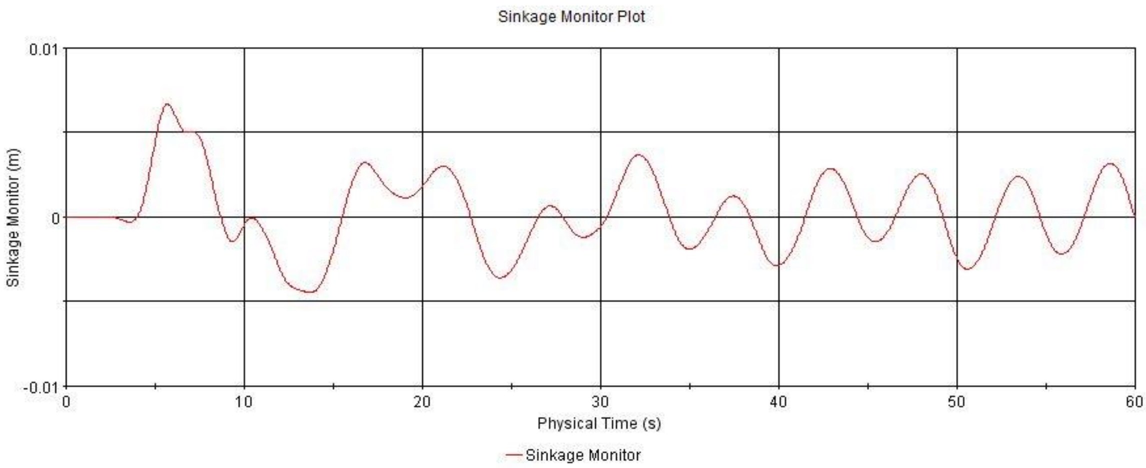


Figure 21: Plot of the sinkage change with time.

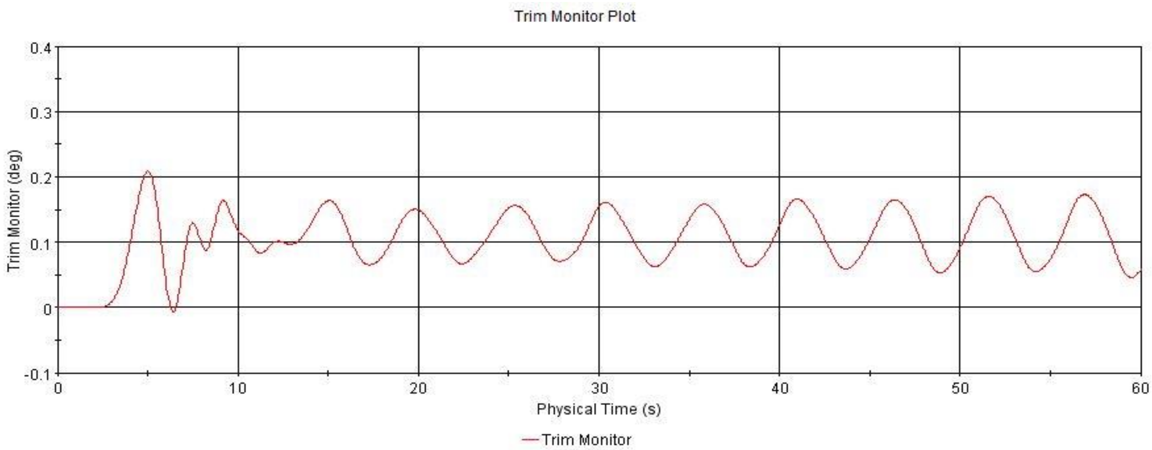


Figure 20: Plot of the change in trim with time.

