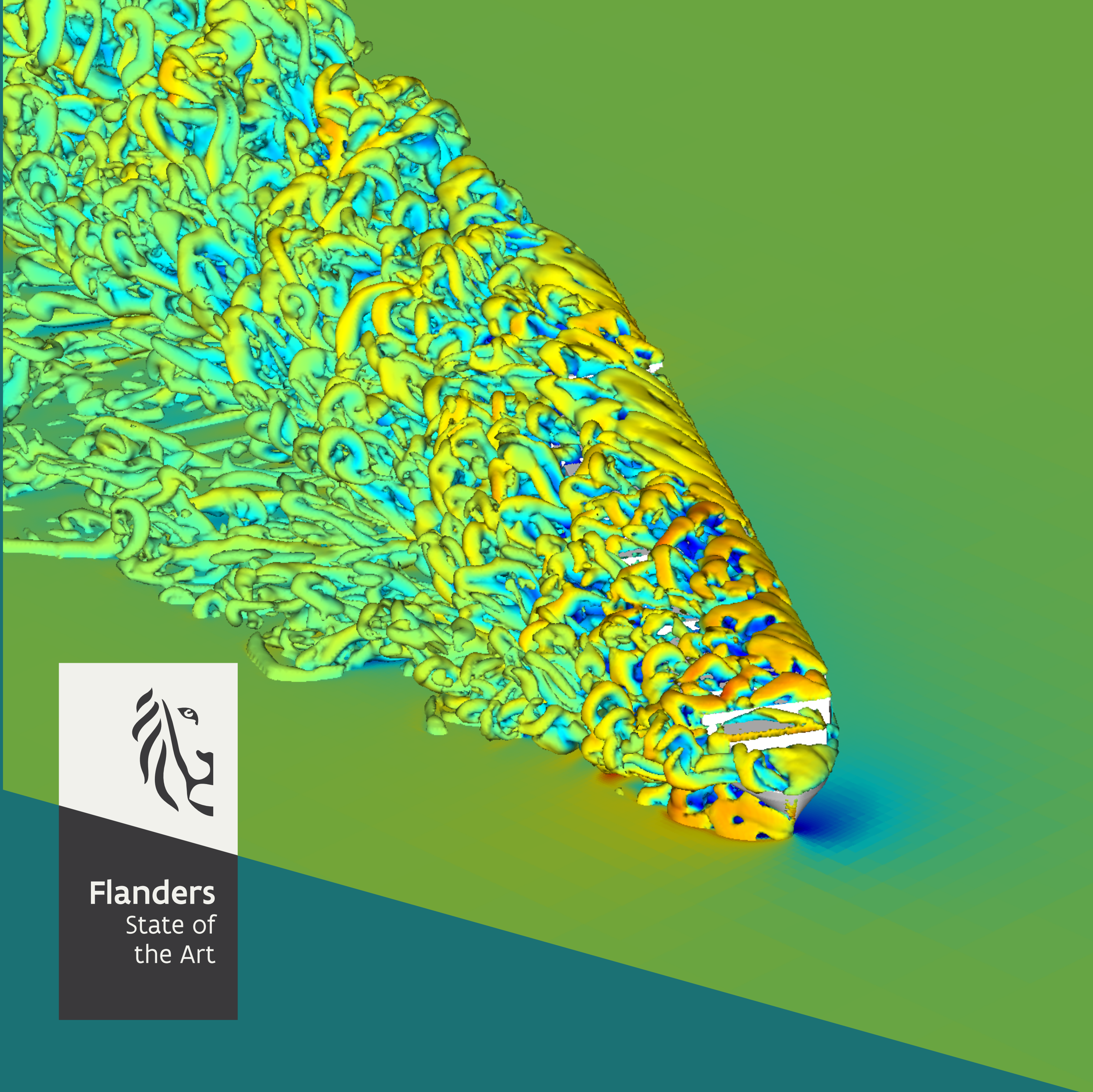


Flanders
State of
the Art

16_058_2
FHR reports

Evaluation of Gerris flow solver for the computation of wind coefficients

Parameter variations and validation

DEPARTMENT
MOBILITY &
PUBLIC
WORKS

www.flandershydraulicsresearch.be

Evaluation of Gerris flow solver for the computation of wind coefficients

Parameter variations and validation

Van Hoydonck, W.; Verwilligen, J.; López Castaño, S.

Flanders Hydraulics Research
Flanders State of the Art

Cover figure © The Government of Flanders, Department of Mobility and Public Works, Flanders Hydraulics Research

Legal notice

Copyright and citation

D/2022/3241/183

This publication should be cited as follows:

Van Hoydonck, W.; Verwilligen, J.; López Castaño, S. (2022). Evaluation of Gerris flow solver for the computation of wind coefficients: Parameter variations and validation. Version 3.0. FHR Reports, 16_058_2. Flanders Hydraulics Research: Antwerp.

Document identification

Customer:	Flanders Hydraulics Research	Ref.:	WL2022R16_058_2
Keywords (3-5):	CFD, ship wind coefficients, Gerris flow solver, unsteady computation		
Knowledge domains:	Harbours and waterways > Manoeuvring behaviour > Wind > Numerical calculations		
Text (p.):	60	Appendices (p.):	35
Confidentiality:	☒ No	☒ Available online	

Author(s):	Van Hoydonck, W.

Control

	Name	Signature
Reviser(s):	Verwilligen, J.; López Castaño, S.	Getekend door:Jeroen Verwilligen (Signat Getekend op:2022-09-06 11:27:46 +02:00 Reden:Ik keur dit document goed Jeroen Verwilligen Santiago Lopez Castaño (Signature) Digitaal ondertekend door Santiago Lopez Castaño (Signature) Datum: 2022.09.07 08:58:50 +02'00'
Project leader:	Van Hoydonck, W.	Wim Van Hoydonck (Signature) Digitaal ondertekend door Wim Van Hoydonck (Signature) Datum: 2022.09.06 16:58:24 +02'00'

Approval

Head of Division:	Bellafkih, K.	Getekend door:Abdelkarim Bellafkih (Sig Getekend op:2022-09-06 10:43:43 +02:0 Reden:Ik keur dit document goed Abdelkarim Bellafkih

CERTIFIED ISO 9001

Abstract

The objective of project 16_058 is to evaluate the open-source Gerris flow solver for the computation of ship wind coefficients.

In the original project plan, it was envisaged to only execute a parameter variation and grid convergence study with Gerris. While this research was started in 2016, due to shifting priorities, it has only been finalised in 2022. Not only was time spend to validate the Gerris flow solver, but time was also used to automate tasks that are executed before this type of Computational Fluid Dynamics (CFD) computations are executed. In the first report (Van Hoydonck *et al.*, 2021), automation of tasks to prepare geometry for CFD computations were reported. The current report will present details on validation and use of Gerris for the determination of wind coefficients.

Multiple sources of experimental (wind tunnel) data have been used to validate Gerris for the computation of wind coefficients on ship-like structures, ranging from simple wall-mounted cubes and rectangular prisms to ship scale models. Before executing a validation, the effects on the solution of certain choices in the configuration of the flow solver are determined. The maximum refinement level (both in the wake and at the ship structure), the length of the refinement region and the blockage are all subject to parameter variations to determine their respective influence on the solution. Gerris is configured such that the viscous interaction of the flow with the ship structure (friction) is not included in the solution, only the pressure part is taken into account. The inherent unsteady nature of the flow is taken into account by solving the Navier-Stokes equations in a time-accurate manner. Overall, the predicted hull forces agree well with experimental data, following the trends for the complete range of incidence angles. In some cases, the peak values of the yawing moment shown significant deviations from the experimental values, the cause of this is unknown. A grid convergence study shows that predicted forces converge as the grid is refined.

One of the biggest disadvantages of Gerris is that in order to capture small details in the ship geometry, the maximum refinement level has to be set fairly high, resulting in a large cell count. Combined with the poor parallel scalability of the solver on the Flanders Hydraulics Research (FHR) cluster, this means that the required time to execute computations may be more than what was expected. The successor of Gerris (Basilisk[1]) has been reported to show significantly better parallel scalability, which may make it more suitable for computations where a large refinement is required on the hull. This will have to be verified in future research.

In an appendix of this report, a discussion is held about a typographical error that was discovered in a peer-reviewed publication related to the experimental determination of wind coefficients Andersen (2013). It is found that the error affects the accuracy and reliability of steady Reynolds-Averaged Navier Stokes (RANS) CFD methods reported in literature that used Andersen's data as validation material. This second report also contains a critical review of past research related to the simulation of atmospheric boundary layers in FINE/Marine.

[1] `http://www.basilisk.fr`

Contents

List of Figures

List of Tables

Nomenclature

Abbreviations

AGR	Adaptive Grid Refinement
CAD	Computer Aided Design
CFD	Computational Fluid Dynamics
CMA	cumulative moving average
DES	Detached Eddy Simulation
DOF	Degree Of Freedom
EFD	Experimental Fluid Dynamics
FHR	Flanders Hydraulics Research
LES	Large Eddy Simulation
RANS	Reynolds-Averaged Navier Stokes
WT	Windtunnel

Latin symbols

$_0$	X-axis of earth-fixed axes system pointing North	—
A_0	Area of base of truncated pyramid	m^2
A_1	Area of top of truncated pyramid	m^2
A_F	Frontal projected area	m^2
A_L	Lateral projected area	m^2
B	ship beam	m
C	Courant number	—
c	Reference length	m
C_K	Non-dimensional moment around the x-axis	—
C_M	Non-dimensional moment around the z-axis	—
C_N	Non-dimensional moment around the y-axis	—
C_X	Non-dimensional force in the x-direction	—
$C_{X_{AF}}$	Non-dimensional force in the x-direction, using the frontal projected area A_F	—
C_Y	Non-dimensional force in the y-direction	—
C_Z	Non-dimensional force in the z-direction	—
e_w	Wake expansion factor	—
$\vec{F}$	Force vector	N
$\vec{F}'_s$	Full scale force vector	N
h	height, measured positive up	m
h_o	Height of an object	m
H_S	Geometric centre of gravity of the lateral projected area, measured from the waterline	m
$\bar{H}$	Mean height of the lateral plane	m
i	Refinement level	—
K	Moment around the x-axis	Nm
K'	Non-dimensional moment around the x-axis	—
L_{oa}	Length over all	m
L_{pp}	Length between perpendiculars	m

L_{ref}	Reference length	m
l_w	Length of wake region	m
M	Moment around the z-axis	Nm
$\overrightarrow{M}$	Momemt vector	Nm
$\overrightarrow{M}'_s$	Full scale momemt vector	Nm
N	Moment around the y-axis	Nm
N'	Non-dimensional moment around the y-axis	—
n_c	Cell count	—
N_s	Number of refinements on the body	—
N_w	Number of refinements in the wake	—
q	Dynamic pressure: $\frac{1}{2}\rho U^2$	Pa
q_{ref}	Reference dynamic pressure	Pa
$\overrightarrow{r}$	Position vector	m
r_c	Radial coordinate of wake coarsening region	—
Re	Reynolds number	—
T	Dimensionless time	—
T'	Dimensional time	s
U_a	Apparent wind velocity	m/s
U_s	Ship velocity	m/s
U_w	Wind velocity	m/s
U_∞	Reference velocity	m/s
V_w	Wake volume	m^3
w_o	Width of an object	m
X	Force in the x-direction	N
X'	Non-dimensional force in the x-direction	—
x_c	X-coordinate of start of grid coarsening region	—
$x_{c,max}$	Maximum x-coordinate of wake coarsening region	—
$x_{c,min}$	Minimum x-coordinate of wake coarsening region	—
Y	Force in the y-direction	N
Y'	Non-dimensional force in the y-direction	—
y_0	Y-axis of earth-fixed axes system pointing East	—
y_c	y-coordinate of start of grid coarsening region	—
Z	Force in the z-direction	N
z_c	z-coordinate of start of grid coarsening region	—
Δt	Time step	—
Δx	Length of a cell	—

Greek symbols

β	Drift angle: arctan $\frac{-v}{u}$	°
γ	Angle between U_a and U_s	°
λ	Linear scale of ship model	—
μ	Dynamic viscosity	Ns/m^2
μ_∞	Reference dynamic viscosity	Ns/m^2
ρ	Density	kg/m^3
ρ_∞	Reference density	kg/m^3
φ	Angle of relative wind, $0°$ is head wind	°

1 Introduction

1.1 Computation of wind coefficients using FINE/Marine

In recent years, the CFD software suite FINE/Marine has been used to compute improved wind coefficients of the Tripoli estuary vessel after feedback from the skippers during real-time simulation trials (Van Hoydonck *et al.*, 2015b). The improved wind coefficients gave acceptable results, but a very long computing time was required (approximately one month cluster time). As a consequence, this type of computations cannot be executed on a regular basis to improve existing wind coefficients or create new wind coefficients for new simulator vessels. In another project (Van Hoydonck *et al.*, 2016) the wind field on the leeward side of a roro ship was computed using FINE/Marine to evaluate the shelter effect of a moored vessel on the manoeuvrability of another vessel. Multiple days (both cluster time and man hours) were required to generate a suitable wind field. The origin of the long turn-around time (discussed in more detail in the paragraphs that follow) of these computations is two-fold:

- computing times were long, and
- a significant number of man-hours was needed for the preparation of the computations.[2]

The long computing times have two principle causes: a) the inclusion of viscosity in the governing (Navier-Stokes) equations next to the pressure and momentum equations, and b) the unsteadiness of the problem. The total force experienced by an object exposed to a relative wind field is a combination of pressure forces (acting perpendicular to the surfaces of the object) and viscous forces (acting parallel to the surfaces of the object). To reliably compute the viscous forces, a very large number of very small cells is required close to the surface of the object[3] which increases the computation time significantly.

For most vessels, the superstructure consists of a set of blunt objects stacked next to and on top of each other that result in massive separation regions on the leeward side of said objects. Sharp edges make the problem unsteady: vortices are shed periodically from these parts. To get the best possible result, one should use a time-accurate (at least second order in time) solver instead of a solver that is only first-order accurate in time. The time-accuracy requirement poses stricter demands on the maximum time step that can be used, with a consequence that computing times will increase further. In industrial applications, RANS solvers are most commonly used for this type of application. Apart from the momentum and pressure equations, turbulence models are required for closure of the governing equations. The need for a turbulence model in RANS solvers increases the problem even further, because the specific turbulence model used will have an effect on the results as well.

Apart from the long computing times for this type of CFD computation, the amount of man-hours required for the setup of the computations should not be underestimated either. For a single set of coefficients (from $\varphi = 0°$ to $180°$) with a $10°$ increment, 19 computations are required. If the CFD tool supports sliding meshes, a computational domain consisting of two separate domains (one large rectangular hexahedron and a smaller cylindrical domain in which the ship is located) can be used where the inner cylindrical domain can be rotated around the z-axis to change the wind incidence angle without creating a new mesh. This feature was however not yet available in FINE/Marine when the coefficients of the Tripoli were initially computed, hence 19 different grids were created. During the course of the computations, (some of) the resulting grid (or grids) may have

[2]In recent years, both sliding and overset grids and Adaptive Grid Refinement (AGR) have been implemented in the flow solver of FINE/Marine. These solver features can help reduce the engineering time to some extent. The development of the C-Wizard also helps for streamlining the computational setup and grid generation process especially when combined with AGR. However, robust settings for AGR for a general case can still be a challenge to get right, which partially offsets the reduced engineering time due to the automated setup.

[3]This can be 50% of the total number of cells used to discretize the complete domain.

to be altered to capture the wake regions close to the ship with sufficient detail. The use of AGR can reduce the turn around time in this case by refining the grid during the computation based on e.g. the magnitude of vorticity in the wake.

1.2 Wind coefficient data used in the simulator

For most of the existing mathematical models of vessels currently available in the simulator at FHR, the origin of the wind coefficient data used is unknown. Some were copied from vessels with a similar shape (but different absolute dimensions), others are taken from similar vessels found in literature. For most of them, the exact origin is unknown with as a result that it is unclear if the data used is suitable in the first place[4].

Some of the mathematical models contain multiple configurations where a single parameter such as the draft is changed. Also, for some container ships and bulk cargo ships, there are multiple configurations present where only the shape of the cargo is changed (for example an even distribution of containers over the full length of the vessel, or a more step-wise distribution, with fewer containers near the bow). Without a large database of wind coefficients for all possible classes of ships (either computed using CFD or as a result of extensive wind tunnel measurements), it is difficult to know how wind coefficients should be changed to reflect a change in e.g. the container configuration for a specific case.

It should also be noted that the effect of wind is not always very important (or, not the most important part of a simulation). The forces experienced by vessels due to the proximity of banks, the bottom of the waterway or other ships may be significantly larger in magnitude than wind effects. On the other hand, there are situations where wind may be a limiting factor in the manoeuvrability of vessels (for example, at low speed during manoeuvres in a harbour in high wind conditions). At the moment only the drag force, lateral force and yawing moment are used in the simulator: this is sufficient for a mathematical model with three degrees of freedom in the horizontal plane. For the upgrade of the simulator mathematical models to six degrees of freedom, the inclusion of the roll moment due to (lateral) wind is a logical extension. The vertical (lifting) force and the pitch moment are generally less important but could be included if the coefficients are obtained from CFD computations because their computation has no additional overhead.

In the simulators at FHR, each variant of a ship has a separate `load_*.xml` file that contains data about the general characteristics of the ship, the wind coefficients table and related longitudinal and lateral windage areas, a wave table, and the type of mathematical models for bank suction and squat behaviour. For example, for the Tripoli estuary vessel, six load files are available with drafts between 2.52 m and 4.5 m. These all have the same wind coefficients, but they do have unique windage area values that correspond to the different drafts. If the draft of a vessel changes (and nothing else), the average height of the vessel that is exposed to wind changes as well. In a uniform wind field, the pressure on the superstructure will be (close to) proportional to the exposed area. When the wind field is non-uniform, this is not the case: the distribution of pressure (and hence the coefficients derived from it) will be a function of both the reference area and a characteristic height relative to the reference height of the velocity profile. As an example, a long vessel with a limited height with the same lateral area as another shorter but higher vessel, will have a different absolute value for the pressure acting on the lateral area.

1.3 Reference systems and definition of coefficients

When a vessel moves in a still atmosphere, the superstructure is exposed to a uniform relative wind equal to its own velocity. When the vessel is not moving in the presence of wind, the vertical wind profile will be highly

[4]Although it is perfectly possible to add references in comments in `xml` files, the `load_*xml` files used in the simulator at FHR generally do not contain such information.

non-uniform due to the viscous interaction between the moving air and the earth. For the combination of these two, when the vessel sails in the presence of wind, the total wind load consists of the wind load due to the motion of the ship in addition to the wind load due to the non-uniform atmospheric velocity (Andersen, 2013). Hence, two sets of wind coefficients are required to compute the total load: one set determined for a uniform wind profile, and a second set determined using a suitable atmospheric profile. In most nautical simulation applications, this distinction is not made: the total relative velocity U_a is computed by vector subtraction of the ships' motion U_s from the atmospheric wind U_w (see Fig. 1),

$$\vec{U}_a = \vec{U}_w - \vec{U}_s. \tag{1}$$

The relative wind angle φ can be computed[5] from the drift angle β and the three velocity components U_a, U_w and U_s. The total wind loads are obtained by multiplying the coefficient values interpolated at the φ with the reference dynamic pressure q and suitable reference areas and lengths.

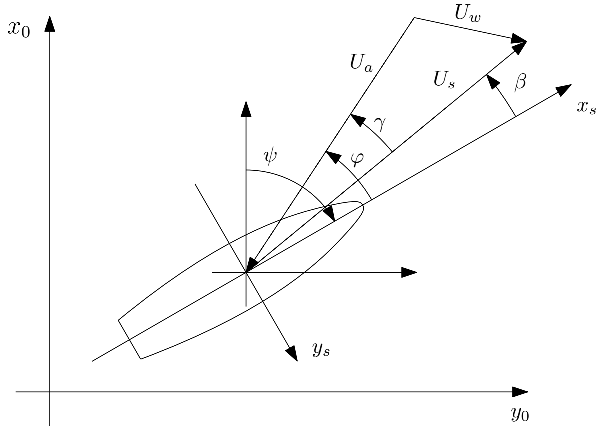


Figure 1 – Definition of the apparent wind vector U_a and the relative wind angle φ.