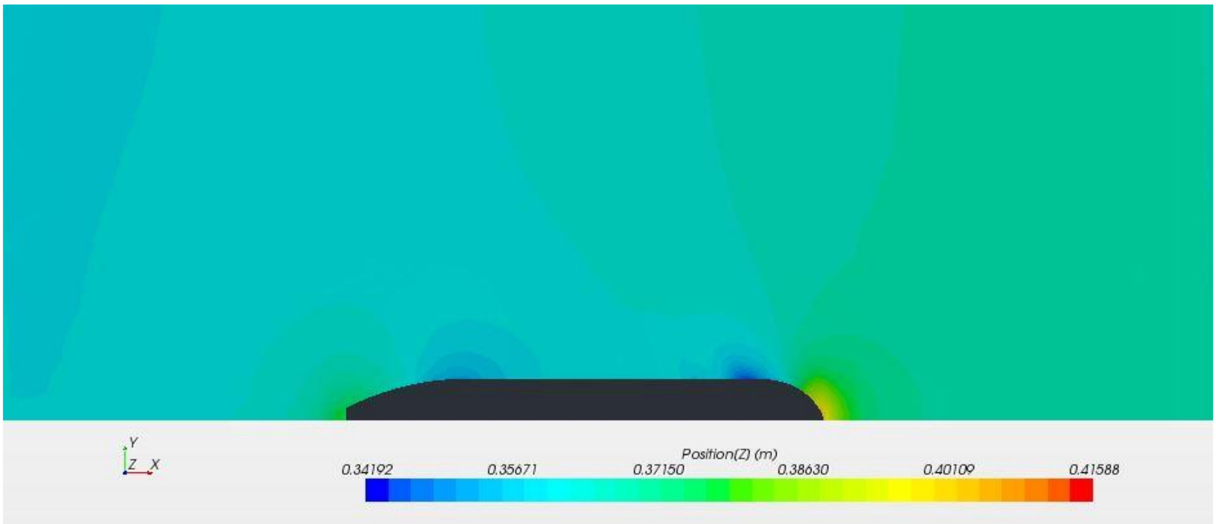


Figure 23: Contour plot of the surface elevation around the hull

The drag force plot was taken into Excel and a mean value was found to be 9.03 N, which is 18.12 N for the whole hull.

The mean value in the trim degree plot was 0.108 degrees.

The sinkage plot varies a lot, and is not converging towards any value. But by the use of some averaging it seems to be oscillating around a value of 0.14 millimeters.

Speed [m/s]	1.045			
Froude	0.1420			
Model tests		CFD calc		Difference
Sinkage [mm]	-6.419	Sinkage [mm]	-0.14	4485.00 %
Pitch [degrees]	-0.126	Pitch [degrees]	-0.108	16.67 %
Resistance[N]	18.2	Resistance [N]	18.12	0.44 %

Table 7: Table showing CFD results compared to model tests.

As seen in the table, the resistance is very close to what it should be, and only differs with about 0.44%. The pitch degree is also somewhat close and differs from the model tests with 16.67%.

While the sinkage is completely wrong and is far from the desired value.

6.2 Resistance in waves

6.2.1 Waves

When doing the simulation in waves, there are a couple of things that has to be changed in comparison to the calm water simulation. Firstly one has to specify the wave, and give it the wanted properties like wave height, wave period and wave length. In this thesis different simulations were done with changing wave parameters. The waves used are as following, and the same as used in the thesis of Guo Bingjie.

LPP/Wave length	Wave length [m]	Wave period [s]	Wave height [m]	Time step [s]
2.453	2.249	1.2	0.05	0.00385
1.667	3.310	1.456	0.1	0.00498
1.571	3.512	1.499	0.15	0.00518
1.224	4.508	1.699	0.15	0.00609
1.09	5.062	1.8	0.15	0.00656
0.909	6.070	1.972	0.1	0.00736
0.625	8.828	2.378	0.1	0.00927

Table 8: List of different waves used

Another important change is that with waves present even more refinement around the free surface is required such that the waves can be captured properly. Here the same open office document was used as for FINE/Marine, where optimal refinement in x,y and z direction was obtained based on wave and ship parameters.