The dimensionless wind load coefficients are defined as (Blendermann, 2013)

$$C_X = \frac{X}{qA_L}, \tag{2}$$
$$C_Y = \frac{Y}{qA_L}, \tag{3}$$
$$C_Z = \frac{Z}{qA_L}, \tag{4}$$
$$C_K = \frac{K}{qA_L\bar{H}}, \tag{5}$$
$$C_M = \frac{M}{qA_L L_{oa}}, \tag{6}$$
$$C_N = \frac{N}{qA_L L_{oa}}, \tag{7}$$

where $q = \frac{1}{2}\rho {U_a}^2$ is the dynamic pressure[6], A_L the lateral projected area, L_{oa} the reference length of the lateral plane and $\bar{H}$ is the mean height of the lateral plane,

$$\bar{H} = \frac{A_L}{L_{oa}}. \tag{8}$$

[5] For the current investigation, φ is set directly.
[6] The dynamic pressure is computed using the reference speed U_a taken at an altitude of 10 m

Blendermann (2013) relates longitudinal force to the frontal projected area,

$$C_{X_{AF}} = C_X \frac{A_L}{A_F}. \tag{9}$$

Fujiwara and Nimura (2005) and Ueno *et al.* (2012) use Eqs. 3, 5, 7 and 9 to derive their mathematical model of the wind loads. However, the marine literature is not entirely consistent when it comes to defining wind coefficients. For example, Andersen (2007) and Wagner (2005) use the following definitions for C_X, C_Y, C_K and C_N:

$$C_X = \frac{X}{qA_F}, \tag{10}$$

$$C_Y = \frac{Y}{qA_L}, \tag{11}$$

$$C_K = \frac{K}{qA_L H_S}, \tag{12}$$

$$C_N = \frac{N}{qA_L L_{oa}}, \tag{13}$$

where H_S is the geometric centre of gravity of the lateral projected area measured from the waterline (Wagner, 2005). Both H_S and $\bar{H}$ are reference heights, and for a rectangle with base L_{oa} and height h, they are related as follows:

$$H_S = \frac{\bar{H}}{2}. \tag{14}$$

Dimensionless values computed with Eqs. 10 to 13 are preferred when experimental values are to be used for a ship that is not geometrically similar to the one for which the coefficients were determined (Andersen, 2013). To allow for a comparison of difference configurations of the same ship, coefficients that make use of reference dimensions that are independent of the reference areas A_L and A_F or quantities derived from these reference areas are preferred (Andersen, 2007; Andersen, 2013; Wagner, 2005):

$$X' = \frac{X}{q{L_{pp}}^2}, \tag{15}$$

$$Y' = \frac{Y}{q{L_{pp}}^2}, \tag{16}$$

$$K' = \frac{K}{q{L_{pp}}^3}, \tag{17}$$

$$N' = \frac{N}{q{L_{pp}}^3}. \tag{18}$$

1.4 Problem summary

From the computational point of view, there is a question of efficiency that needs to be addressed so that a timely answer can be given to questions of the nautical researchers related to the adaptation of existing and creation of new wind coefficient sets for simulator vessels. This holds true for both the required time for preparing Computer Aided Design (CAD) models and the time to configure and execute computations.

From the side of the nautical researchers at FHR, there is a desire to reduce some of the guess work that is currently involved in creating wind coefficients for new simulator vessels or adapting them based on feedback from pilots. Without knowing the magnitude of the influence of configuration changes on the wind coefficient values, it is difficult to determine if such changes require modifications to existing wind coefficient data.

1.5 Proposed solution

1.5.1 Contributions of viscosity and pressure

An analysis of the components of the resultant forces and moments from $\varphi = 0°$ to $180°$ has shown that the contribution of the pressure to the resultant forces and moments is by far the largest (Van Hoydonck *et al.*, 2015b). The relative contribution of friction forces is rather small, as can be seen in the graph in Fig. 2 for the lateral force ($\varphi = 0°\text{to}180°$), which shows that the viscous component is smaller than 3%.

The importance of the viscous forces in relation to the pressure forces for a certain problem is governed by the Reynolds number (Anderson, 1991),

$$Re = \frac{\text{pressure forces}}{\text{viscous forces}} = \frac{p - p_\infty}{\tau} = \frac{\rho U^2}{\mu \partial u / \partial y} = \frac{\rho_\infty {U_\infty}^2}{\mu_\infty U_\infty / c} = \frac{\rho_\infty U_\infty c}{\mu_\infty} = Re_\infty, \tag{19}$$

where U_∞, ρ_∞ and μ_∞ are the reference velocity, density, and dynamic viscosity of the fluid and c is a reference length. When the ship length is used as the reference length, a vessel with $L_{oa} = 300$ m sailing in a wind field with magnitude $U_\infty = 10$ m/s has a Reynolds number equal to $Re_{L_{oa}} = 205 \times 10^6$. In comparison, for a scale model of a ship with a length of 4 m that is tested in the towing tank at a model speed of 0.5 m/s, the Reynolds number for the superstructure of the hull equals $Re_a = 137 \times 10^3$ while for the submerged hull, $Re_w = 1.76 \times 10^6$. The lower values of Re for model scale tests mean that viscous forces are significantly more important there (especially for streamlined bodies) than at full scale.

For bluff bodies that lack sharp corners where the flow will separate (such as cylinders and spheres), viscosity is still important even for larger Reynolds numbers because it affects the flow inside the boundary layer: lower Reynolds numbers result in a laminar boundary layer that will detach faster from the object, while for higher Reynolds numbers, the boundary layer will first transition to turbulent flow. For this case, boundary layer separation is postponed which results in a smaller wake region and less drag. For bluff bodies with sharp corners, the flow detaches at the sharp corners and further downstream it may reattach again. For the Reynolds numbers considered here, flow in the boundary layer will become turbulent before separation.

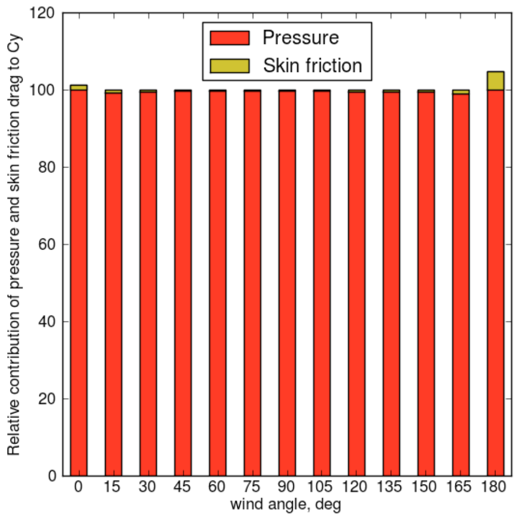


Figure 2 – Relative contribution of the viscous and pressure components to the total lateral force on a container ship at wind angles from $0°$ to $180°$ (Van Hoydonck *et al.*, 2015b).

This result (and the complete analysis presented in Van Hoydonck *et al.* (2015b)) suggests that it may be sufficient[7] to use a CFD solver that neglects the contribution of viscosity: using an Euler solver instead of a Navier-Stokes solver. As stated before, this can have a significant impact on the computing time as this reduces the number of cells (required for the resolution of the boundary layer) significantly.

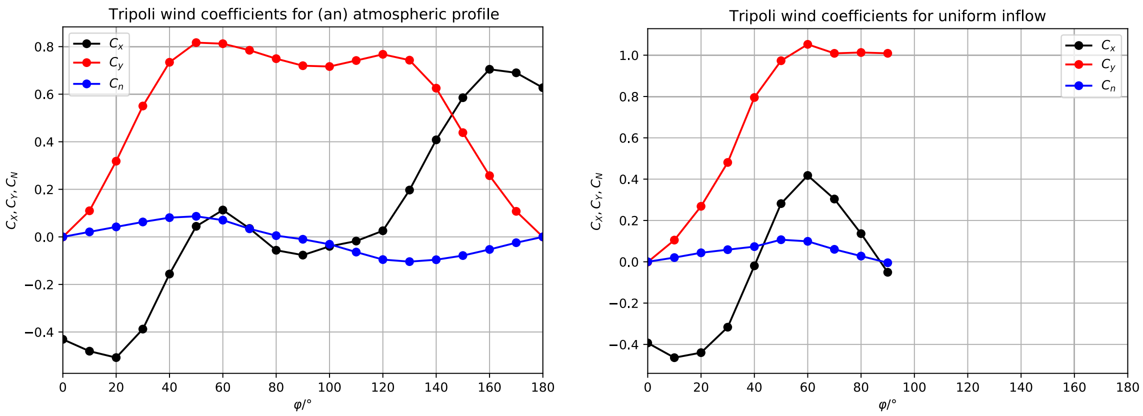
1.5.2 Grid generation

The second issue that contributes to the high turn-around time is the pre-processing time required to generate numerical grids (Blendermann, 2013). The solver should ideally be able to adapt its grid to the problem at hand in a completely automatic way, using simple logical rules.

1.5.3 CFD Solver

To the knowledge of the author, there is only one CFD solver that addresses these points: the open-source Gerris flow solver (Popinet, 2009, 2003), discussed in more detail in chapter 2.

An initial evaluation of Gerris has shown that for the Tripoli, the shape of the resultant lateral force is very similar (although somewhat higher) to the one computed using FINE/Marine (see Fig. 3), in a time frame that was almost two orders of magnitude shorter than using FINE/Marine. The latter results were computed using an atmospheric boundary layer at the inlet whereas the Gerris results were computed using a uniform wind field at the inlet, which explains the higher C_Y values for Gerris as compared to FINE/Marine.



(a) (b)

Figure 3 – Wind coefficients of the Tripoli estuary vessel computed using (a) FINE/Marine with an atmospheric boundary layer and (b) Gerris with a uniform wind field.

There is a significant difference in the longitudinal force (especially above 45 degrees) for which an explanation should be found. However, this does not mean that Gerris should be discarded if discrepancies remain, because as stated above, the velocity profile at the position of the vessel in the computational domain for the FINE/Marine result is actually unknown. Both for the Gerris results and the FINE/Marine results, the period of the C_X graph is proportional to sin(4φ). In literature, this type of graph for C_X is associated with wind tunnel tests of passenger ships (Blendermann, 2013; Fujiwara and Nimura, 2005; Ueno *et al.*, 2012). This may mean that simplifying the containers on the deck of the CAD geometry of the Tripoli to a single block is one simplification too much; in other words, the gaps between the container bays may need to be preserved in order to get a correct graph for C_X. This is also concluded by Janssen *et al.* (2017), where wind coefficients computed

[7] Blendermann comes to the same conclusion that for ships and offshore structures above the water surface the contribution of friction forces can be neglected (Blendermann, 2013).

using four different geometric approximations of a container ship are compared with experimental results. A significant improvement was found in the results when the gaps between container stacks are included in the ship geometry[8].

Multiple approaches are available to capture the effect of a wind gradient (Blendermann, 2013, p. 33):

- experimentally, utilising a wind tunnel equipped to simulate atmospheric boundary layer flow (Armitt and Counihan, 1968);
- by book-keeping where the structure is decomposed in its principle components (Walree and Willemsen, 1988);
- by direct specification of the wind profile in CFD computations (Janssen *et al.*, 2017).
- by transformation of wind load coefficients obtained in uniform flow using the effective dynamic pressure of the non-uniform flow (Blendermann, 1993);

Although it is possible to set a non-uniform velocity profile at the inlet in Gerris, this will not be pursued here for multiple reasons:

- the addition of a non-uniform velocity profile at the inlet in Gerris means that the cell count will be enlarged to resolve the profile until it arrives at the vessel. Due to the use of square/cubic cells, Gerris has a disadvantage over a solver that uses hexahedral cells of arbitrary aspect ratio. The impact on the required computation time is significant;
- there is no guarantee that an arbitrary velocity profile defined at the inlet remains constant throughout the computational domain (Blocken *et al.*, 2007; Richards and Norris, 2012);
- wind tunnel tests are often performed in a uniform wind field (Blendermann, 1993).

1.5.4 Accounting for a velocity gradient in wind coefficients determined in a uniform wind field

Given a set of coefficients obtained in a uniform wind field (either in a wind tunnel or as a result of CFD computations), Blendermann (2013, 1993) outlines a procedure to account for a gradient in the wind field by adapting the coefficients. Fig. 4 (from Blendermann (2013)) shows an application of this procedure, displaying wind coefficients of a ferry in a flow with a gradient reduced to uniform flow and where – for comparison – wind coefficients determined in uniform flow are presented as well.

Since a similar conversion may be required for coefficients determined with Gerris, Blendermann's procedure is repeated below.

The wind forces and moments acting on a ship are computed from

$$X = C_{X_{AF}} q_{ref} A_F, \tag{20}$$
$$Y = C_Y q_{ref} A_L, \tag{21}$$
$$K = C_K q_{ref} A_L \bar{H}, \tag{22}$$
$$N = C_N q_{ref} A_L L_{oa}, \tag{23}$$

with q_{ref} the effective dynamic pressure of the apparent wind at 10 m height. According to Blendermann, it can be formulated as a weighted average of the mean dynamic pressure $\bar{q}_H$ over the mean height of the ship's lateral plane and the dynamic pressure q_H of the oncoming flow at that height:

$$q_{ref} = k_q \bar{q}_H + (1 - k_q) q_H, \tag{24}$$

with $0 \leq k_q \leq 1$. Reference dynamic pressures for the above four force and moment components are listed in Table 1 (Blendermann, 2013).

[8]They have other issues that are discussed in Appendix A6.

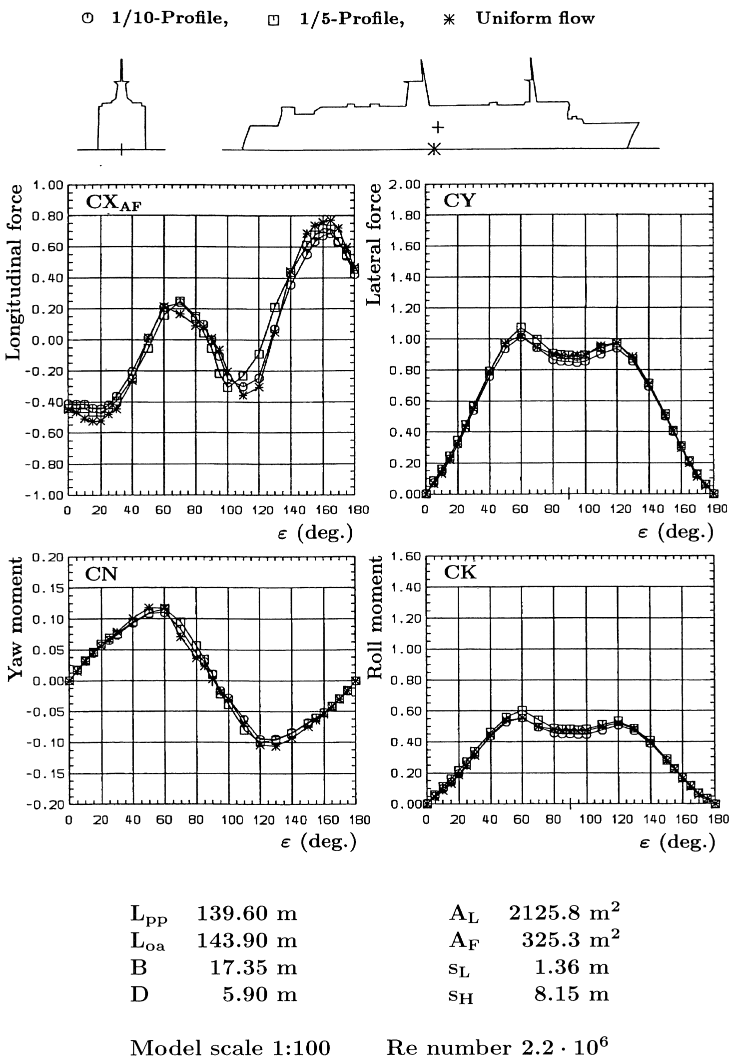


Figure 4 – Wind load coefficients of a ferry in flow with a gradient reduced to uniform flow, coefficients from uniform flow for comparison (Blendermann, 2013).

Table 1 – Reference dynamic pressures according to Blendermann (2013) for converting coefficients from a uniform wind field to one with a gradient.

Force component	q_{ref}
X	q_H
Y	$k_q\bar{q}_H + (1 - k_q)q_H$, $k_q \approx 0.6$
N	same as for Y
K	q_H, or same as for Y

1.5.5 Accounting for (partial) sheltering

Apart from the above mentioned need to account for velocity gradients in the vertical direction (due to an atmospheric boundary layer), horizontal gradients (due to e.g. sheltering behind another ship) are required. The current implementation of the wind influence in the simulator contains an implementation that can account for asymmetries in the horizontal wind field. They apply to the computation of the lateral force Y and the yawing moment N,

$$Y = \frac{1}{2}\rho C_Y(\varphi)\bar{V_r}^2 A_L + Y_{ur}\bar{u_r}\bar{r_r}, \tag{25}$$

$$N = \frac{1}{2}\rho C_N(\varphi)\bar{V_r}^2 A_L L_{pp} + N_{rr}\bar{r}|\bar{r}| + N_{ur}\bar{u_r}\bar{r_r}, \tag{26}$$

where $\bar{r_r}$ is the average relative wind gradient over the ship length at the reference height, $\bar{u_r}$ is the average relative longitudinal wind speed at the reference height, and Y_{ur}, N_{rr} and N_{ur} are additional coefficients that account for longitudinal asymmetry in the ship geometry. This method cannot take vertical gradients into account, and assumes that longitudinal gradients can be accounted for by the longitudinal two-dimensional distribution of the ship superstructure.

During the course of this research, a different concept is suggested to investigate in the future. By dividing the geometry of the ship is multiple complementary parts, force and moment coefficients can be computed for the parts separately. For example, by splitting the ship at the midship location in a forward and aft part, and computing the force and moment coefficients for both parts separately in uniform flow, two sets of coefficients result that can be used to compute a rudimentary form of sheltering in the longitudinal direction.

By dividing the hull in more than two sections in the longitudinal direction (e.g. three to five), a more gradual build-up of sheltering can be accounted for. In the simulator, this obviously requires a wind field that contains regions on the leeward side of objects where the wind condition is altered as compared to unobstructed space. In the past, one such wake has been computed using CFD for a simulation study in the port of Zeebrugge, at that time using FINE/Marine (Van Hoydonck *et al.*, 2016). Alternative methods to generate wake fields that require either less preparatory time, or are more generic will not be addressed in the current research.

1.6 Report contents

In this report, research is documented related to the execution of CFD computations using Gerris to determine wind coefficients of a ship in a uniform wind field.

Chapter 2 discusses the flow solver that is used and how it is configured for the wind coefficient computations. The domain and reference systems are discussed, conversion of resultant forces and moments to dimensionless quantities is being treated and the reference geometries are introduced. This is followed by a discussion of the boundary conditions, initial conditions, and the methods used in Gerris to automatically adapt the grid to the flow features. The chapter is finished with a discussion of solver settings.

In Chapter 3, parameter variation studies are executed: the influence on the solution of blockage, the maximum refinement level in the wake and the length of the wake refinement region are investigated. All these parameters also affect the computing times as they determine to a large extent the grid size. Afterwards, a grid the convergence study is undertaken, first for a simple geometric shape without small details, and afterwards for a ship geometry that includes small details. For all these investigations, no comparison with experimental data is executed.

In Chapter 4 numerical results are compared with experimental data for the geometric shapes introduced in Chapter 2.

Conclusions of the research and an outlook to the future are presented in Chapter 5.

Some additional material is presented in several appendices at the end of the report. Appendix A1 presents a complete sample Gerris configuration file for computation the forces on a wall-mounted object subject to a uniform flow field. In Appendix A2, a method is outlined to compute reference parameters (such as windage areas) of a vessel based on available (experimental) wind coefficient values. Appendix A3 contains additional time series plots and flow field visualisations for the wall mounted cube results presented in § 4.1.2.

Appendix A4 discusses an issue with the inlet boundary condition that is often used in Gerris for setting a uniform velocity field. Additional time history plots related to the investigation of the influence of blockage for the VLCC (discussed in § 3.3.1) are displayed in Appendix A5. Appendix A6 is a lengthy discussion of the consequences of a typographical error in a published journal article related to wind tunnel experiments to determine wind coefficients for a container ship and how this error affects other publications that use the data as validation material. The final appendix (A7) discusses an issue related to the simulation of an atmospheric boundary layer in FINE/Marine and the effect on simulations executed in the past with FINE/Marine.

2 Gerris Flow Solver Computational Setup

In this chapter, some background information is given regarding the use of the Gerris[9] Flow Solver (Popinet, 2003) in the current research for the computation of wind coefficients. A minimal configuration file is listed in Appendix A1.

2.1 Domain and reference systems

2.1.1 Domain

In Gerris, physical parameters are dimensionless, the default box that is used to create a domain has linear dimensions of 1 m). Two of these boxes placed behind each other define the computational domain. The vessel is placed at the centre of the bottom of the first box. For the initial computations, the vessel occupies 1/3rd of the length of the box: the overall length of the vessel is 0.3333 m, which gives a scale factor $\lambda = \frac{0.3333}{340} = \frac{1}{1020}$. For $\varphi = 0°$, the distance from the bow to the inlet is equal to the ship length ($1L_{oa}$) and at $\varphi = 90°$, this distance is larger ($= 1.5L_{oa} - B/2$)[10].

To minimise the influence of blockage on the resultant forces and moments on the ship, the ship dimensions should be significantly smaller than the domain dimensions. For $\varphi = 0°$, the blockage is approximately $0.18\,\%$ and determined by A_F. When $\varphi = 90°$, the blockage is higher at $1\,\%$. Lateral blockage in this case approaches $33\,\%$. These blockage values are significantly lower than the experimental results of Andersen (2013) where the lateral blockage for $\varphi = 90°$ approaches $88\,\%$.

For $\varphi = 0°$, the computational domain is visualised in Fig. 5, which shows one lateral side, the outlet surface and the bottom surface with the initial discretization applied at the start of the simulation. Each domain box (that by default, consists of a single cell) is subdivided four times in each direction. The resulting cells have a linear dimension of $\frac{1020}{16} = 63.75$ m. Around the vessel, the subdivisions are increased to the maximum value as specified in the input file (typically eight to eleven). Cell sizes as a function of the subdivision level are given in Table 2.

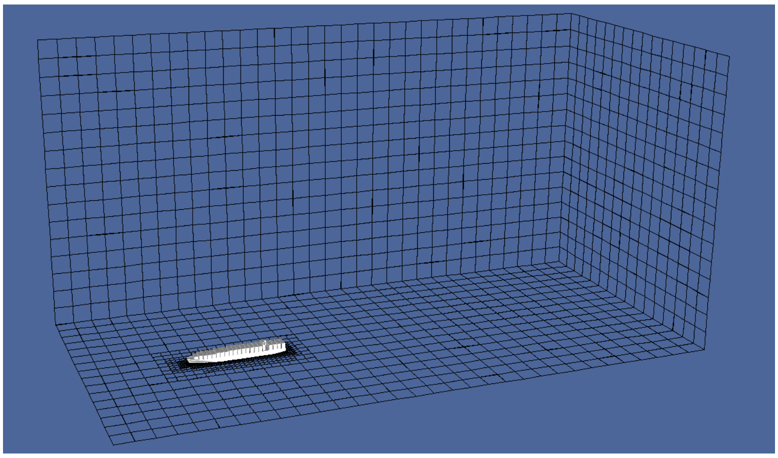


Figure 5 – Location and size of the container vessel w.r.t the computational domain.