Also the optimal time step for ship in waves with solved motions were calculated and applied to each different case. The different time steps used is also listed in the table above.

6.2.2 Turbulence model

In the calm water resistance test the turbulence model used was the well-known and much used k-omega model. But when doing so for these cases with waves, the results did not look good and the free surface looked unphysical. This may be because of what is mentioned before about this model being very sensitive to the free stream parameters of the turbulence model, and also the initial conditions and boundary conditions for the turbulence. So by looking at how Star-CCM+ solved similar simulations in their tutorials, the spalart-allmaras model was instead used.

6.2.3 Results

After all the simulations were done a table was made which summaries the results obtained. The table consists of each wave, and its resistances connected with it.

		Calm water resistance	18.064	N	
λ/LPP	Wave amp. [m]	Wave period [s]	Mean resistance [N]	Raw [N]	Caw [-]
0.408	0.025	T12	18.795	0.731	0.658
0.600	0.05	T145	24.986	6.921	1.559
0.637	0.075	T14	42.921	24.856	2.488
0.817	0.075	T16	52.854	34.789	3.482
0.917	0.075	T18	82.492	64.427	6.449
1.100	0.05	T19	43.973	25.908	5.835
1.600	0.05	T23	24.764	6.700	1.509

Table 9: List of all the waves and the forces related to it.

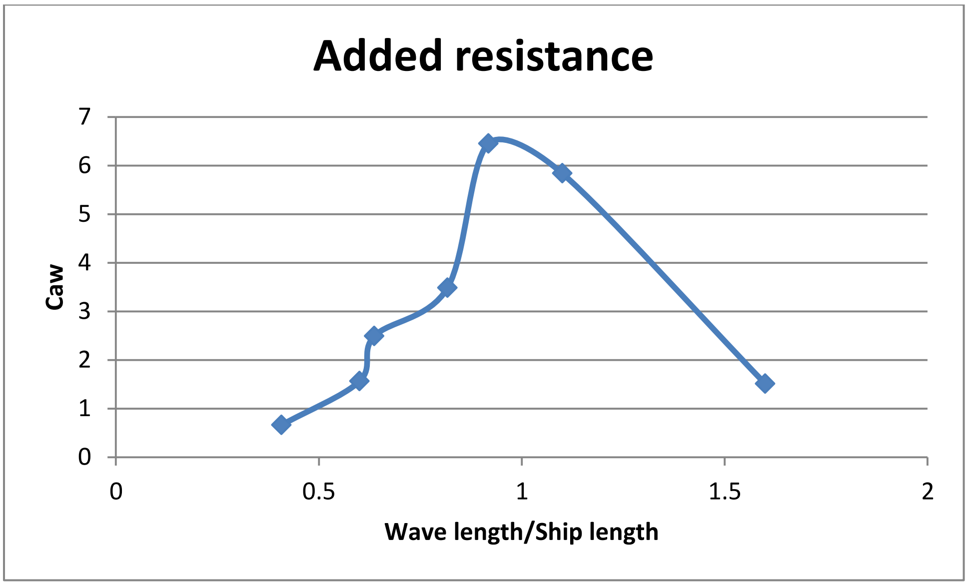


Figure 24: Plot of the previous table. This shows the relation between wave length and added resistance coefficient.

By the look of plot made, it is similar to the general graph presented earlier which shows how added resistance varies with different wave lengths. But it is clearly not optimal, and a lot of improvements can thus be done to the simulation.

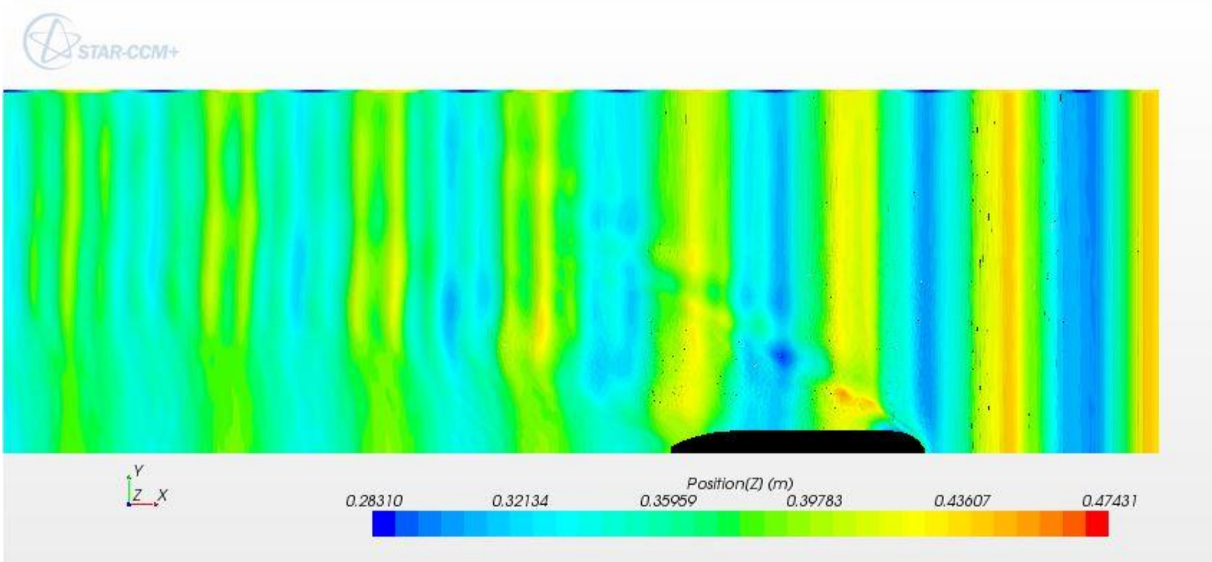


Figure 25: Free surface plot with wave present. Wavelength= 4.508 and Froude=0.1420

On the picture above one can see the surface plot showing the z-position of the surface particles.

It is also seen that an even further refinement of the free surface could have been better.

Below a couple of examples of some post-processing is shown. Another wave elevation plot and a pressure plot, respectively.

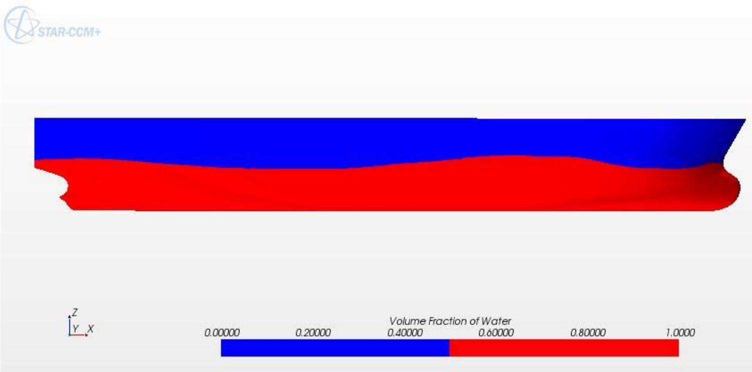


Figure 26: Illustration of the water level at the side of the hull.

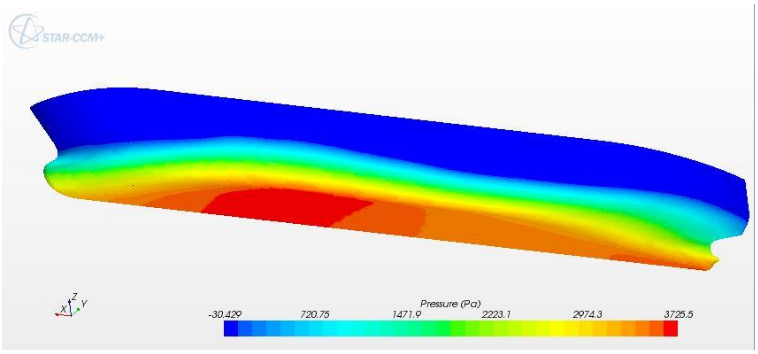


Figure 27: Pressure plot on the hull when passing over a wave.