[9]The solver is named after the common water strider *Gerris lacustris*, see `http://gfs.sf.net` for more information.
[10]The distance between the ship and the inlet can always be increased if it is deemed too short.

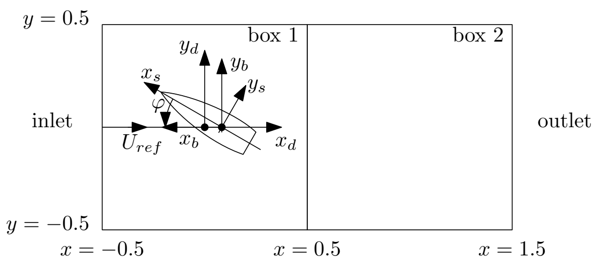
Table 2 – Cell sizes as a function of subdivision level.

subdivision level	cell size	(full scale) cell size/m
0	1.0	1020.0
1	0.5	510.0
2	0.25	255.0
3	0.125	127.5
4	0.0625	63.75
5	3.125×10^{-2}	31.88
6	1.563×10^{-2}	15.94
7	7.813×10^{-3}	7.969
8	3.906×10^{-3}	3.984
9	1.953×10^{-3}	1.992
10	9.766×10^{-4}	0.9961
11	4.883×10^{-4}	0.4980
12	2.441×10^{-4}	0.2490
13	1.221×10^{-4}	0.1245
14	6.104×10^{-5}	0.062 26

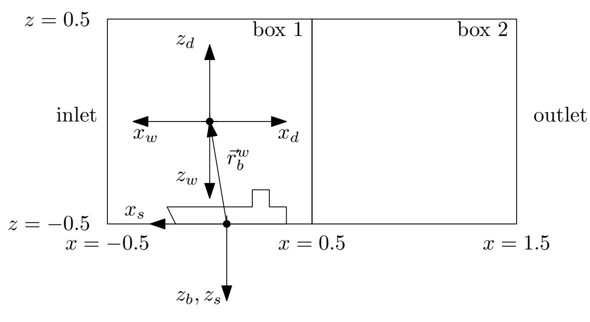
2.1.2 Reference systems

Integral quantities such as resultant forces and moments on solid bodies are computed in the global reference system of the domain. For the ship simulator, the quantities are required in a ship-fixed axes system. The origin and orientation of the domain reference frame is shown in Fig. 6. The origin is located at the centre of the first domain box.

Each of the six faces of a domain box is associated with a spatial direction. For box 1 for example, the face at $x = -0.5$ is its *left* boundary, while the face at $x = 0.5$ is its *right* boundary. When a domain consists of multiple boxes, the spatial directions are used to position the boxes relative to each other. All six spatial directions are defined in Fig. 7.



(a) Top view



(b) Side view

Figure 6 – Computational domain with conventions of domain axis system and ship-fixed axis system.

The conversion from the domain axes system to the ship axes system is achieved with four transformation matrices and an offset between the second and the third one. Initially, the transformations were implemented in scalar form, but it became apparent that a formulation in terms of matrix multiplications is significantly more efficient in Python, especially for long time series.

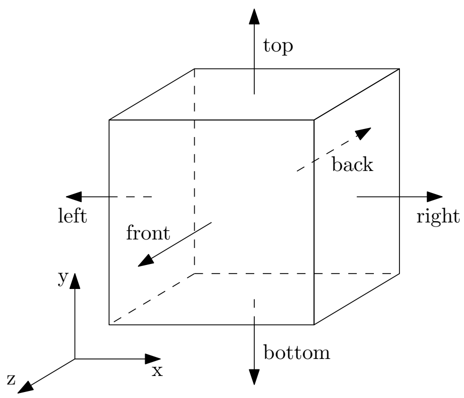


Figure 7 – Definition of spatial direction in Gerris.

Transformation matrix $\mathbf{E}_{w|d}$ converts a vector $\vec{F}_d$ in the domain axis system to a vector $\vec{F}_w$ in the wind axis system,

$$\vec{F}_w = \mathbf{E}_{w|d} \vec{F}_d, \tag{27}$$

where

$$\mathbf{E}_{w|d} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix}. \tag{28}$$

At the bottom of the domain, a third (bottom) and fourth (ship) axes system are defined to convert forces resolved in the wind axis system to the ship axes system. The bottom axes system has the same orientation as the wind axes system, so no transformation is required for the forces, but due to the offset $\vec{r}_b^{\,w} = [0, 0, -0.5]^T$, moments get a contribution of the forces:[11]

$$\vec{F}_b = \vec{F}_w, \tag{29}$$
$$\vec{M}_b = \vec{M}_w + \vec{r}_b^{\,w} \times \vec{F}_w. \tag{30}$$

Finally, the wind angle φ defines the orientation between the bottom axes system and the ship axes system,

$$\mathbf{E}_{s|b} = \begin{bmatrix} \cos\varphi & \sin\varphi & 0 \\ -\sin\varphi & \cos\varphi & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{31}$$

Hence,

$$\vec{F}_s = \mathbf{E}_{s|b} \vec{F}_b, \tag{32}$$
$$\vec{M}_s = \mathbf{E}_{s|b} \vec{M}_b. \tag{33}$$

2.2 Computation of coefficient values

The influence of the parameter variations are judged based on the resultant forces and moments acting on the ship's upper structure. After conversion to the ship-fixed axes system, the resultant forces $\vec{F}_s$ and moments

[11]If the ship is moved further aft (e.g. by 0.2) to increase the distance between the ship and the domain inlet, $\vec{r}_b^{\,w} = [0.2, 0, -0.5]^T$.

$\vec{M}_s$ are first redimensionalized to full scale quantities,

$$\vec{F}'_s = \vec{F}_s \rho {U_a}^2 {L_{ref}}^2, \tag{34}$$
$$\vec{M}'_s = \vec{M}_s \rho {U_a}^2 {L_{ref}}^3, \tag{35}$$

where $U_a = 10$ m/s and L_{ref} is the scale factor (1020 m $= 3L_{oa}$ for the VLCC of Andersen (2007)) and not the vessel reference length L_{oa}.

Time must be scaled as well, as it is also dimensionless. In a simulation that runs for ten (dimensionless) seconds, the fluid is flushed five times out of the domain (since the inlet velocity has unit value and the domain has a length of two). To make time dimensional again, it is multiplied by the reference length and divided by the reference velocity,

$$T' = T\frac{L_{ref}}{U_a}. \tag{36}$$

Hence, for the VLCC of Andersen one dimensionless second of simulation corresponds to a full-scale simulation time of 102 s.

For comparison and validation purposes, non-dimensional quantities are preferred. As already discussed in Van Hoydonck *et al.* (2021), the marine literature has not settled on a single system for computation of coefficient values from dimensional quantities. In addition, in most cases only two forces (longitudinal and lateral) and one (yaw) or two moments (yaw and roll) are included. The vertical force Z and pitch moment M are generally deemed less important to adequately predict the 6 Degree Of Freedom (DOF) dynamic behaviour of a vessel due to wind.

For the grid convergence study, all six coefficients for the three forces and moments will be shown in the results. The coefficients as defined by Blendermann (2013, p. 38) (Eqs. 2 to 7) will be used for this purpose. For comparison with experimental data, the coefficients utilized in the experiments will be used.

2.3 Reference Geometries

The ship geometry used for the parameter variations and grid convergence study is configuration *01-01-02* (see Fig. 8) of Andersen (2007) and Andersen (2013). The lines plans in Andersen (2007) were used to create the hull geometry in Rhino (Van Hoydonck *et al.*, 2015a). It is a container ship with an overall length of 340 m, its breadth is approximately 45 m and the length between perpendiculars equals 320 m. The freeboard is approximately 17 m. Two variants of the ship geometry are tested: one with gaps between container rows (Fig. 9a) and one where the gaps between the container rows are filled (Fig. 9b). As was noted in Van Hoydonck *et al.* (2015b) where the coefficients of the Tripoli estuary vessel were determined, removing the gaps between the container rows to simplify the meshing, may have been one simplification too much.

The reference point for the force and moment computations (and the origin of the CAD model) is located on the waterline, in the symmetry plane of the ship at the midship location (see Fig. 8).

Due to issues with the wind tunnel data of Andersen (2007) as discussed in Appendix A6, results obtained with the VLCC geometry will only be used for parameter variations, not for validation.

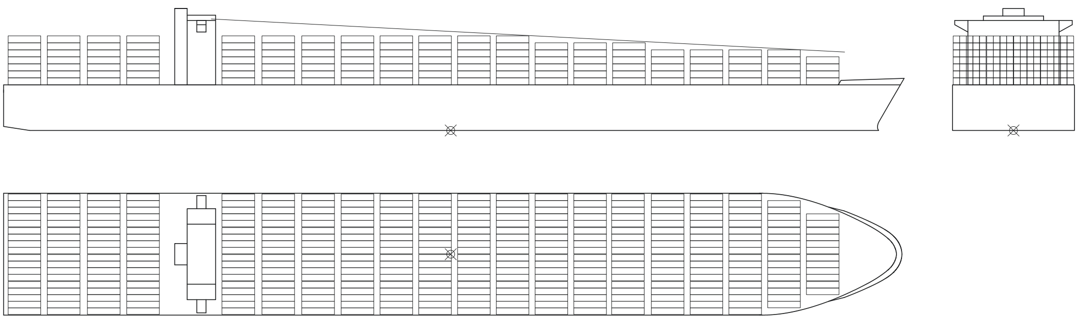
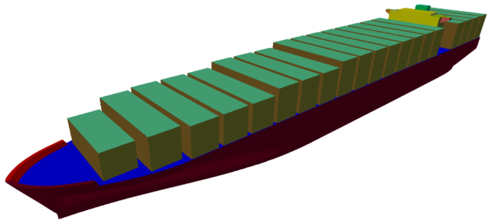
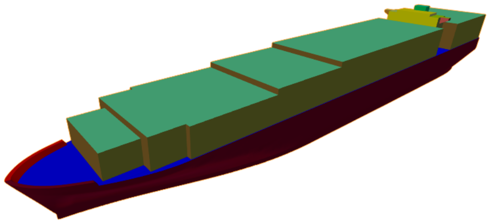


Figure 8 – Top view, side view and front view of configuration 01-01-02 (Andersen, 2007).



(a) Variant G0: with gaps between container stacks.



(b) Variant G1: without gaps between container stacks.

Figure 9 – Geometric variants of the container vessel.

For validation purposes, simpler geometries are used. Blendermann (2013) reports measurements taken on rectangular prisms with four different width-to-height ratios of 1, 2, 4 and 6.66. These geometries are shown in Fig. 10. The scale of these prisms as used in the computations is different from the values as shown in this figure. For the smallest (cubic) prism, all sides have length 0.05, giving a blockage of $0.25\,\%$. The blockages of the wider prisms are $0.5\,\%$, $1\,\%$ and $1.66\,\%$, respectively. For these blockage values, corrections to account for blockage effects are not required (Engineering Sciences Data Unit, 1998).

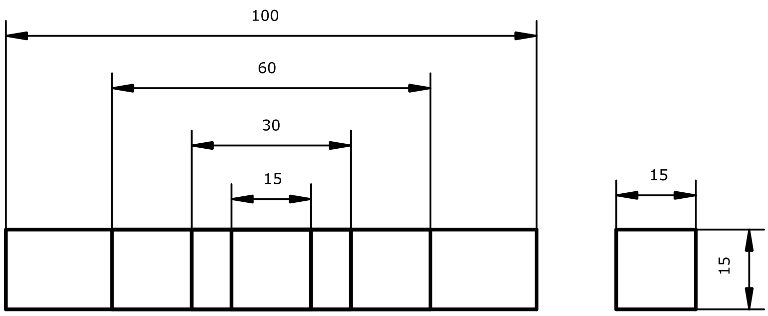


Figure 10 – Front view and side view of four rectangular prisms with different width-to-height ratios tested by Blendermann (2013).

2.4 Boundary conditions

By default, Gerris assumes that all domain boundaries are solid walls where a slip condition for the velocity is applied: the tangential stress on the wall equals zero. Since we are not interested in simulating an atmospheric boundary layer but rather a uniform velocity field, this condition is suitable for the domain bottom ($z = -0.5$). For the lateral sides and the domain upper surfaces, application of this boundary condition means that streamlines will be parallel to the surfaces (flow will not exit nor enter through these surfaces). Apart from the lack of development of a boundary layer on these surfaces, this condition is similar to the condition in a closed wind tunnel.

At the inlet, a uniform velocity field is applied normal to the domain surface, for all other variables, a zero-gradient condition is assumed (using `GfsBoundaryInflowConstant`). Initially, the more general `GfsBoundary` condition was used for the inlet, with a `BcDirichlet` condition on the normal velocity component. This method is used in multiple examples that are supplied with the solver (notably the Tangaroa[12] example), and works well for short simulation times. However, for simulations where the fluid is flushed out of the domain more than once ($t_{end} > 2$ for a domain with length equal to two), spurious jets appear at random locations in the domain inlet that can affect the convergence of forces and moments on the ship. An example is shown in Appendix A4. For the outlet surface, the pressure is equated to zero as are the gradients for all other variables (using `GfsBoundaryOutflow`).

2.5 Initial conditions

To start a simulation, an initial condition is required for the velocity field. This has to be set as accurate as possible to reduce initial transients in the solution. For the current research, the velocity field in the entire domain is set to the same value as the boundary condition at the inlet: `U = 1`. After the first step, the solution looks very similar to a potential solution: one where the flow does not (yet) detach from solid objects, and where pressure is (almost fully) recovered on the leeward side of objects, see Fig. 11.

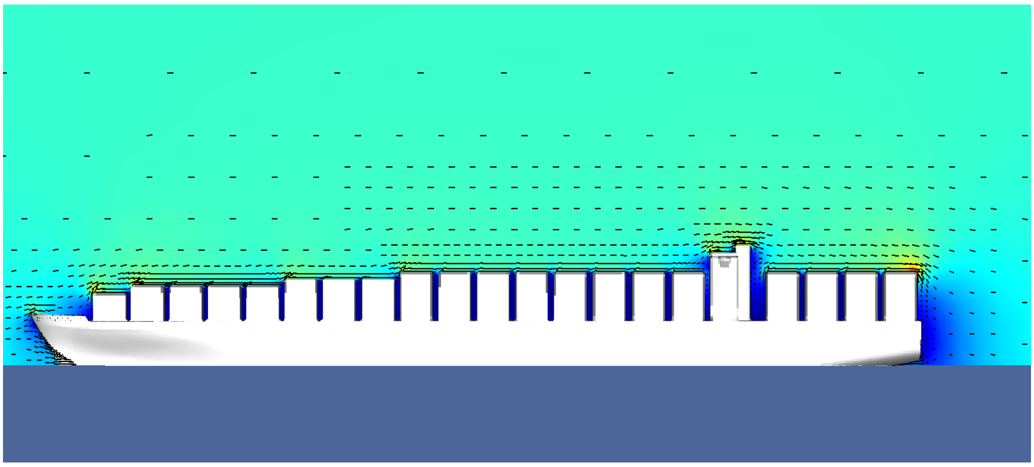


Figure 11 – Velocity field in the symmetry plane of the ship for $\varphi = 0°$ after the first time step, resembling a potential solution.

2.6 Grid adaptation to the flow features

One of the primary reasons to use Gerris for wind coefficient computations is its ability to automatically adapt the grid to the flow features. The high Reynolds number of the (turbulent) flow and the blunt bodies from which most ship superstructures are composed make that the flow detaches from sharp corners which forms a wake with significant fluctuations. The wake shed by upstream parts of the superstructure affects the flow around parts of the superstructure further downstream, which means that it must be adequately resolved in

[12] See here: `http://gfs.sourceforge.net/examples/examples/tangaroa.html`

the neighbourhood of the hull. In addition, the size and location of the wake depends significantly on the orientation of the ship with respect to the oncoming flow. For CFD solvers that do not contain features to automatically adapt the grid in the wake to the flow (both refinement and derefinement), the engineering time to create a sufficiently refined grid in the wake (for each required φ) can be a time-consuming task (Van Hoydonck *et al.*, 2015a,b).

For the current simulations, the grid in the domain is initially adapted five times in each direction (`Refine 5`). In addition, near the boundary of the hull, the grid is adapted maximally (`RefineSolid MAXREF`, where `MAXREF` is user settable, by default 10 refinements for a ship with a length of $1/3$). The wake shed by the superstructure is tracked and refined up to `WAKEMAXREF` based on the value of the vorticity and velocity magnitude in a cell using `GfsAdaptVorticity`, which uses a cell cost defined as the norm of the local vorticity vector multiplied by the cell size and divided by the maximum of the velocity over the whole domain[13].

2.6.1 Refinement zones

Aft of the hull, eddies are artificially damped by coarsening the grid before the outlet is reached. Multiple methods can be used for this. The simplest method simply sets the x-coordinate x_c beyond which the maximum refinement is gradually reduced to zero. The starting point this region can be defined programmatically. For example, it can be made dependent on the wind angle φ:

$$x_c = (x_{c,max} - x_{c,min})|\cos\varphi| + x_{c,min}, \tag{37}$$

where $x_{c,max}$ and $x_{c,min}$ are the extrema of the starting point of the region (see Fig. 12). An alternative for Eq. 37 with a continuous (smooth) first derivative over the complete range of wind angles is also shown in Fig. 12, its function definition is

$$x_c = (x_{c,max} - x_{c,min})\frac{1+\cos 2\varphi}{2} + x_{c,min}. \tag{38}$$

As an alternative, the boundary beyond which the grid is coarsened can be defined as a vertical cylindrical or spherical surface using

$$x_c{}^2 + y_c{}^2 < r_c{}^2, \tag{39}$$
$$x_c{}^2 + y_c{}^2 + z_c{}^2 < r_c{}^2, \tag{40}$$

where r_c is the radius of the cylinder or sphere beyond which the grid is coarsened. All of these conditions are implemented as ternary operators in Gerris input files, an example is shown below:

```
AdaptVorticity { istep=1 } {maxlevel = (x*x+y*y < 0.4*0.4 ? MAXREF : 0) cmax = 1e-2}
```

The conditions defined as such can be nested as well, for example to spread the derefinement zone over a larger area, such that the maximum refinement in the wake is reduced in steps.

2.6.2 Wake cell count estimates

Long wakes in combination with a high refinement can result in a significant cell count that can increase the computation time to weeks on a limited number of CPUs[14]. To get an idea of the order of magnitude of the cell count for an object of a specific size and a wake length aft of it, the number of cells at each refinement level in a truncated rectangular pyramid are computed. Given the (approximate) width w_o and height h_o of an object,

[13] See `http://gfs.sf.net/wiki/index.php/GfsAdaptVorticity`
[14] The parallel efficiency of Gerris is evaluated later in the report.

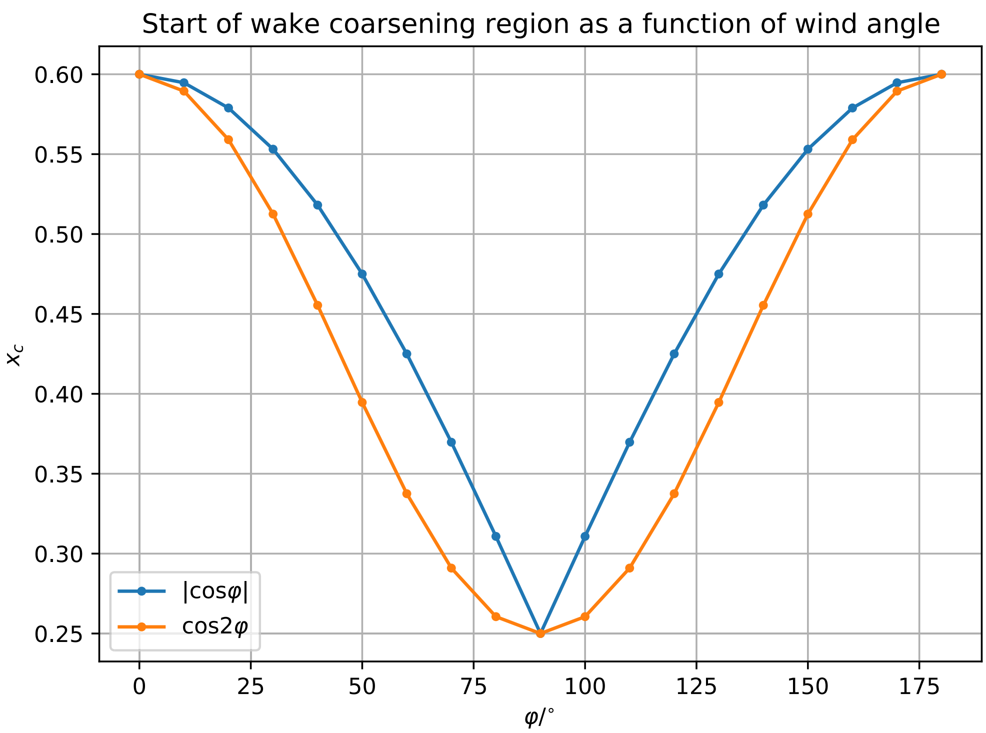


Figure 12 – Start of the wake coarsening region (x_c) as a function of the wind angle φ for two functions.

the length l_w of the wake region, and a wake expansion factor e_w that defines the scaling of the aft end of the pyramid, reference areas are computed:

$$A_0 = w_o h_o; A_1 = w_o e_w h_o e_w. \tag{41}$$

The volume of the (inverted) rectangular pyramid with base area A_0, top area A_0 and height l_w follows from

$$V_w = \frac{l_w(A_0 + A_1 + \sqrt{A_0 A_1})}{3}. \tag{42}$$

The cell size Δx at each subdivision level i follows from

$$\Delta x = \frac{1}{2^i}, \qquad 0 \leq i \leq n, \tag{43}$$

so that the number of cells n_c in wake volume equals

$$n_c = \frac{V_w}{\Delta x^3}. \tag{44}$$

For three representative cases (a cube with sides $w_o = 0.05$, a rectangular prism ($w_o = 0.2$), both with height $h_o = 0.05$ and third, a ship at $\varphi = 90°$ with length 0.7 and height 0.1), estimated cell counts are shown in Fig. 13 as a function of the refinement level (e_w was fixed at 1 while the wake length l_w was fixed at three times the object width). Even with a relatively small wake region aft of the cube (and more so for the rectangle and ship), the estimated cell count increases quickly beyond values that can be run in a reasonable amount of time on a small number of processors.

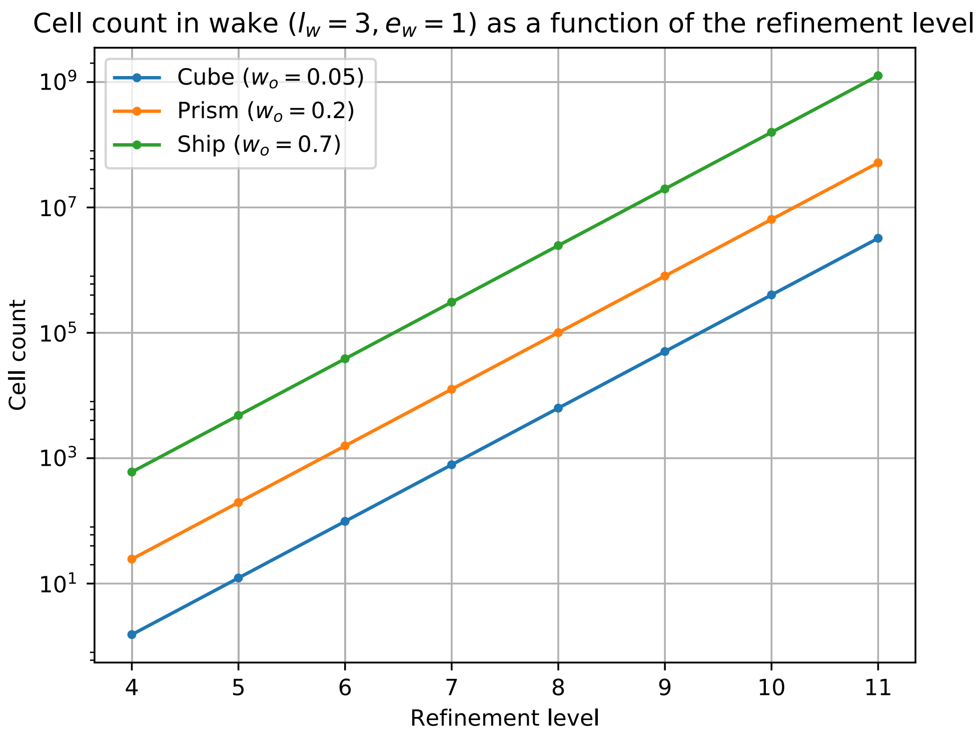


Figure 13 – Estimates of cell count in the wake region aft of a cube, a prism and a ship at φ = $90°$.

2.7 Solver settings

By default, physical viscosity is not taken into account in Gerris. This does not mean that there is no viscosity as all numerical schemes contain numerical viscosity due to the finite discretization of the numerical solution. Physical viscosity can be taken into account with `GfsSourceViscosity`. Its value is directly linked to the Reynolds number due to the dimensionless implementation in Gerris: `GfsSourceViscosity` $= 1/Re$.

When viscosity is not taken into account, the Reynolds number is infinite. For the type of geometry studied in this research, the assumption of an infinite Reynolds number is a reasonable approximation, as the ship upper structure is a bluff body where flow separates at sharp corners. The Reynolds number for the VLCC of Andersen (2013) based on the ship length (L_{oa}) and a relative velocity of 10 m/s in standard atmosphere at sea level equals $Re = 232.76 \times 10^6$. This means that `GfsSourceViscosity` should be set to 4.296×10^{-9} to add physical viscosity.

Gerris solves the Navier-Stokes equations in a time advancing manner. The domain length and the velocity defined at the inlet determine how many times the flow is completely flushed out of the domain. For the current research, the domain length is at most 2 and the inlet velocity is set to 1. With an end time set to 10, the fluid is flushed out of the domain five times. The time step is automatically computed based on the Courant number:

$$C = \frac{u\Delta t}{\Delta x} \leq 0.8, \tag{45}$$

where $C \leq 0.8$ is the default value in Gerris[15]. Assuming a maximum dimensionless velocity u of 2.5 (hence a maximum speed that is 2.5 times as high as the velocity at the inlet), the maximum time step can be estimated using Eq. 45 since $\Delta t_{max} = 0.8\frac{\Delta x}{u}$. For different maximum refinement levels, the cell size at the finest level and the maximum time steps are shown in Table 3. For a simulation with a maximum refinement level of 10, an estimated time step of $\Delta t = 0.000\,312\,5$ is found.

[15]The maximum allowable C number can be adjusted inside `GfsAdvectionParams`.

Table 3 – Estimated maximum time step values as a function of the refinement levels such that $C \leq 0.8$ assuming a maximum velocity of 2.5.

subdivision level	Δx	Δt
4	$1/16$	$1/50$
5	$1/32$	$1/100$
6	$1/64$	$1/200$
7	$1/128$	$1/400$
8	$1/256$	$1/800$
9	$1/512$	$1/1600$
10	$1/1024$	$1/3200$
11	$1/2048$	$1/6400$
12	$1/4096$	$1/12800$

Output of statistics related to the solid object (`OutputSolidStats`), output related to timing (such as the simulation time, current time step and wall clock time (`OutputTime`)), output related to the balance of the domain when running a computation in parallel (`OutputBalance`) are all emitted to the standard output file at every solver time step. For postprocessing or to restart the computation with different settings, the simulation state at specific time instances can be written to disk with the `OutputSimulation` keyword as well. Gerris can also directly write visualisations of certain flow field variables to disk when the `gfsview` module is loaded. Then, the `OutputView` command can be used to render a certain scene defined in a `gfview` file, and write it to a `ppm` file, an example is shown in Fig. 14. A summary of the timing of the different solver parts is output at the end of a simulation with the `OutputTiming` statememt.

By default, Gerris only saves instantaneous field variables such as U, V, W and P with calls to `OutputSimulation`. For validation with experiments, instantaneous values alone are not very useful, especially if there are significant fluctuations. For this reason, field values summed over (part of) the total simulation time can be computed as well. Using the `EventSum` statement, one can add values of a field variable, divide by the time step and store the result in a new variable. This variable is automatically saved with calls to `OutputSimulation`. Output of a simulation to a different format than the *native* `gfs` format is possible (e.g. Tecplot and VTK are supported), but the writer does not work correctly with parallel computations, it only exports that part of the computational domain that is handled by the first processor. This can be circumvented by converting the native output of Gerris to Tecplot or VTK format with the bash macros shown in Listing 1.

Listing 1 – Shell scripts to convert Gerris output to VTK and Tecplot format.

```
gerris2vtk()
  {
    gerris3D -e "OutputSimulation { step = 1 } ${1%.*}.vtk { variables = U,V,W,P$2
    format=VTK} " $1 > /dev/null
  }
gerris2tecplot()
  {
    gerris3D -e "OutputSimulation { step = 1 } ${1%.*}.dat { variables = U,V,W,P$2
    format=Tecplot} " $1 > /dev/null
  }
```

These should be called on the command line with the simulation output file (in `gfs` format) as first argument and optionally, can specify a list of fields such as `,SU,SV,SW,SP`. The output can then be processed further in Paraview.

As an example,

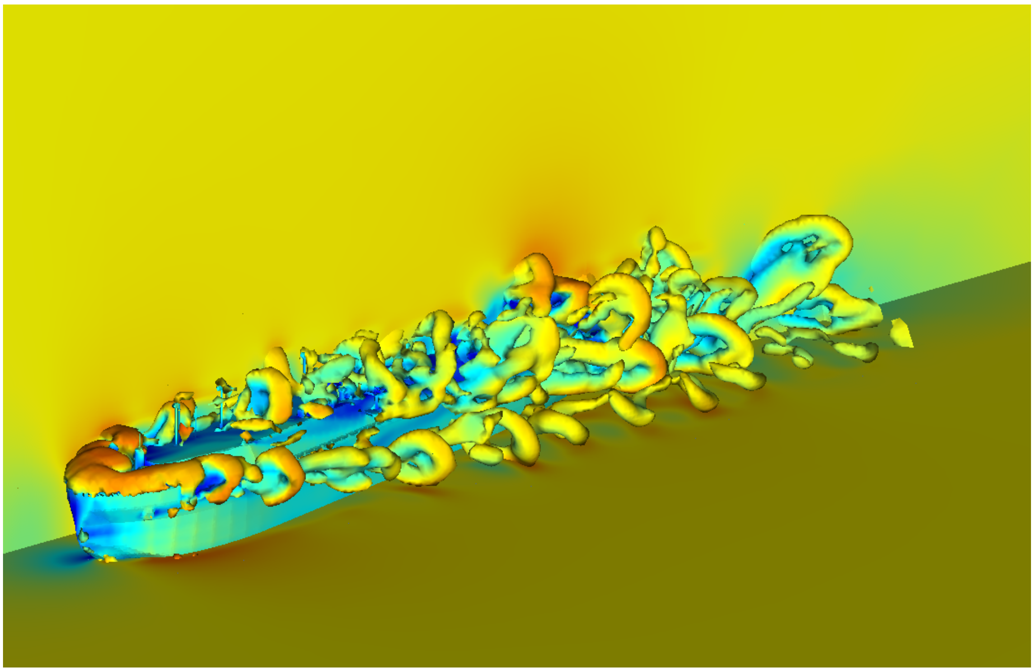


Figure 14 – Visualisation of the velocity magnitude in a horizontal and vertical plane and vortices shed by the superstructure using the $\lambda 2$ criterion for the *LNG Yamal* for $\varphi = -10°$.

```
$ gerris2vtk last_step.gfs ,SU,SV,SW,SP
```

will convert the output stored in `last_step.gfs` of a Gerris simulation to VTK format and add four extra fields to the resulting VTK file[16].

[16]If any of the field variables is not available, the macro will quit with an error.

3 Parameter variations and grid convergence study

In this chapter, computations are executed to determine the influence on the solution of certain parameters related to the setup of the computations. A grid convergence study will be executed as well.

3.1 Introduction

A grid convergence study will be executed using the VLCC hull of Andersen (2013). Computations are executed in a domain with unit cross section and with a unit velocity. The resulting forces and moments are converted to dimensionless coefficient values for which the reference areas must be known. Due to the nature of the flow solver (adaptive octree), instead of changing the mesh sizes relative to the geometry, the geometry itself will be scaled, with factors similar to what is typically used when executing a grid convergence study in FINE/Marine. A downside of this approach is that the reference areas do not remain constant. Knowing the exact scaling factors used to generate the geometries, the computed reference areas should give the same scaling factors. In § 3.2, first the reference areas are computed for the three different geometries by the method discussed in Van Hoydonck *et al.* (2021).

Scaling the geometry has an influence on the blockage of the ship in the domain. The grid convergence study should be executed with ship geometries that are small enough (as compared to the domain size) that blockage effects are negligible. This is investigated in § 3.3.1 where only the scale of the ship is altered (in multiples of two). All other parameters (such as the wake length aft of the ship, the maximum refinement level and the simulation time) will be kept constant relative to the ship length. Results of this investigation can be used to determine appropriate ship sizes for which blockage is negligible.

For the current research, the length of the wake aft of the ship (i.e. the starting point of the grid coarsening in the domain) has been made dependent on the wind angle (Eq. 37). For a single configuration, the effect of the wake length on the force coefficients will be examined to ensure that the time averages and their standard deviations are not affected.

3.2 Reference area computation

The use of unit dimensions for the domain in Gerris means that the reference geometry (shown in Fig. 15) is rather small: $L_{OA} = \frac{1}{3}$m, $B = 0.0454$ m and the total height $H = 0.0471$ m. Note that for these wind computations, only that part of the hull above the water level is included in the CAD model. The horizontal (bottom) plane of the CAD model at the intersection with the water surface has been extruded in the downward direction to ensure that no numerical errors occur because of coincident faces. The hull geometry is then positioned such that the water plane intersection coincides with the $z = 0$ m plane. The CAD models are given a small positive draft value to ensure that the downward extrusion of the bottom is not taken into account in the surface area computation; these draft values are also shown in Table 4. For one of the cases, the lateral and frontal surface areas are shown in Fig. 16.

For the same horizontal resolutions as used in Van Hoydonck *et al.* (2021) (from 100 to 6400 pixels), the convergence of the surface area is shown in Fig. 17 for the three geometries. The resultant values of the d parameter (i.e. the estimated surface areas for an infinite resolution) of the sigmoidal function, the surface area ratios

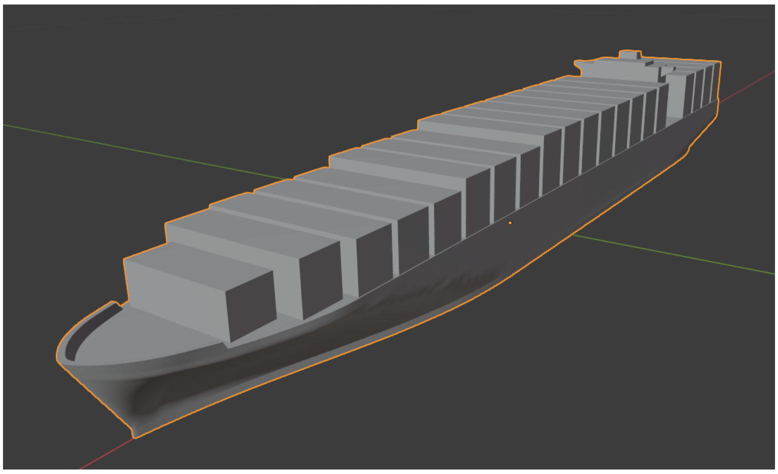


Figure 15 – Medium geometry of the VLCC (Andersen, 2013) used in the grid convergence study.

Table 4 – Scaling factors and blockage values of the different VLCC geometries.

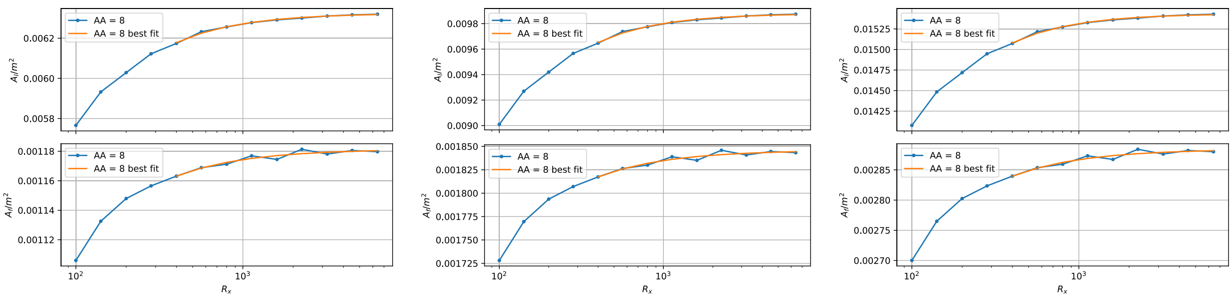
geometry	scaling factor	approximate blockage	draft/m
small	$\frac{4}{5}$	$0.1184\,\%$	0.0016
medium	1.0	$0.1850\,\%$	0.0020
large	$\frac{5}{4}$	$0.2313\,\%$	0.0025

Figure 16 – Lateral and frontal view of VLCC.

and the square root of the relative surface areas are collected in Table 5. The latter ones should be (and are) equal to the scaling factors as documented in Table 4. For the three hull geometries, the lateral to frontal surface area ratio approaches the same value.

Table 5 – Computed lateral and frontal surface areas A_l and A_f for the three geometries, the lateral to frontal ratio and the medium-to-small and medium-to-large ratios.

Geometry	A_l/m^2	A_f/m^2	A_l/A_f	$\sqrt{A_f/A_{f_{medium}}}$	$\sqrt{A_l/A_{l_{medium}}}$
small	$0.006\,324\,53$	$0.001\,181\,41$	5.3534	0.8000	$0.800\,00$
medium	$0.009\,880\,07$	$0.001\,845\,96$	5.3522	1.0	1.0
large	$0.015\,436\,48$	$0.002\,884\,32$	5.3519	1.2500	1.2500



(a) Small geometry. (b) Medium geometry. (c) Large geometry.

Figure 17 – Convergence of the frontal and lateral surface area computation as a function of the picture resolution.

As mentioned in the previous chapter, wind coefficients will also be computed for a simplified variant of the VLCC without gaps between the container stacks (shown in Fig. 9b). This geometry will have a different (larger) lateral surface area whose value is also required. The lateral surface area for this case equals $A_L = 0.010\,591\,2\,\mathrm{m}^2$ ($8\,\%$ larger than the value in Table 5), while for the frontal area computation, $A_F = 0.001\,846\,02\,\mathrm{m}^2$ is found. This last value differs only very little from the value displayed in Table 5. It is expected that due to the uninterrupted surface area, both the lateral force coefficient and the yawing moment will be significantly larger than for the case with gaps between the container stacks.

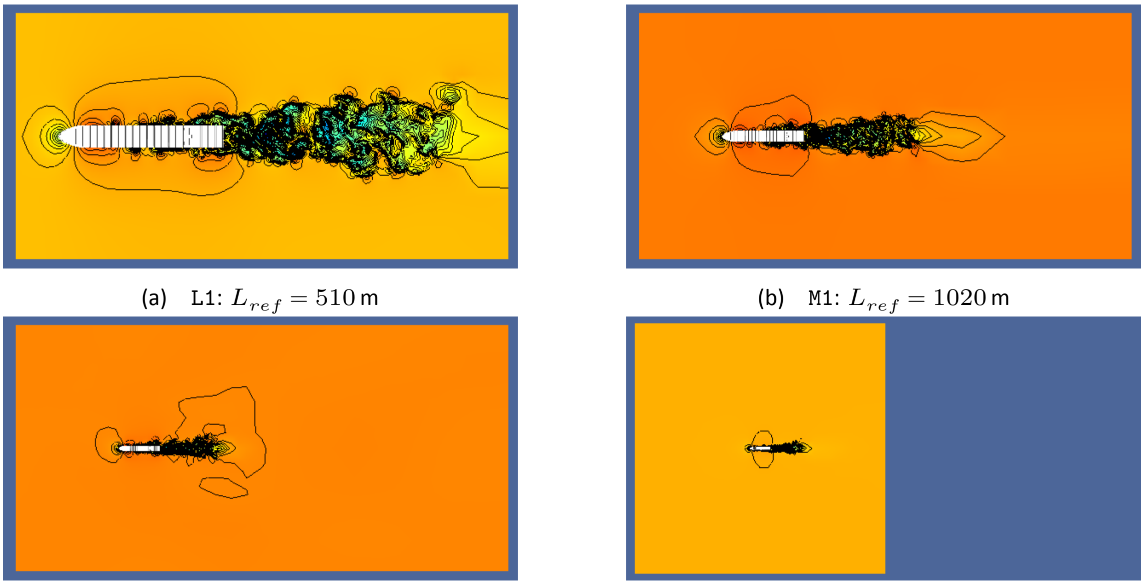
3.3 Parameter variations

3.3.1 Influence of blockage on the solution

In this section, computations are executed with the geometry scaled up and down by multiples of two in order to determine the influence of (lateral) blockage on the solution. If blockage is negligible and the cell size relative to the ship size is kept constant, resultant forces and moments should remain constant as well. Computations are executed for $\varphi = 0°$, $30°$, $60°$ and $90°$ with the container ship of Andersen (2013) (configuration 01-01-02).

To keep the overall simulation properties the same except for the size of the ship with respect to the domain box, the ship scaling factor[17] is adapted (from 1 to 2, 0.5 and 0.25). For convenience, the different sizes are named `L1` (Large), `M1` (Medium), `S1` (Small 1) and `S2` (Small 2), Fig. 18 shows a top view of the computational domains including the ship geometry and the velocity at the domain bottom. The blue colour at the edges of the figures is the background colour of the visualisation program GFSView. The blockages and lateral blockages for the 16 cases are shown in Fig. 19. The maximum refinement levels for the hull and wake are adapted accordingly (from 10 to 9, 11 and 12) as is the initial refinement in the entire domain (from 5 to 4, 6 and 7). The end time of the simulation is adapted as well to ensure that the full scale simulation time remains the same (from 10 to 20, 5 and 2.5). Furthermore, the start of the derefinement region aft of the hull is adapted such that for all cases, the wake length to ship length ratio remains equal (from 0.6 to 1.2, 0.3 and 0.15 for $\varphi = 0°$ and from 0.25 to 0.5, 0.125 and 0.0625 for $\varphi = 90°$).

[17] The ship scaling factor denoted here is the value by which the CAD geometry is scaled in the configuration files from the default geometry with a length of 0.333.



(a) L1: $L_{ref} = 510\,\mathrm{m}$

(b) M1: $L_{ref} = 1020\,\mathrm{m}$

(c) S1: $L_{ref} = 2040\,\mathrm{m}$

(d) S2: $L_{ref} = 4080\,\mathrm{m}$

Figure 18 – Top view of the computational domain lat $T' = 1020\,\mathrm{s}$ for four ship scales L_{ref} at $\varphi = 0°$ showing the relative size of the ship compared to the domain size. Domain bottom is coloured with velocity magnitude.