7 Conclusion

Due to the time limitation and all of the problems that has occurred, way too few and inaccurate results are shown here. No real comparison can be made between the two codes, mostly due to the fact that in FINE/Marine only one wave case has been simulated. If we compare this wave case with the same one in Star-CCM+ we can see that the results at least are in the vicinity of each other.

Wave length	**5.062**	**meters**
	Added resistance	
FINE/Marine	79.332	N
Star-CCM+	64.427	N
Difference	23 %	

Table 10: Difference between FINE/Marine and Star-CCM+

FINE/Marine is over-predicting Star-CCM+ with about 23%, but this is a gap that most likely would have been more closed if the FINE/Marine simulation would be able to run for a physical time equal to the one of Star-CCM+. In comparison, the result for this wave case from model tests presented in Guo Bingjie's PhD thesis is 45.96 N. So FINE/Marine over-predicts by 73% and Star-CCM+ by 40%.

If we look at the graph made, showing the relation between simulated added resistance and the wave length, that was based on the results in Star-CCM+. It somewhat has the desired shape of a graph like this, so the results does vary with wave length in the assumed way. But there is much room for more accuracy when it comes to the exact resistance numbers. Much because of the big fluctuations in the resistance values, and a clear mean value is hard to acquire. We also just saw in both FINE and Star the results were too big.

So much of the point of this thesis has disappeared because of all the problems and time. But a study of CFD in general has been done, and some preliminary results have been made.

8 References

Faltinsen, O. M. (1990). Sea Loads on Ships and Offshore Structures, Cambridge University Press.

White, F. M. (2006). Viscous Fluid Flow, 3. Edition

H.K. Versteeg (1995). An introduction to computational fluid dynamics, The finite volume method.

J. Blazek (2001). Computational Fluid Dynamics – Principles and Applications

John F. Wendt (2009). Computational Fluid Dynamics, An Introduction, 3. Edition

UserManual_FINEMarine_3.0

Theoretical_Manual_FINEMarine_3.0

UserManual-HEXPRESS_212-1

PhD thesis of Guo Bingjie

Best Practice Guidelines for Marine Applications of Computational Fluid Dynamics **(**https://pronet.wsatkins.co.uk/marnet/publications/bpg.pdf)

http://en.wikipedia.org/wiki/Computational_fluid_dynamics

http://www.cfd-online.com/

9 Appendices

9.1 Appendix A: Hydrostatics provided by Sverre Steen

MARINTEK SINTEF	HYDROSTATICS		
		ENCL.	1.
		REPORT	
		DATE	2012-12-14
		REF	

HULL MODEL NO.:	M2846A	Model Scale:	58.000
Loading condition:	Design WL		
Draught AP/FP:	20.800 / 20.800 [m]		