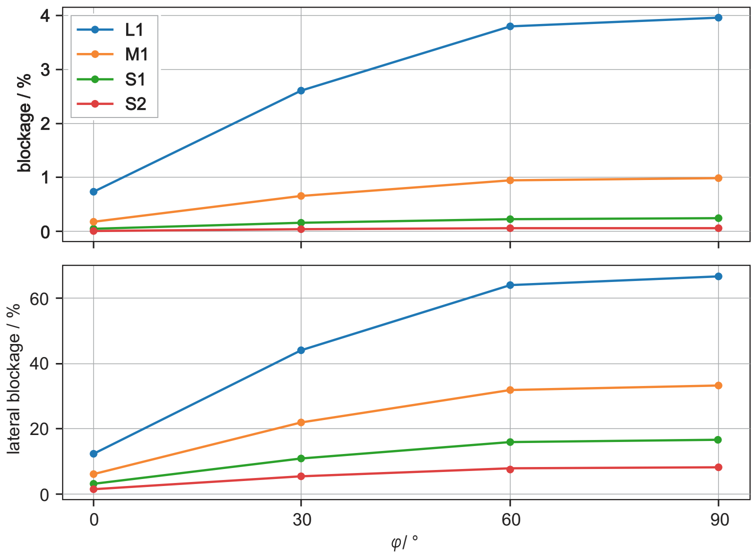


Figure 19 – Blockage and lateral blockage as a function of the wind angle φ and the scale of the ship w.r.t. the domain box.

For the dimensionalisation of the forces and moments, L_{ref} is adapted from 1020 m to 510 m, 2040 m and 4080 m. For the smallest ship size, the domain geometry was modified to remove the aft box resulting in a cubic computational domain (see Fig. 18d). Due to the small size of the vessel with respect to the domain for that case, the currently used settings to partition the computation over multiple processors gives a significant imbalance in processor load which leads to longer computing times. By removing the aft box from the domain, the imbalance is reduced.

For $\varphi = 0°$, the time histories of $C_{X_{AF}}$, C_Y, C_K and C_N for the different ship sizes are shown in Fig. 78 in Appendix A5. The red line in each graph depicts the cumulative moving average (CMA) that is computed to judge the steadiness of the results. The last value of the cumulative moving averages (i.e. the time-averaged value of the signal for which the CMA is computed.) are presented in Fig. 20 for each of the coefficients together with the standard deviation of the signals. Due to lateral symmetry of the hull for this wind angle, $C_{X_{AF}}$ is the only coefficient with a value that deviates significantly from zero (≈ -0.51). For the largest ship model, the average value of $C_{X_{AF}}$ deviates more from the average values of the smaller hulls than the maximum of their standard deviation. (Fig. 20). It is concluded than for this wind angle, blockage does not play a significant role for the three smallest ship models.

For $\varphi = 30°$, the average values of the coefficients are depicted in Fig. 21. The time traces from which these averages were computed are displayed in Fig. 79 in Appendix A5. Compared to the $\varphi = 0°$ case, the low frequency vortex shedding is not present any more and all coefficients attain non-zero values. The longitudinal force $C_{X_{AF}}$ has increased approximately $10\,\%$. For this case, blockage seems to influence the medium (`M1`) case as well: now only the values obtained with `S1` and `S2` are equal, although the difference with `M1` is not very big. Averaged values for the largest ship (`L1`) start to deviate significantly from the rest of the averages.

For $\varphi = 60°$ and $90°$ the averaged values together with their standard deviations are presented in Figs. 22 and 23. The time traces are again shown in Appendix A5 (Figs. 80 and 81). Similar trends are obtained as for $\varphi = 30°$, although now, the deviations for the results of `M1` are larger, especially for $\varphi = 90°$. This indicates that for this ship size, the default ship-to-domain size as discussed in § 2.1.1 is actually too large for blockage to be negligible. The trends of $C_{X_{AF}}$ and C_N are different from all other trend lines for these two ship orientations: the deviation of `M1` is in the opposite direction as compared to the deviation of `L1` with respect to the near constant values of the smallest two hull sizes. A top view of the domain for $\varphi = 60°$ (where the bottom is coloured with velocity magnitude) at the last time step in the simulations is shown in Fig. 24. The maximum velocity in the domains for the four cases show very similar trends: they are practically the same for the two smallest ship sizes (1.85 m/s), while for `M1`, the maximum velocity is 1.89 m/s and for `L1`, it is close to 1.97 m/s. These numbers also give an indication about the influence of blockage.

For validation computations, the ship (or object) should be scaled down to ensure that blockage is negligible. For an object with similar aspect ratios as the VLCC of Andersen, its length should definitely be less than one third of the domain width, preferably one sixth (similar to `S1`).

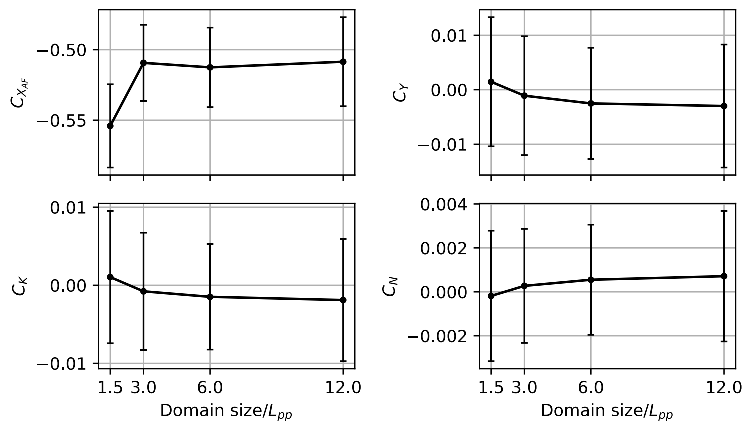


Figure 20 – Average values and standard deviations of force and moment coefficients for the VLCC hull as a function of the ship length relative to the domain cross section for $\varphi = 0°$.

3.3.2 Influence of the maximum refinement in the wake

The maximum refinement level in the wake (N_w) compared to the maximum refinement level on the solid object (N_s) may affect the solution as well. In the *Tangaroa* example that is part of the standard examples of

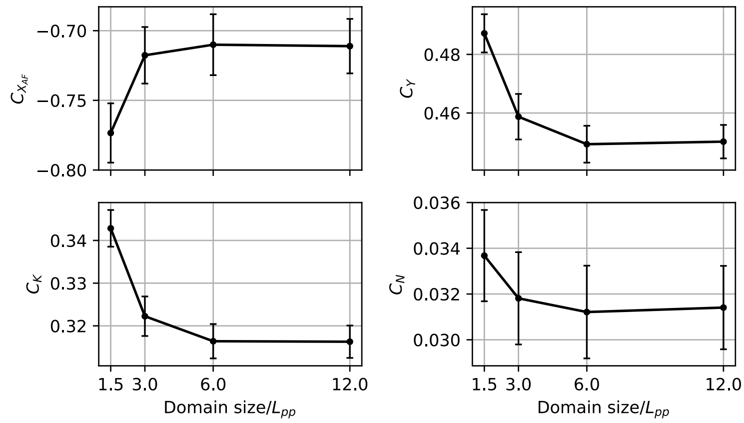


Figure 21 – Average values and standard deviations of force and moment coefficients for the VLCC hull as a function of the ship length relative to the domain cross section for $\varphi = 30°$.

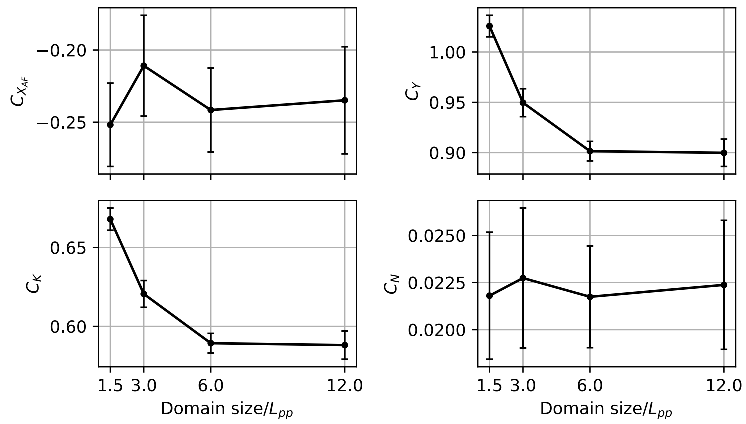


Figure 22 – Average values and standard deviations of force and moment coefficients for the VLCC hull as a function of the ship length relative to the domain cross section for $\varphi = 60°$.

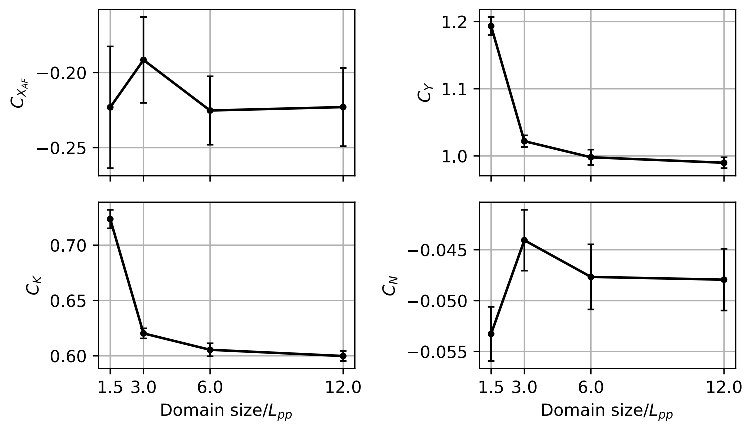
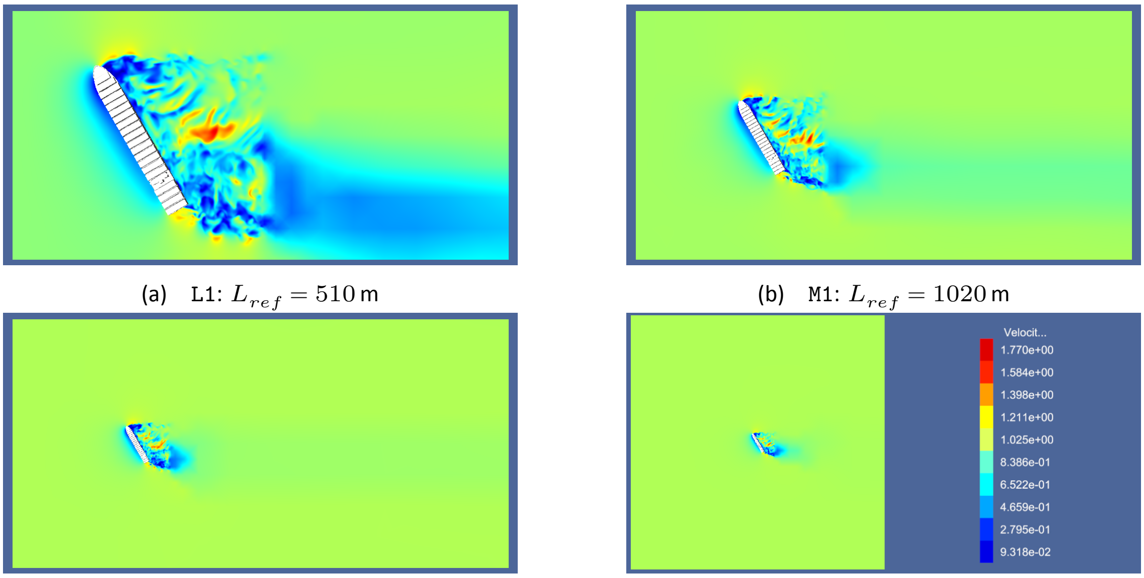


Figure 23 – Average values and standard deviations of force and moment coefficients for the VLCC hull as a function of the ship length relative to the domain cross section for $\varphi = 90°$.



(a) L1: $L_{ref} = 510\,\mathrm{m}$

(b) M1: $L_{ref} = 1020\,\mathrm{m}$

(c) S1: $L_{ref} = 2040\,\mathrm{m}$

(d) S2: $L_{ref} = 4080\,\mathrm{m}$

Figure 24 – Top view of the computational domain at $T' = 1020\,\mathrm{s}$ for four ship scales L_{ref} at $\varphi = 60°$ showing the relative size of the ship compared to the domain size. Domain bottom is coloured with velocity magnitude.

Gerris[18], the maximum refinement in the wake is one level less than the maximum refinement on the solid object. However, nothing prevents one from assigning the same level of refinement as used for the solid to (a part of) the wake. This is investigated with a wall-mounted cube (edge length 0.05) placed at the origin. With these dimensions, blockage is so low ($0.25\,\%$) that it can be neglected. The refinement level is maintained for six cube edge lengths aft of the cube, after which it is reduced by one level for four edge lengths. This means that the total wake length is 10 times the length of the cube's edge. With a constant refinement level on the cube, the duration of these computations mainly depends on the total cell count. By default, computations are run for 10 seconds.

The solid body refinement level N_s is varied between 8 and 11 and for each of these, the wake refinement level N_w is varied between 7 and N_s. This gives two computations with $N_s = 8$, three computations with $N_s = 9$, four computations for $N_s = 10$ and five computations with $N_s = 11$. For the latter one, there were issues with the computation with 9 levels of wake refinement: this case crashed[19] after about $4.6\,\mathrm{s}$. The case with 11 refinement levels in the wake was not run due to the high computing time of the case with 10 refinements in the wake (Fig. 25). Results are shown in Fig. 26, where the values of the coefficients $C_{X_{AF}}$, C_Y, C_K and C_N are displayed as a function of the maximum refinement level in the wake, for each of the four solid body refinement levels. Looking at the same data in another way, Fig. 27 shows the same data as a function of the solid body refinement level, for each of the maximum refinement levels in the wake. Due to the symmetry of the body with respect to the oncoming flow, $C_{X_{AF}}$ is the only coefficient with non-zero values. For this particular case, Blendermann (2013, p. 141, Fig. D.1b.) reports an experimental value for $C_{X_{AF}}$ of 1.2. When the refinement level on the body is increased from 9 to 11, $C_{X_{AF}}$ values converge towards a value of 1.2. For the current case, it seems that a refinement level of 11 on the body is required for $C_{X_{AF}}$ to be in the general neighbourhood of the experimental reference value. Looking at the values of $C_{X_{AF}}$ in Fig. 26, it is clear that the cases for $N_s = 8$, 9 and 10 all display a increasing trend as the refinement level in the wake is increased. For the highest level of refinement on the body ($N_s = 11$), this is not the case: the four results have values whose average is close to 1.2. From this observation, one could deduce that the maximum refinement level on the object is the most important parameter to get accurate results: the maximum refinement level in the wake is less important. But, as long as the refinement level on the body is too low, increasing the refinement level in the wake gives results that are closer to the experimental value. By dividing the characteristic dimension

[18]See: `http://gerris.dalembert.upmc.fr/gerris/examples/examples/tangaroa.html`

[19]Crashed is not really the correct word as the computation seemed to have entered an infinite loop and had to be stopped manually.

of the object by N_s a recommended minimum refinement level for an object with different dimensions can be computed. For the current cube, the characteristic size relative to the smallest cell[20] equals 102. In other words, the cube is discretised by approximately $10\,000$ cells on each of its five exposed sides. For an object that is twice as large (linear dimension equal to 0.1), 10 refinements are required to get similar refinements on the object as for the cube under investigation here.

For the computations with N_s = 11, the computing times for the different refinement levels in the wake are shown in Fig. 25. The significant increase in computing time with small cell sizes in the wake shows that results can be obtained in a shorter time frame by using larger cells in the wake region. Another way to reduce computing times is by shortening the region aft of the object where the wake is captured. This will be investigated in the next section.

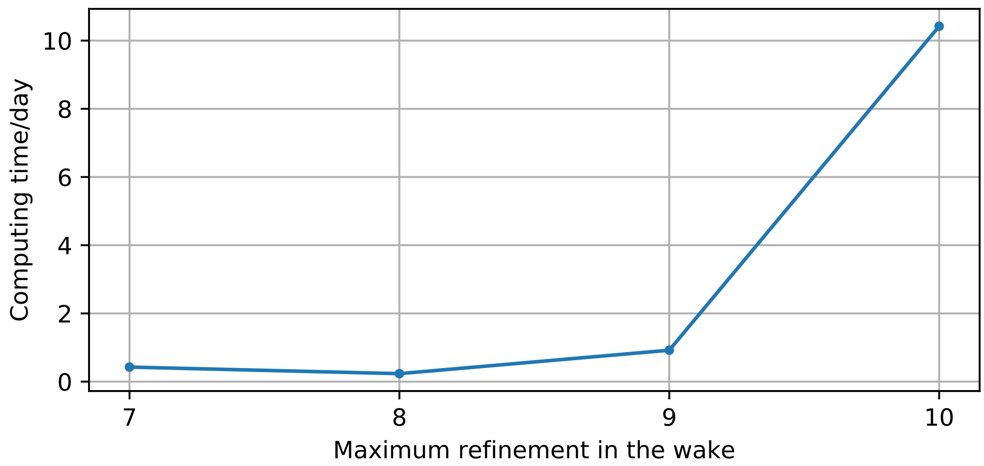


Figure 25 – Computing time with N_s = 11 as a function of the maximum refinement in the wake.

3.3.3 Influence of the wake length

The results in the previous section were obtained in a setup where the wake aft of the cube is significantly longer than the size of the cube itself (see Fig. 28). When the wake region is cut off too close to the object, large-scale low-frequency vortical structures may not be able to develop properly. To investigate the influence of the cut-off distance, computations are executed where this variable is the only one that is modified. The wall mounted cube with 11 refinements on the cube used in the previous section is modified by changing the distance aft of the cube beyond which grid coarsening is applied.

Five extra cases were configured where the cylinder radii that demarcate the boundaries of both wake derefinement zones were reduced. The inner radius (which demarcates the region between 9 and 8 refinements, was reduced in integer multiples of the cube length from 5 to 1.

In addition, a very short wake with length 0.5 was also included. The outer radius (that demarcates the region between 8 refinements and maximum derefinement), was proportionally reduced to keep the same ratio (10/6). The values for the two radii are shown in Table 6.

The averaged values and standard deviations of the resultant forces and moments on the cube are displayed in Fig. 29, where the average of the different values is shown with a red line. The graphs shown that the resultant forces on the cube are hardly affected by the wake length. For $C_{X_{AF}}$, the results for l_w = 1 and 6 deviate from an almost linear trend with a very small slope: for shorter wakes, the longitudinal force is somewhat larger than for longer wakes. Note that the computation with the longest wake only ran for four seconds, while the other

[20]The cell size as a function of refinement level S (in a domain with unit sides) equals $1/2^S$.

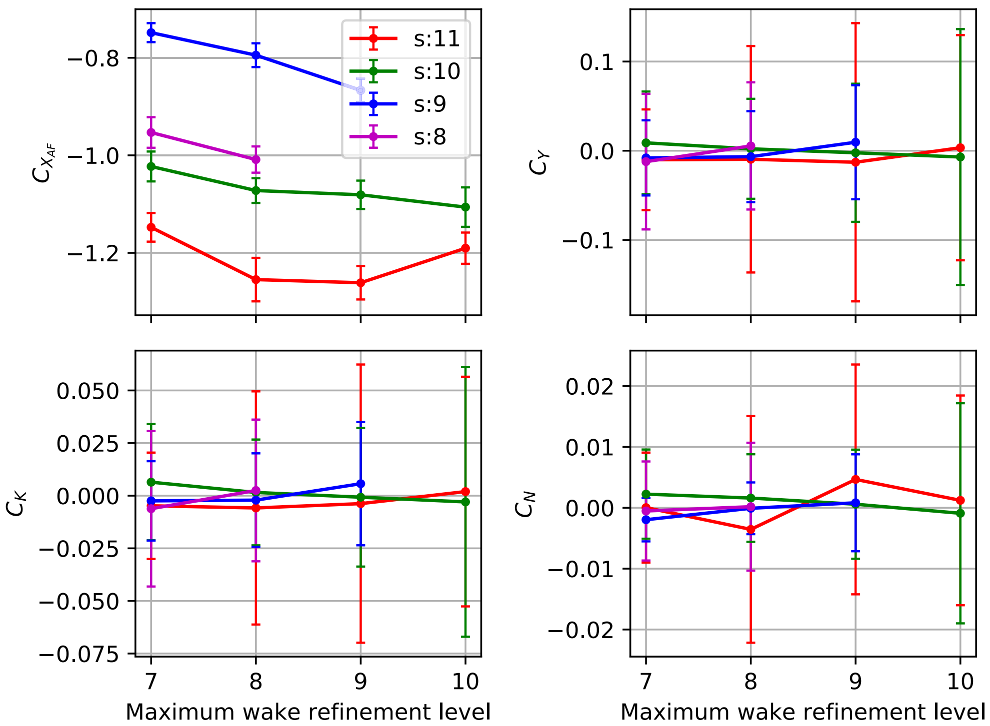


Figure 26 – Influence on the force coefficients of the refinement levels on the cube and in the wake.

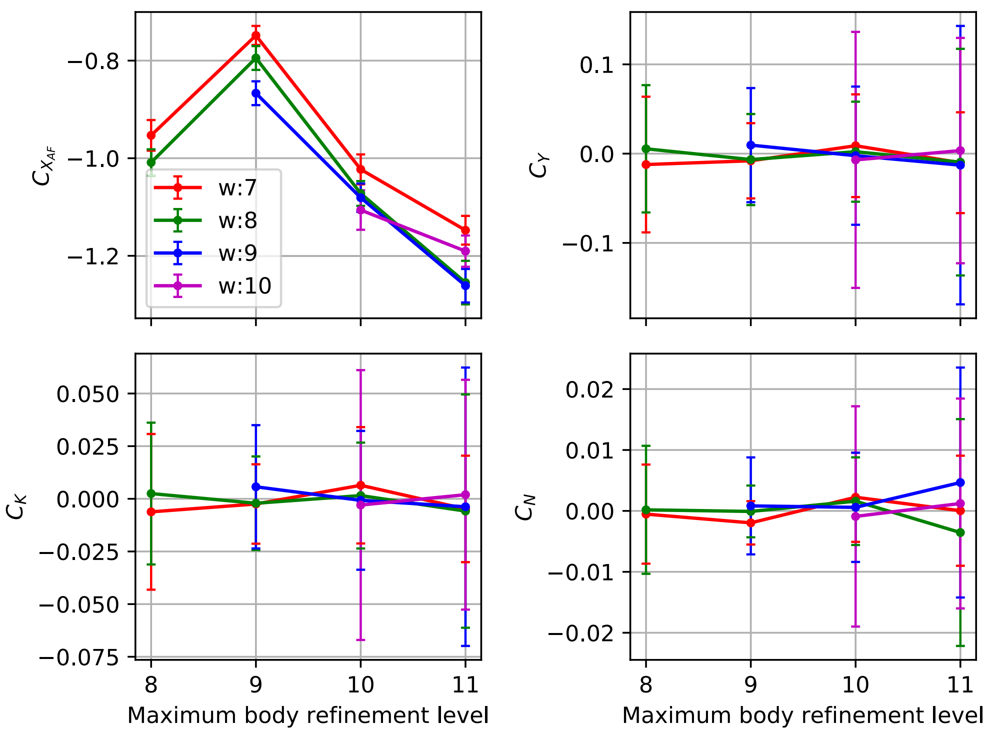
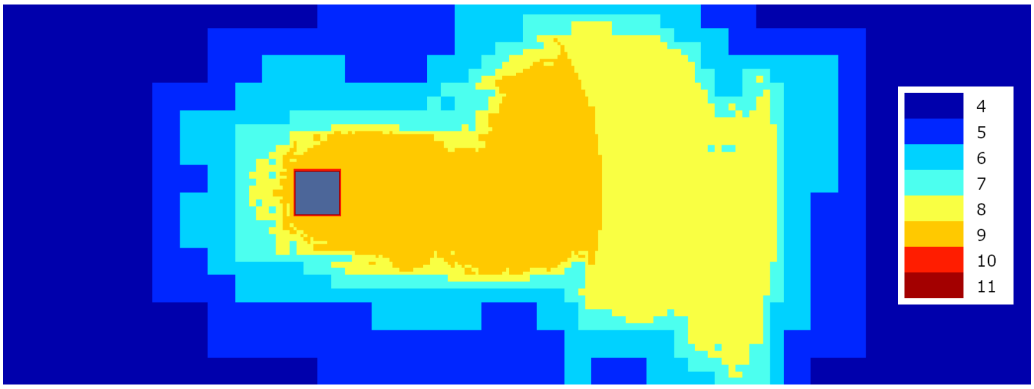
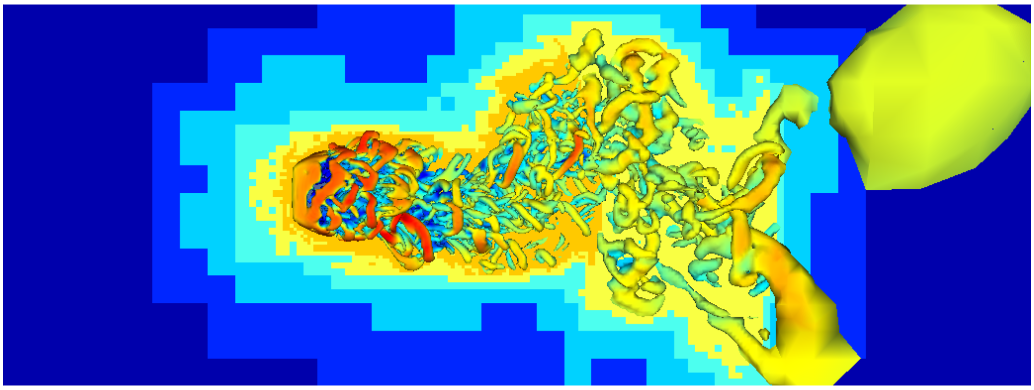


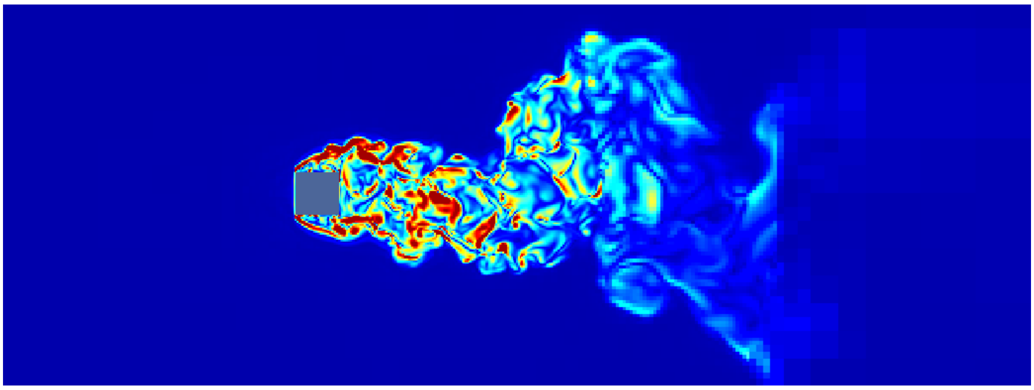
Figure 27 – Influence on the force coefficients of the refinement levels on the cube and in the wake.



(a) Refinement level on the bottom wall.



(b) Vortical structures in the wake ($\lambda 2 = -0.001$).



(c) Vorticity.

Figure 28 – Refinement level on the bottom wall, vortical structures in the wake and vorticity on the bottom wall for the wall-mounted cube ($N_s = 11$, $N_w = 9$).

Table 6 – Radii of wake derefinement zones.

wake length (WL)	inner radius	outer radius
6 (§ 3.3.2)	0.325	0.525
5	0.275	0.444
4	0.225	0.3635
3	0.175	0.2825
2	0.125	0.202
1	0.075	0.121
0.5	0.05	0.0834

computations ran for 10 seconds. Hence, the averaging period for the former computation is shorter, which could explain the deviation from the trend. For the lateral force C_Y and the roll moment C_K, deviations from the trend line reduce as the wake length is reduced. For these seven simulations, a top view of the cube and the bottom wall is shown in Fig. 30, which shows the cell refinement level on the bottom wall and an isosurface of pressure coloured with velocity magnitude. The shape of the bulk of the pressure isosurface is the same for all cases, although somewhat shorter for the two cases with shortest wakes, due to the grid coarsening. One can conclude that for this case, the length of the wake aft of the cube does not need to be very long, which has a positive effect on the computing time.

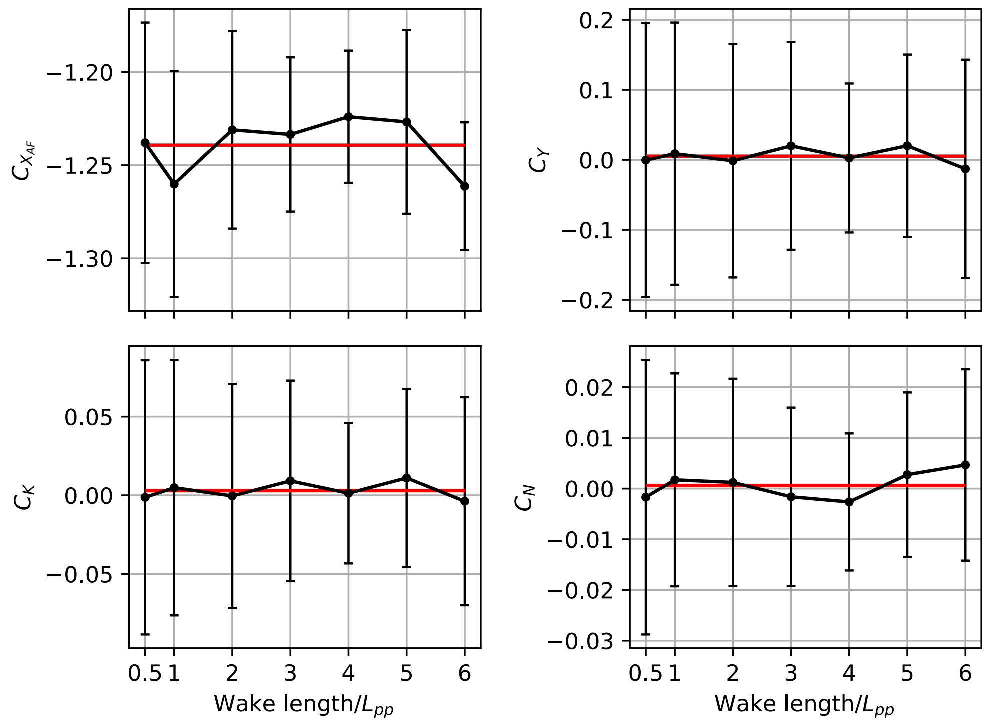


Figure 29 – Influence of the wake length on the resultant forces coefficients on the cube for N_s = 11 and N_w = 9.

3.3.4 Computing times and parallel efficiency

By distributing independent parts of a computation over multiple processors, a significant reduction in computing time can be achieved. When too many processors are used for a computation of a specific size, the inter-process communication will start to dominate the computing time, reducing the efficiency up to the point where adding more processors is no longer useful.

The parallel efficiency of Gerris has been evaluated for a configuration with a rectangular prism that is four times as long as its height and width. Computations were run at $\varphi = 0°$ for four seconds using 1, 2, 4 and 8 processors of a single node of the Stokes queue. The relative speed-up compared to the single-core computation and the absolute wall clock times are reported in Fig. 31. With four processors, the speed-up with respect to a computation run on a single processor equals three. For eight processors, the overhead increases significantly, because now, only a four-fold speed-up is achieved. In absolute numbers, the computing time reduces from 40 hours to eight hours. For this configuration, a computation run for 10 seconds would finish in 20 hours on eight cores, while the serial computation would require 80 hours.

There seems to be an issue with the parallellisation of Gerris computations when more than eight processors are requested. The standard graph partitioning algorithm (recursive bisection) sometimes fails to return a valid

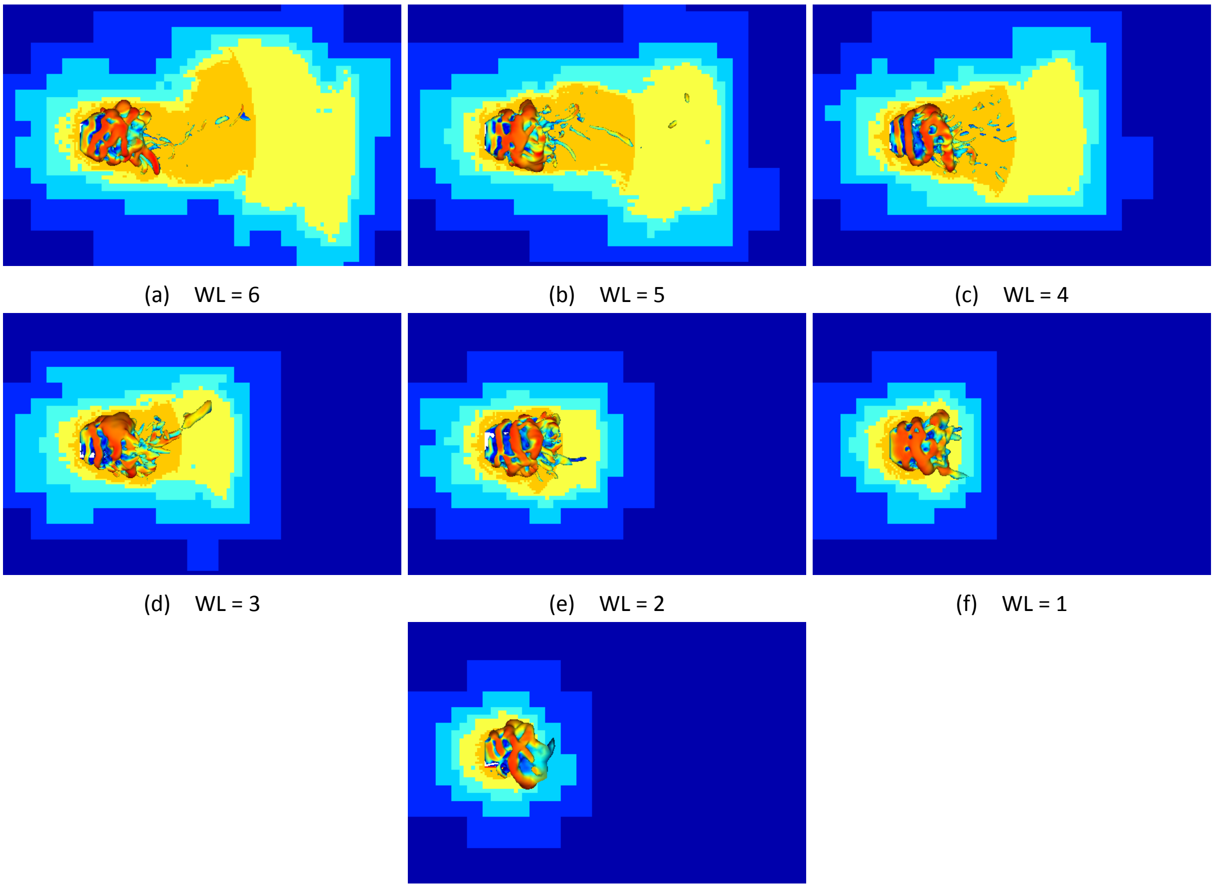


Figure 30 – Top view of wall-mounted cube with isosurface of pressure coloured with velocity magnitude and refinement level on the bottom surface for different wake lengths.

simulation file when 16 processors are requested. The software suggests to use the simple bubble partitioning algorithm as an alternative (which is also recommended when AMR is used) which results in a valid partitioned simulation file. However, while the simulations are running, load balancing does not appear to work as it should (with very large standard deviations for the average number of cells per processor) which eventually results in crashes.

3.4 Grid convergence study

3.4.1 Introduction

In this section, the convergence characteristics of Gerris are examined. In general, grid convergence studies are executed by uniformly (de)refining a grid with a certain constant factor such that topologically similar coarser and/or finer grids are created. For multiple reasons, this is not possible with a solver like Gerris. Firstly, by simply increasing and decreasing the maximum refinement levels with one, the differences in the grid sizes are too large: different meshes in the grid convergence study will differ by a factor eight in size, which is generally considered too much for a grid convergence study[21]. Secondly, Gerris automatically adapts the grid to the features of the flow. When the maximum refinement level is increased, progressively smaller features in the flow are resolved (down to the Kolmogorov scale in a DNS computation) which foregoes the purpose of

[21]ITTC recommends a grid refinement ratio $r_G = \sqrt{2}$ (ITTC, 1999)

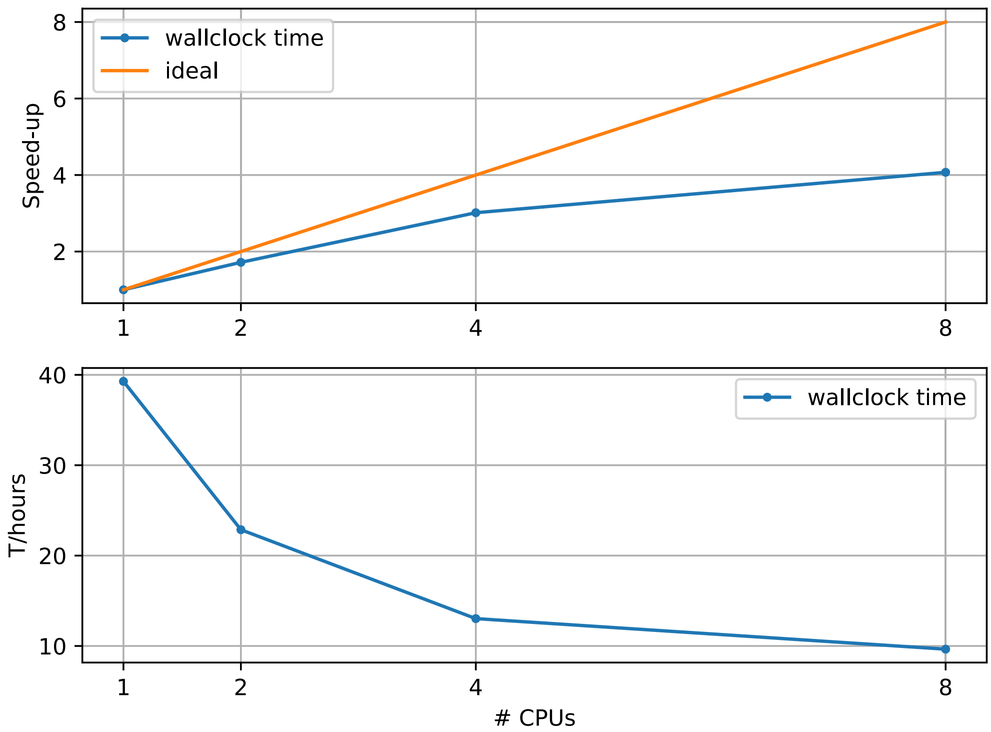


Figure 31 – Parallel scalability of Gerris

a grid convergence study where one wants to verify the convergence of the solver for a case around a specific grid size. Automatic grid adaptation as used in Gerris also hampers a formal grid convergence study where grids with a fixed number of cells are utilized in the convergence study.

For these reasons, a slightly different approach is used. Instead of changing the maximum refinement levels to get coarser and finer meshes, the object in the domain is scaled up and down by a small factor. The effect of this is that the relative size of the cells is different from the base computation: when the object is scaled down with a factor 0.8, the cells will be proportionally larger with a factor 1.25. The same holds true when the object is scaled up with a factor 1.25: relative to the object, all cells will be proportionally smaller with a factor 0.8. Obviously, one should ensure that a change in blockage does not affect the solution. For $\varphi = 0°$, this is not considered to be an issue given the very low blockage values, but with higher values of φ, this may not be true. The effect of blockage on the solution has been investigated earlier in section 3.3.1 and the conclusion was that the width of the object should be smaller than one third of the domain width for a minimal influence of the blockage near φ = $90°$, and preferably close to one sixth of the domain width to ensure that blockage effects are negligible.

To ensure that the particular geometry does not affect generic conclusions about the solver's convergence characteristics in a negative way, one of the rectangular blocks as tested by Blendermann (2013) will be used for this purpose (these cases will be covered in more detail in section 4.1). Some of the small (and thin) details of the VLCC geometry (such as the deck railing at the bow and the gaps between container stacks) may be too small to draw generic conclusions about the solver convergence characteristics. This is also noted in Popinet (2003) where it is stated that geometries should not contain features with spatial scales smaller than the mesh size. On the other hand, results that show non-converging trends may be an indication that the refinement level at the object is not high enough to properly resolve the geometry. Hence, the main focus will be on the convergence characteristics of the solver for simple (but relevant) geometries. For the current investigation, the rectangular block with length-to-breadth ratio of $L/B = 4$ will be used first. Afterwards, results of a grid convergence study with the VLCC will be presented. For all cases, results will be presented for $\varphi = 0°,\ 30°,\ 60°$ and $90°$.

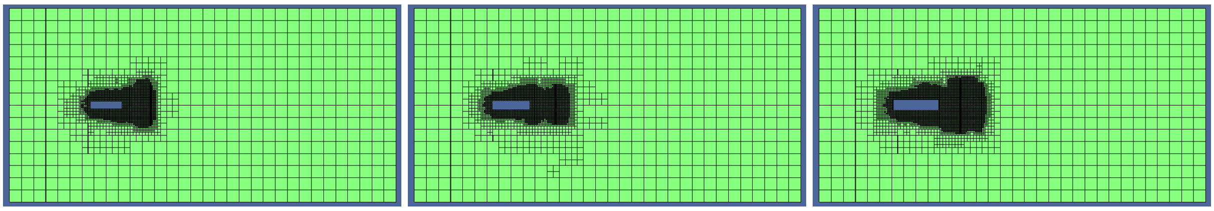
A formal grid convergence analysis will not be undertaken, only trends will be judged.

3.4.2 Rectangular cuboid $(L/B = 4)$

3.4.2.1 Setup

The setup of this case is fairly simple. The block length is one fifth of the domain width, while the block height is one twentieth of the domain height. For $\varphi = 90°$, this results in a blockage of $1\,\%$ for the computation on the medium grid. As explained before, the size of the block is adjusted instead of the maximum refinement level. For the fine grid, a scale factor of 1.2 gives a blockage[22] of $1.44\,\%$, while for the coarse grid, a scale factor of 0.8 is used (with a blockage of $0.64\,\%$). These values are low enough to ensure that blockage does not affect the solutions. The wake refinement region is scaled with the same factor used to scale the block for the fine and coarse cases, ensuring a wake refinement region that is equal in size for the three cases compared to the block length. The maximum refinement level in the wake is set to nine and the refinement level at the solid is set to ten. Computations were run for 10 s for $\varphi = 0°$ and 20 s for the other angles.

A top view of the domain bottom at the last time step of the computations for the three cases for $\varphi = 0°$ is shown in Fig. 32. The grid refinement to capture the wake is shown up to level 8.