	Symbol	Unit	SHIP	MODEL
Length overall	L_{OA}	[m]	333.500	5.750
Length betw. perp.	L_{PP}	[m]	320.000	5.517
Breadth moulded	B	[m]	58.021	1.000
Depth to 1st deck	D	[m]	26.007	0.448
Draught at $L_{PP}/2$	T	[m]	20.800	0.359
Draught at FP	T_{FP}	[m]	20.800	0.359
Draught at AP	T_{AP}	[m]	20.800	0.359
Trim (pos. aft)	t	[m]	0.000	0.000
Rake of keel		[m]	0.000	0.000
Rise of floor		[m]	0.000	0.000
Bilge radius		[m]	0.000	0.000
Water density	ρ_s	[kg/m³]	1025.00	1000.00
Shell plating thickness		[mm]	0.00	0
Shell plating in % of displ.		[%]	0.50	0.00
Length on waterline	L_{WL}	[m]	325.503	5.612
Breadth waterline	B_{WL}	[m]	58.021	1.000
Volume displacement	∇	[m³]	312677.4	1.603
Displacement	Δ	[t]	322096.8	1.611
Prismatic coefficient*	C_P	[-]	0.8115	0.8115
Block coefficient*	C_B	[-]	0.8097	0.8097
Midship section coefficient	C_M	[-]	0.9977	0.9977
Longitudinal C.B. from $L_{PP}/2$	LCB	[m]	11.185	0.193
Longitudinal C.B. from $L_{PP}/2$*	LCB	[% L_{PP}]	3.495	3.495
Longitudinal C.B. from AP	LCB	[m]	171.185	2.951
Vertical C.B.	VCB	[m]	10.874	0.187
Wetted surface	S	[m²]	27800.68	8.264
Wetted surf. of transom stern	A_T	[m²]	13.48	0.004
Waterplane area	A_W	[m²]	16743.75	4.977
Waterplane area coefficient	$C_W(L_{WL})$	[-]	0.887	0.887
Longitudinal C.F. from $L_{PP}/2$	LCF	[m]	-0.070	-0.001
Longitudinal C.F. from AP	LCF	[m]	159.930	2.757
Immersion	DP_1	[t/cm]	171.623	0.051
Trim moment	MT_1	[t m/cm]	3880.008	0.020
Transv. metacenter above keel	KM_T	[m]	24.310	0.419
Longit. metacenter above keel	KM_L	[m]	387.402	6.679

Remarks: *Refers to L_{PP}
Hydrostatic corrections not included
Turbulence stimulator: Trip wire at station 19.5

9.2 Appendix B : Open office sheet to calculate time step

Data provided by the user	*Value*	
Ship length (m)	5.5172000	
Ship speed (m/s)	1.0450000	
Wave Length λ (m)	0.0000000	→ *Set « 0 » if unknown*
Wave period (s)	1.8000000	→ *Set « 0 » if unknown*
Wave Height H = 2A (m)	0.1500000	
Wave direction (deg)	180.0000000	
Interface location (m)	0.3586200	
Gravity (m²/s)	9.8100000	
Dyn. viscosity of water (Pa.s)	0.0010987	
Specific mass of water (Kg/m³)	998.8200000	

Space: computed/recommended	*Value*	*Normalized Value (*)*
Reynolds number (water)	5.241E+06	-
Froude number	1.420E-01	-
Wave length λ (m)	5.0586444	0.9168862
Wave period (s)	1.8000000	0.3409338
Wave Steepness Ak =2πA/λ	0.0931552	-
Wave celerity (m/s)	2.8103580	2.6893378
Box: Zmax (m)	0.4336200	0.0785942
Box: Zmin (m)	0.2836200	0.0514065
Box: ΔZ (m)	0.0125000	0.0022656
Box: ΔX/ΔY(m)	0.0632331	0.0114611
Lofted surface: ΔZ (m)	0.0055172	0.0010000

Time: computed/recommended	*Value*	*Normalized Value (*)*
Wave frequency (Hz)	0.5555556	-
Advance frequency (Hz)	0.2065771	-
Encounter frequency (Hz)	0.7621326	-
Encounter period (s)	1.3121076	0.2485232
Particle velocity (max of) (m/s)	0.2617994	
Time step (s)	0.0065605	0.0012426

(*) using ship lenght & speed

Some default parameters	*Value*	
Number of Pts / wave length	80	→ *Set to 100 for finest grid (60 is*
Number of Pts / wave height	12	→ *Set to 16 for finest grid (10 is t*
Number of Δt / wave period	200	→ *80 is correct for ships without sc ship length, more than 200 time st together with a grid with normal de sufficient to obtain the same order*

9.3 Appendix C: Poster

Master thesis in Marine Technology – 2013

Application of CFD to sea-keeping analysis

Steinar Malin

steinmal@stud.ntnu.no

Supervisor: Sverre Steen

NTNU – Trondheim
Norwegian University of Science and Technology

Introduction

Traditionally hull resistance is estimated under calm water conditions. But this is a state that is seldom met at sea, and the presence of waves is much more likely.
This will lead increased resistance and large induced ship motions.
For safety reasons it is good to know about the induced motions, and in an economic perspective the added resistance is important.