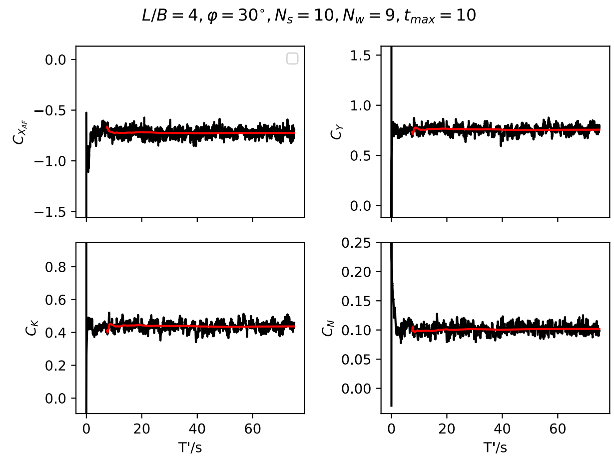
(a) Coarse (b) Medium (c) Fine

Figure 32 – Top view of the computational domain bottom for the grid convergence study of the rectangular cuboid at $\varphi = 0°$.

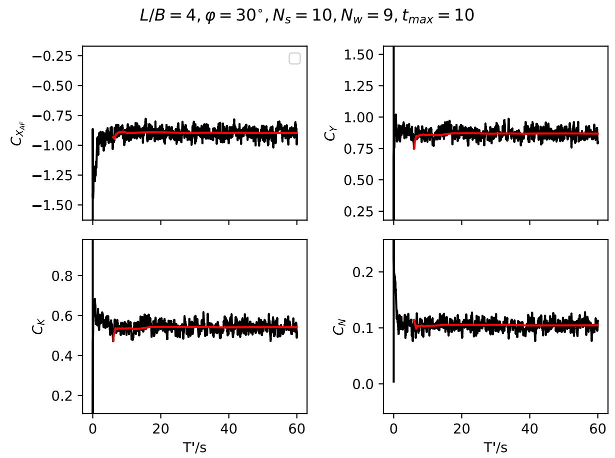
3.4.2.2 Results

Time histories are shown in Fig. 33 for $\varphi = 30°$ for the coarse, medium and fine grids. The red lines show the cumulative moving average which gives an indication of the convergence of the average as a function of time. For all four wind angles, results for the forces and moments are shown in Fig. 34 including the standard deviations of the computed average values. The horizontal axis in these figures (grid density) is the relative size of the cells with respect to the object: 1.2 corresponds to a fine grid (larger object) while 0.8 corresponds to a coarse grid (small object). Except for the yawing moment for $\varphi = 60°$, all quantities that should converge to a non-zero value in the limit show convergence, where the majority shows monotonic convergence (e.g. C_Y for $\varphi = 30°$, $60°$ and $90°$) and a minority displays oscillatory convergence (e.g. C_K for $\varphi = 30°$).

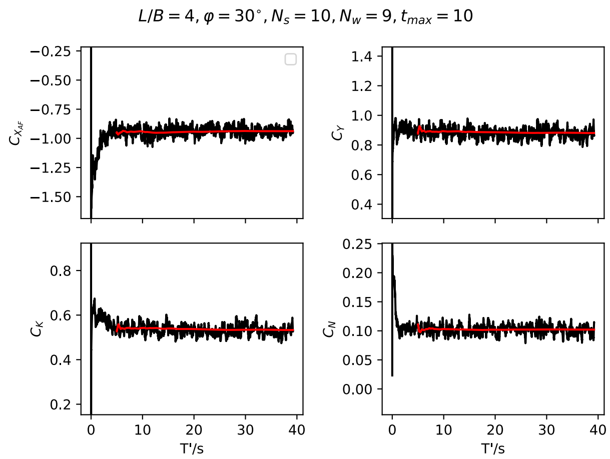
[22] Initially, a scale factor of 1.25 (the reciprocal of 0.8) was used for the fine grid, but this resulted in crashes of the solver after a few time steps into the computation.



(a) Coarse



(b) Medium



(c) Fine

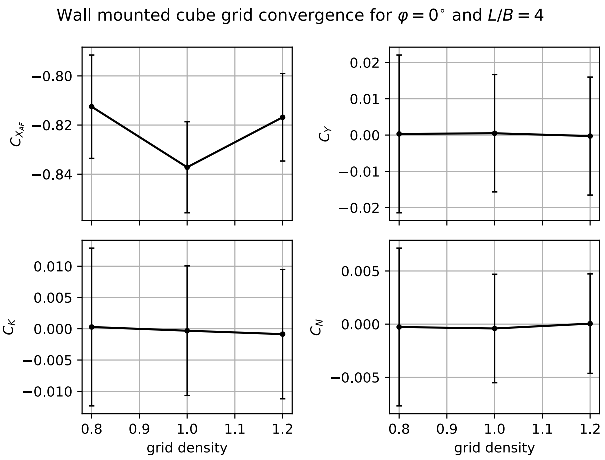
Figure 33 – Time histories of forces and moments for the rectangular cuboid ($L/B = 4$) for $\varphi = 30°$ for the coarse, medium and fine grids.

3.4.3 VLCC

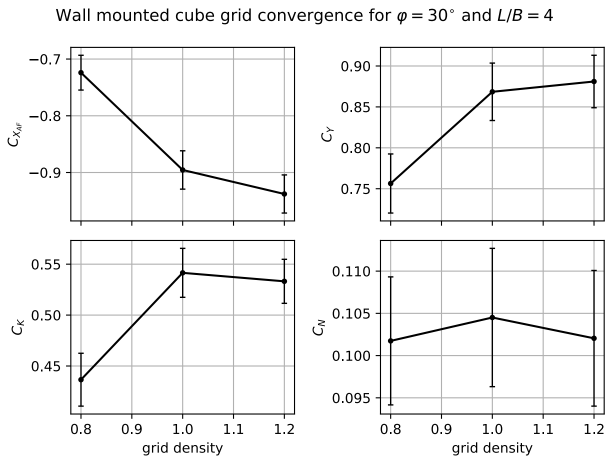
3.4.3.1 Setup

Based on the conclusions drawn in the previous chapter regarding the maximum size of the hull relative to the domain width to neglect blockage, the ship length is set to $\frac{1}{6}$ of the domain width for the medium size. The hull length for the coarse grid is smaller at $\frac{2}{15}$ of the domain width (≈ 0.1333), while for the fine grid, the hull size is larger at $\frac{5}{24}$ (≈ 0.208). Fig. 35 shows a top view of the domain, the hull geometry and the velocity magnitude on the bottom surface for $\varphi = 30°$ for the current ship geometry. By scaling the hull down with a factor two as compared to the previous chapter, the maximum refinement level has to be increased by one to maintain the same relative cell sizes. Hence, at least 12 refinement levels are required. As a consequence, the time step based on the Courant number will be halved as well. The dimensional time step goes up with a factor two, and thus, the total time for the simulations can be reduced by a factor two. Due to the relative small size of the hull compared to the domain width, the aft refinement box is removed for these cases.

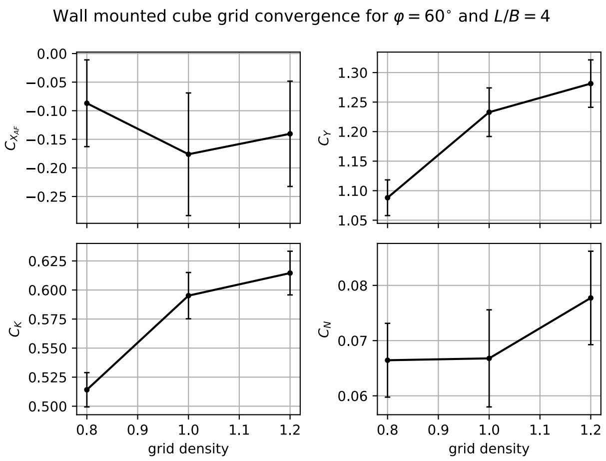
Computations are run for 5 s for $\varphi = 0°, 30°$ and $60°$, while for $\varphi = 90°$, the computations last for 10 s. The maximum refinement level on the hull is set to 12 and in the wake, it is 10.



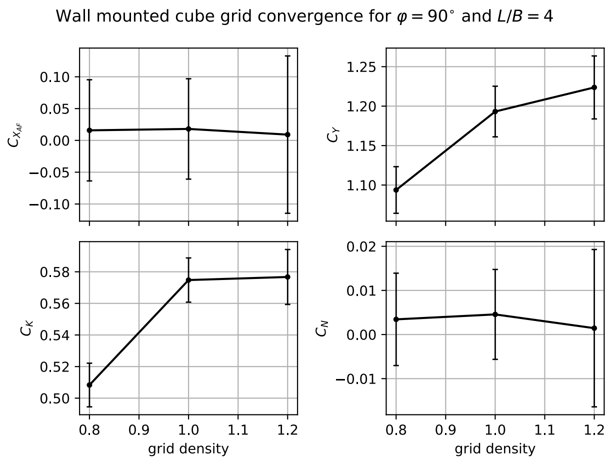
(a) $\varphi = 0°$



(b) $\varphi = 30°$

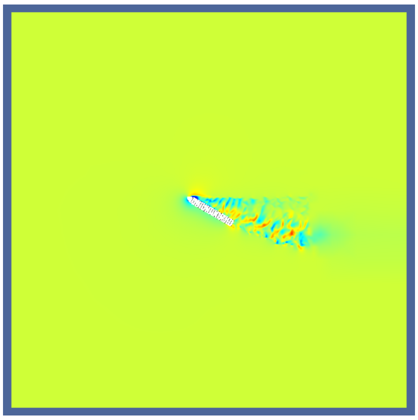


(c) $\varphi = 60°$

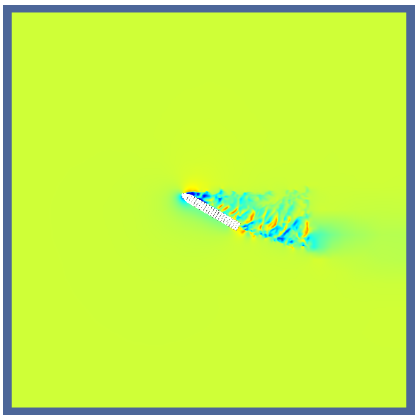


(d) $\varphi = 90°$

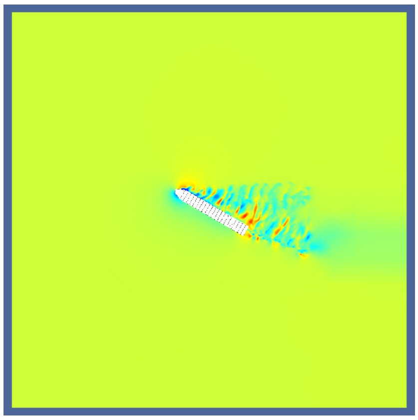
Figure 34 – Grid convergence for the rectangular cuboid ($L/B = 4$) for $\varphi = 0°, 30°, 60° and 90°$. The grid density value of 0.8 corresponds to a scaled-down cuboid (coarse grid), while 1.2 corresponds to a fine grid.



(a) Coarse



(b) Medium



(c) Fine

Figure 35 – Top view of the computational domain, showing the position and size of the container ship and the velocity magnitude on the domain bottom for the three vessel sizes.

3.4.3.2 Results

Time histories are shown in Fig. 36 for $\varphi = 30°$ for the coarse, medium and fine grids. The red lines show the cumulative moving average which gives an indication of the convergence of the average as a function of time. For all four wind angles, results for the forces and moments are shown in Fig. 37 including the standard deviations of the computed average values. Unlike in the case of the rectangular cuboid discussed in § 3.4.2, a reasonable portion of the cases do not show (oscillatory or monotonic) convergence. Notable are C_Y for $\varphi = 30°$ and C_Y and C_K for $\varphi = 90°$. For $\varphi = 60°$, all components show converging behaviour. A possible reason for this lack of convergence is the maximum refinement level at the hull (12) that was used for the current investigation. As noted in Chapter 2, the maximum refinement level should result in cells smaller than the smallest geometric detail. Along the sides of the hull, a small ridge is present that connects the base of the container stacks with the side of the hull. This ridge has a width of approximately $0.000\,199$, which – according to Table 2 – is smaller than the cell size associated with subdivision level 12: 2.441×10^{-4}. Ideally, the grid convergence study should be executed anew with a higher solid refinement level. An alternative would be to remove the small ridge by aligning the sides of the container stacks with the hull sides. This is left as a recommendation for future research.

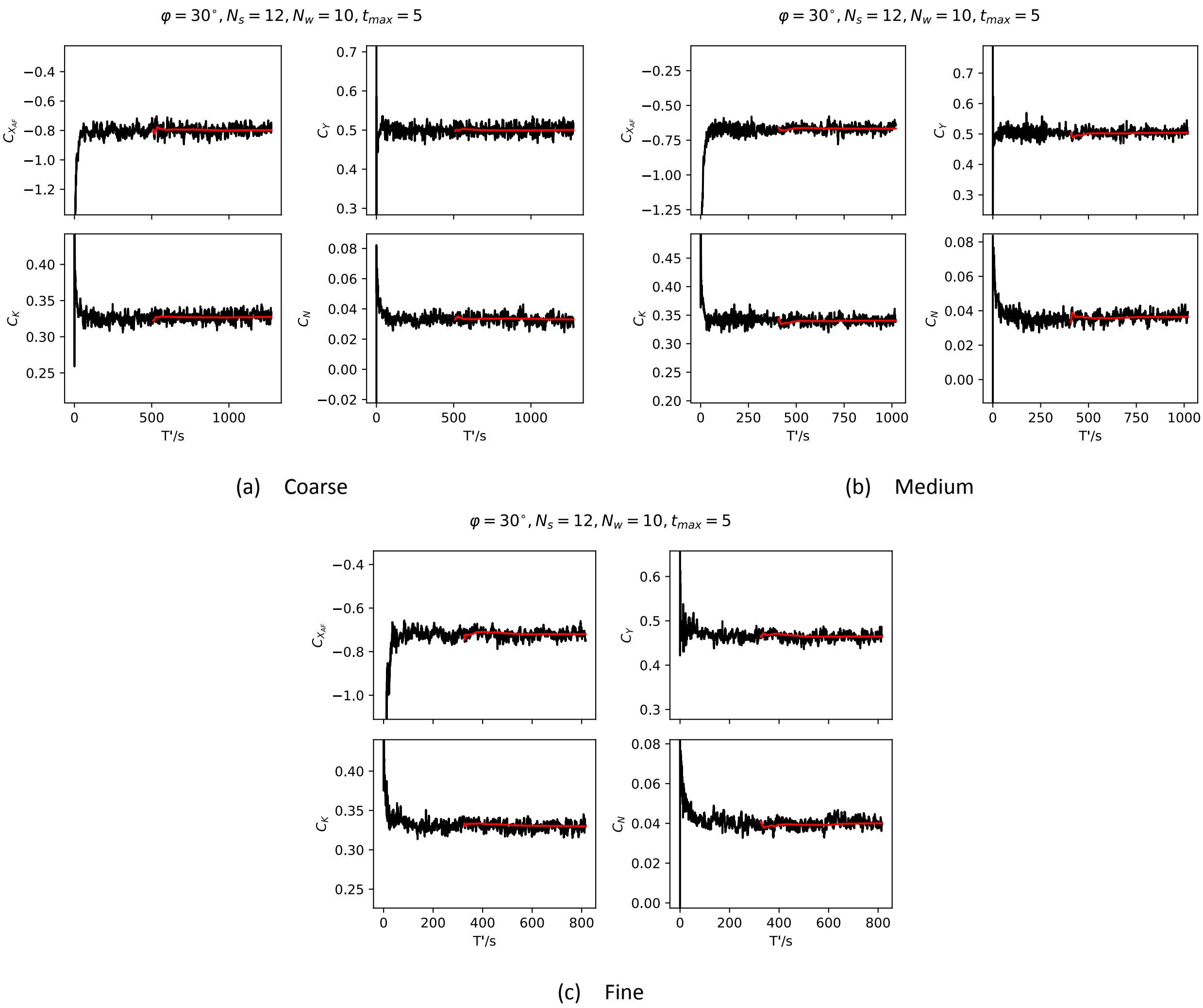
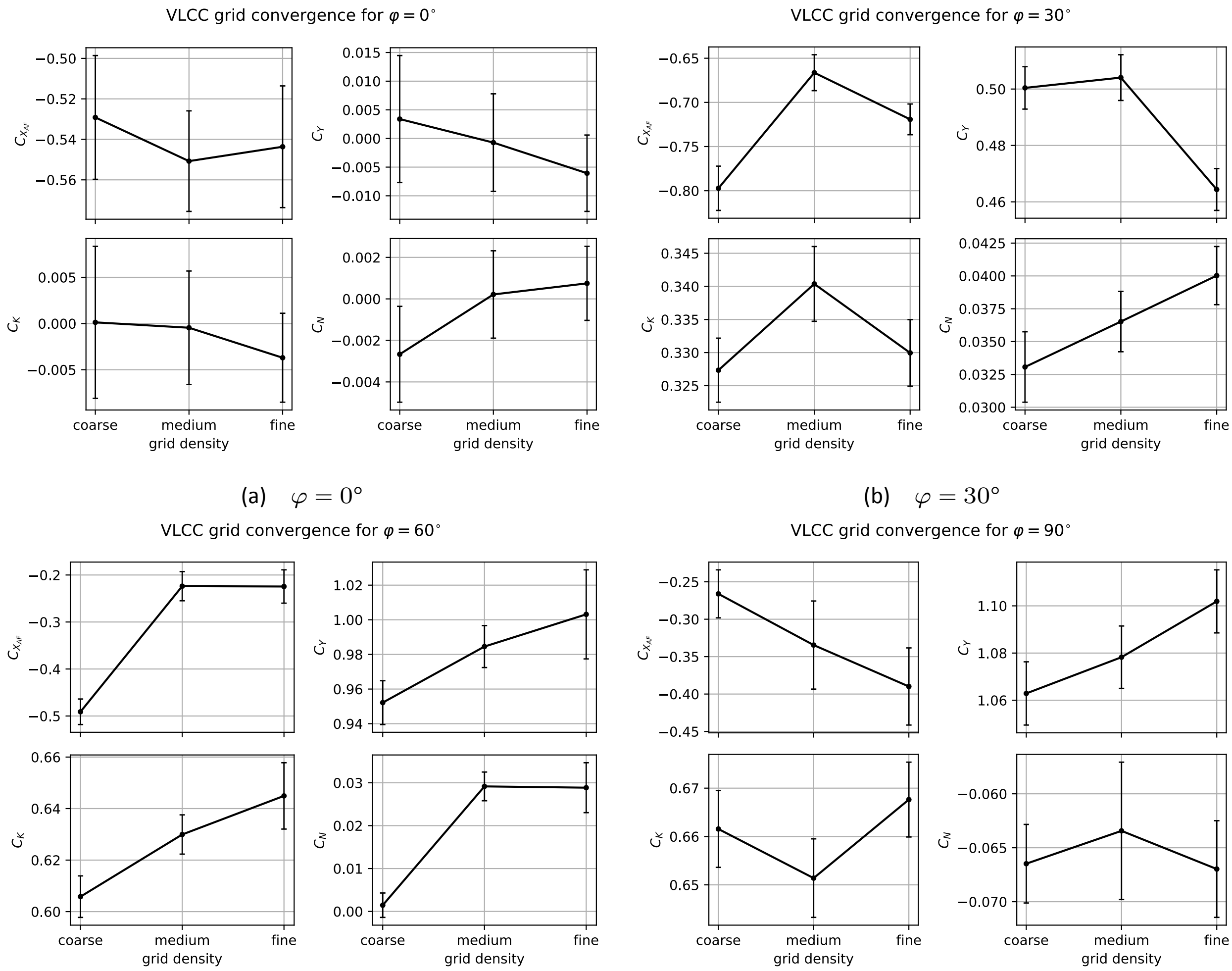


Figure 36 – Time histories of forces and moments for the VLCC of Andersen (2013) for $\varphi = 30°$ for the coarse, medium and fine grids.



(a) $\varphi = 0°$

(b) $\varphi = 30°$

(c) $\varphi = 60°$

(d) $\varphi = 90°$

Figure 37 – Grid convergence for the VLCC for $\varphi = 0°, 30°, 60° and 90°$.

4 Validation

In this chapter, some reference cases from literature will be recreated in Gerris and the resulting forces and moments will be compared with reference values. The knowledge gained in the previous chapter is used to optimise the computational setup, i.e., to get a good agreement with reference data in a reasonable time frame.

4.1 Bottom-mounted rectangular block

The first test case is the bottom-mounted rectangular block as tested by Blendermann (2013). A scan of the reference data is shown in Fig. 38. The data is digitised and used as validation material.

Some uncertainty is associated with this experimental data:

- the digitisation process can add errors, e.g. due to (partially) overlapping symbols,
- no information is available in Blendermann (2013) related to the blockage in the wind tunnel where the experiments were conducted,
- the upstream length of the bottom plate (and the thickness of the boundary layer that develops on it),
- the type of wind tunnel used (open jet or closed).

Therefore, it is assumed that blockage corrections were not applied nor required.

4.1.1 Computational setup

For $L/B = 1$ and 4, 10 computations were configured. The dimensions of the rectangular block (for $L/B = 6.67$) are given in the caption of Fig. 5.30 on p. 69 of Blendermann (2013): 0.15 m, 0.15 m and 1.0 m. Due to the use of a unit domain size in Gerris, the geometry is scaled down by a factor three (as was done in the previous chapter): 0.05 m, 0.05 m and 0.333 m. Flow in the wake is refined within a cylindrical surface with radius 3.5 times the cube length, to two levels lower than the refinement on the cube itself (N_s = 11; N_w = 9)[23]. A uniform velocity is set at the inlet with unit magnitude. Computations are executed for 10 (dimensionless) seconds and averages are computed from the time traces starting at $T = 4$. The angle of the rectangular block is varied between $0°$ and $90°$ in steps of $10°$[24]. The domain consists of two cubic boxes placed behind each other, with the rectangular block located at the centre of the frontal one.