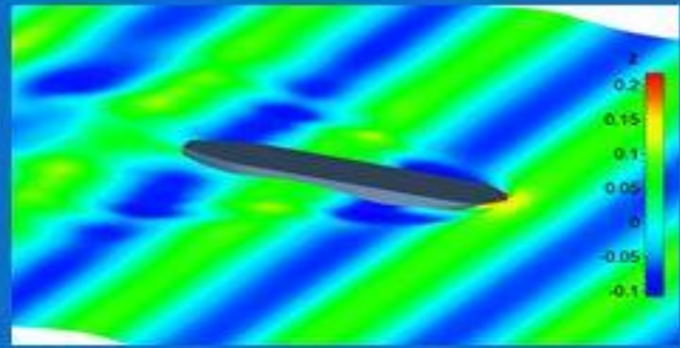
This is something that earlier has been done by the use of analytical methods based on potential theory to get a rough result.

The idea behind this thesis is to investigate the capabilities of some of todays CFD codes.

- Computational fluid dynamics makes use of numerical methods and algorithms to solve problems that involves fluid flow. This makes it very attractive in marine technology because it can be applied to many scenarios.
- Examples of uses are motion prediction, resistance tests, performance evaluations, modeling of propulsors and wave impact analysis.
- This thesis takes a look upon the use of CFD in sea-keeping analysis, and compares the results of two different codes in regards of added resistance.
- The two codes used are Star-CCM+ and FINE/Marine.
- The results will also be compared to model tests done earlier.

Mathematical background

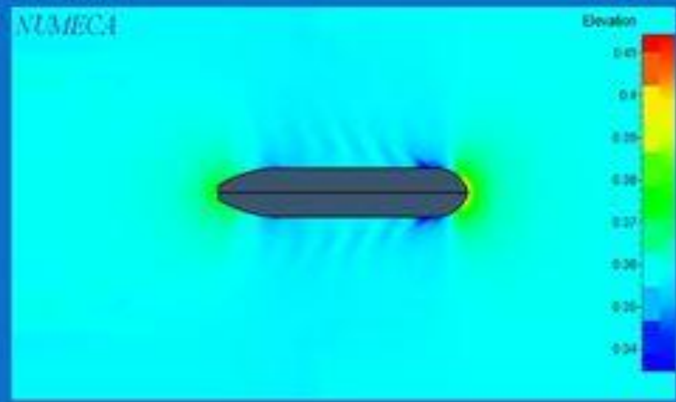
- To solve flow problems one has to solve a collection of equations, commonly referred to as the Navier-Stokes equations. This is something the CFD software does numerically in different ways, depending on the code.
- The governing equations are presented below:



- Mass equation: $\frac{\partial \rho}{\partial t} + div(\rho u) = 0$

- Momentum (x): $\frac{\partial (\rho u)}{\partial t} + div(\rho u u) = -\frac{\partial p}{\partial x} + div(\mu \cdot grad(u))$

- Internal energy: $\frac{\partial (\rho i)}{\partial t} + div(\rho i u) = -p\,div(u) + div(k \cdot grad(T)) + \phi$



- Together with an appropriate turbulence model one is able to simulate different flow problems and show the results. To the right is two cases of a ship moving in water, one with waves present and one in calm water.

References

[1] http://www.sintef.no/home/MARINTEK/Maritime/CFD/Sea-keeping/

[2] An Introduction to computational fluid dynamics – H K Versteeg