In the previous chapter, amongst others, the influence of the maximum refinement levels on the resulting forces and moments was investigated for the wall mounted cube at $\varphi = 0°$. For the rectangular prism with $L/B = 4$, the values of the refinement level will be modified to determine the influence on the comparison with experimental data for the full range of relative wind angles. In addition to the influence on the coefficient values as compared to the experimental reference values, the computational times will be compared as well. The extra tests that will be executed for $L/B = 4$ are summarized in Table 7.

[23] This proved to give acceptable computing times for the $L/B = 1$ case, but increased computational times too much for the $L/B = 4$ case.

[24] Since the rectangular prisms used here have at least two symmetry axes (as opposed to one symmetry axis for ships), one quadrant of circle is sufficient to compute coefficients.

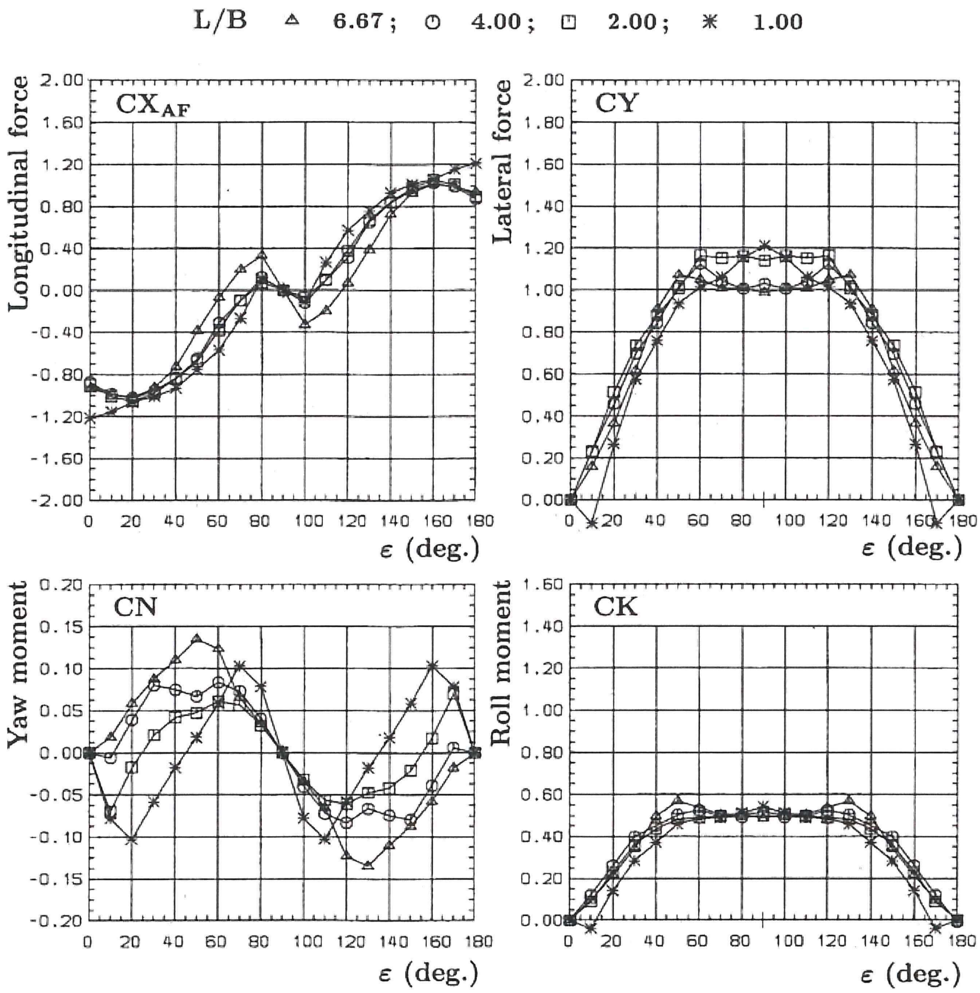


Figure 38 – Force and moment coefficients of a bottom-mounted rectangular block (height/breadth = 1.00) for various values of length-breadth ratio.

Table 7 – Wake refinement parameter variation for $L/B = 4$

Computation	N_s	N_w
default	11	9
c1	10	8
c2	9	7
c3	9	8
c4	10	9

4.1.2 Results for $L/B = 1$

Time histories of the four coefficients (and their cumulative moving averages) for $\varphi = 10°$ and $40°$ are shown in Figs. 39 and 40, respectively. For the other eight angles, the graphs are shown in Appendix A3, in Figs. 68 to 75. Note that the time signal has been made dimensional again: due to the 1/3 scale of the cube compared to the experiment, the dimensional end time T' equals 30 seconds. The convergence of the four signals is not always equally fast: C_N reaches a steady state significantly faster than C_Y and C_K (see e.g. Fig. 40). For this case ($\varphi = 40°$), the latter coefficients need at least three dimensionless seconds to visually reach a steady state. For the cases where the cube is (nearly) aligned with the flow (0, 10, 80 and 90, convergence is significantly faster (one second). All cases reach a steady mean state, although some (such as $\varphi = 30°$) could be run for a longer time to reduce the impact of low-frequency oscillations on the final average value. These averages, together with the standard deviations computed over the same interval, are shown in Fig. 41 as a function of the wind angle φ, where the digitised experimental values for $L/B = 1$ (shown before in Fig. 38) were added for comparison.

Despite the lack of physical viscosity in the computations (and the associated viscous interaction between the flow and the solid objects in it), the resultant forces and moments predict the same trends as found in the experiment, although some of the peak values are over- or underpredicted. The match with the forces ($C_{X_{AF}}$ and C_Y) is especially good, the biggest difference occurs at $\varphi = 80°$ for $C_{X_{AF}}$ (which corresponds to C_Y = $10°$ for C_Y). The numerical results assume a perfectly uniform velocity field that is not disturbed by the presence of the ground plane. This effect is visible in the force results, where the on-axis force predicts slightly larger values than the experimental results[25]. This is also visible in the curve of the roll moment C_K, that shows the same trend as the curve of the lateral force C_Y with a factor two difference in magnitude. The roll moment lever arm (C_K/C_Y) (relative to the total height of the cube), is shown in Fig. 42. For CFD, its average value is located at the lateral plane centroid (0.5). For the experiments, it is located somewhat lower.

For the yawing moment C_N, the trends shown are similar: both the numerical and experimental results predict the maximum value of the yawing moment at $20°$ (and $70°$). The magnitude of this value is lower for the numerical result by a significant margin (40% relative error).

Snapshots of the flow around the cube at $T = 10$ are shown in Fig. 43 for $\varphi = 0°$ to $50°$: the velocity magnitude in a plane parallel to the bottom, halfway between the bottom and top of the cube. For the other wind angles, the visualisations are shown in Fig. 76. For this wall-mounted cube, the angles from $50°$ to $90°$ should not have been executed for reasons of symmetry: the condition for $\varphi = 0°$ is the same as the case for $\varphi = 90°$. This observation is also confirmed with the trend of the yawing moment C_N in Fig. 41 which shows a point-symmetry with respect to $\varphi = 45°$.

4.1.3 Results for $L/B = 4$

To get an idea of the influence of the maximum refinement level at different wind angles, additional computations are executed where the refinement levels on the rectangular prism and in the wake are altered for the rectangular cuboid with $L/B = 4$.

Averages and standard deviations are computed starting at t = 4. These are shown in Fig. 44 together with the reference data from Blendermann (2013) for this case. Relative errors of the numerical results are displayed in Fig. 45, these are computed using

$$E(v_a) = 100\% \times \left|\frac{v_r - v_a}{v_r}\right| \tag{46}$$

where v_a is the approximation and v_r the reference value.

[25] Viscous interaction with the bottom reduces the pressure on the bottom part of the object facing the flow, with a horseshoe vortex wrapping around the base that reduces the pressure near the base of the object slightly.

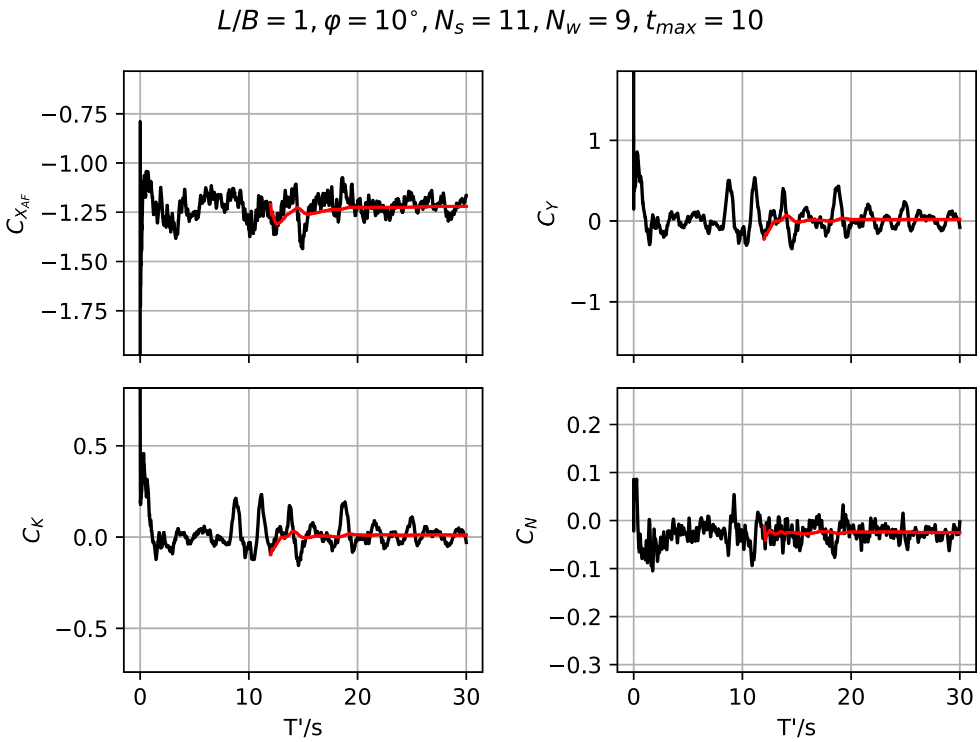


Figure 39 – Time histories and cumulative moving averages of the four coefficients for the wall-mounted rectangular block $(L/B = 1)$ for $\varphi = 10°$.

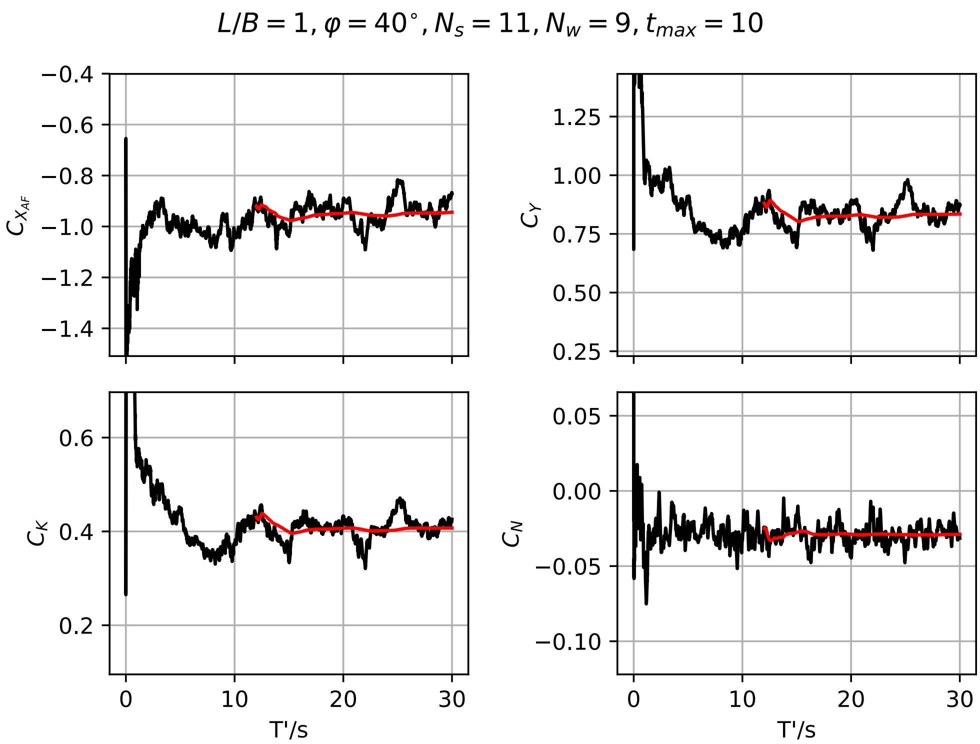


Figure 40 – Time histories and cumulative moving averages of the four coefficients for the wall-mounted rectangular block $(L/B = 1)$ for $\varphi = 40°$.

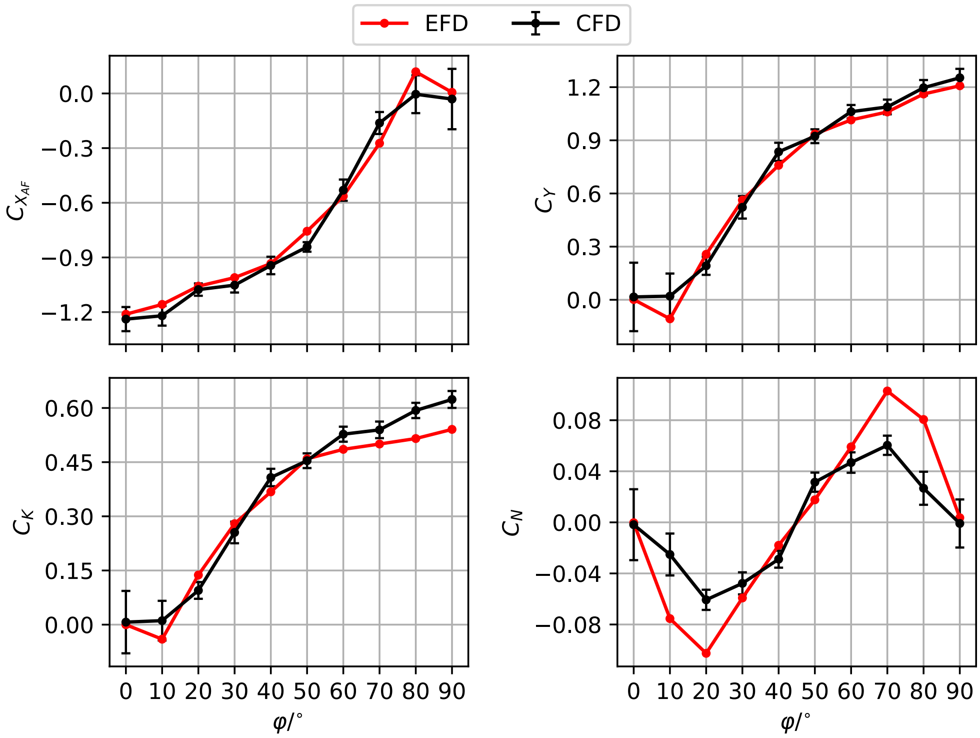


Figure 41 – Comparison of numerical (CFD) and experimental (EFD) data of the wall-mounted rectangular block ($L/B = 1$) as a function of the wind angle φ.

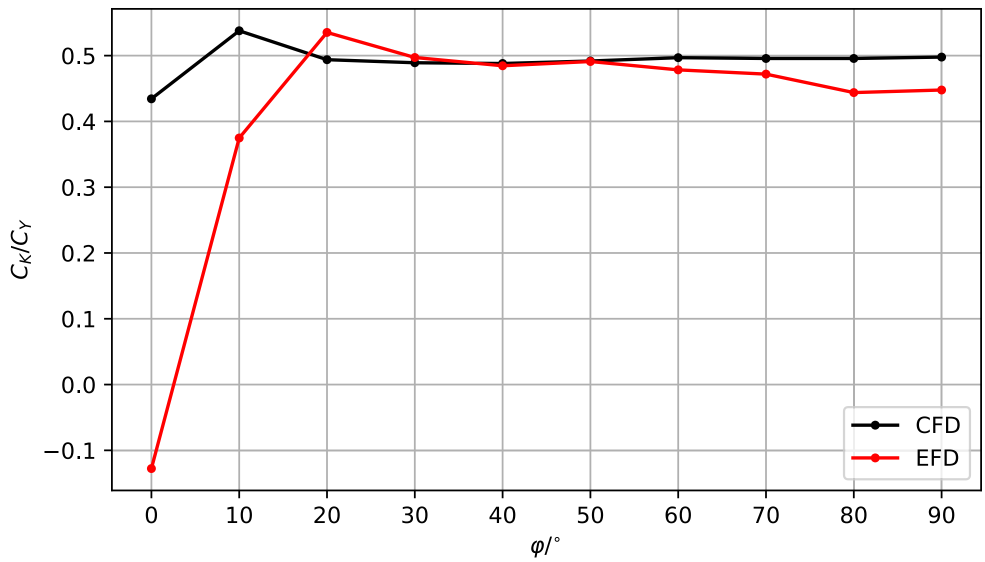
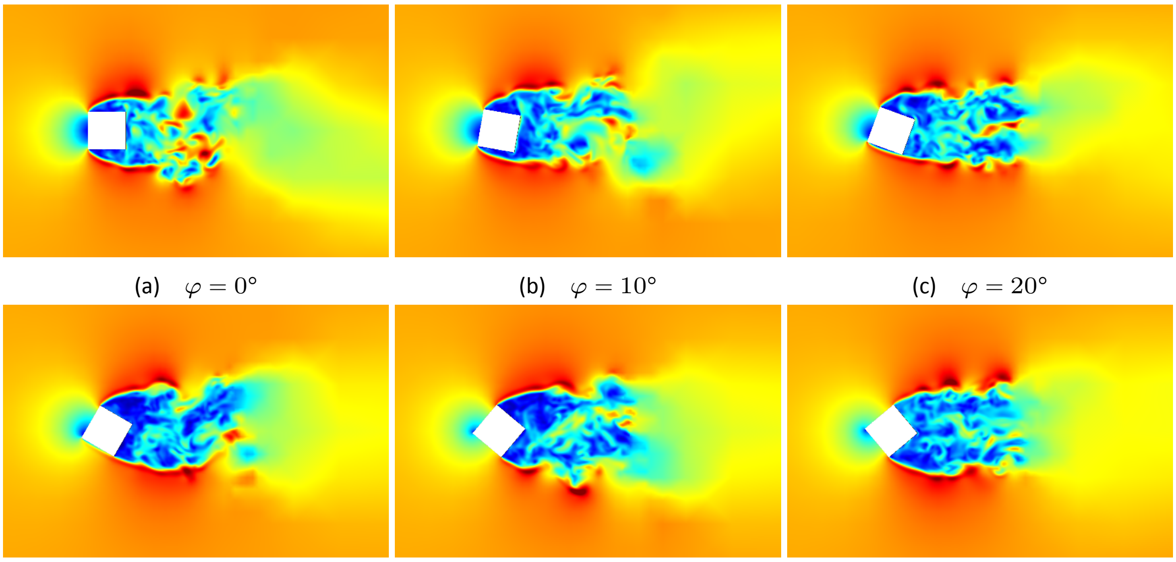


Figure 42 – Vertical application point of the lateral force relative to the height of the wall-mounted rectangular box ($L/B = 1$).

The results with $N_s = 9$ (both with $N_w = 8$ and $N_w = 7$) show very similar trends, where the one with the higher value in the wake shows an almost constant offset over the complete range of wind angles for C_Y and C_K. These two show the biggest difference with the experimental results for small values of φ for $C_{X_{AF}}$, the computations with higher levels of refinement on the body agree well with the experimental data in this area (the relative error is less than 10%), as was the case for $L/B = 1$ (recall Fig. 41).

Another observation regarding the $N_s = 9$ results is that for the lateral force C_Y and the roll moment C_K, the



(a) $\varphi = 0°$ (b) $\varphi = 10°$ (c) $\varphi = 20°$

(d) $\varphi = 30°$ (e) $\varphi = 40°$ (f) $\varphi = 50°$

Figure 43 – Snapshot of velocity magnitude in a horizontal plane halfway between the bottom and top of the cube.

trends are good for small wind angles, but completely break down past $\varphi = 40°$, where the predicted value is significantly lower than the experimental reference. Only for the $\varphi > 70°$ do the trends catch up with the experimental reference again.

The computation using the settings optimised in the previous chapter follows the trends of the experiment quite well, but overall, it does not give the best agreement with the experiments: slightly better results are obtained when the refinement levels are reduced by one (both for the wake and the object). Here again, a similar observation can be made between the two results with $N_s = 10$: both curves show very similar trends, with an offset that is almost constant over the complete range of wind angles. The results with $N_s = 11$ and 10 show an almost linear increase of C_K until $30°$ and a maximum at $60°$ while for higher values of the wind angle, the roll moment is almost constant, similar to the experimental results.

It is clear that although Gerris can predict the general trends of the experiments, there are still differences. Not a single set of computational settings gives the best overall results compared to the experiments. Therefore, the required computing time should be included in the comparison as well: there is no point running computations for a week if a computation run on a coarser grid that finishes in a single day gives similar results. The computing times (normalised to 8 CPUs) for the different computations are shown in Fig. 46. The differences or improvements one gets when executing computations with the highest settings for the grid refinement around the solid object and in its wake do not outweigh the required time (which is more than four days on average). By reducing the refinement level on the object by one, the average time is reduced to less than two days, while an additional reduction of the refinement in the wake reduces the required time to less than half a day.

4.2 Wind coefficients for configuration 01-01-02 of Andersens' VLCC

4.2.1 Introduction

In the second test case, the experience gained during this project is used to compute wind coefficients for the VLCC model of Andersen (2007) and Andersen (2013). An analysis of this source of validation data is given in Appendix A6. Reference data will only be used for the fully stacked condition (code `01-01-02`) for which

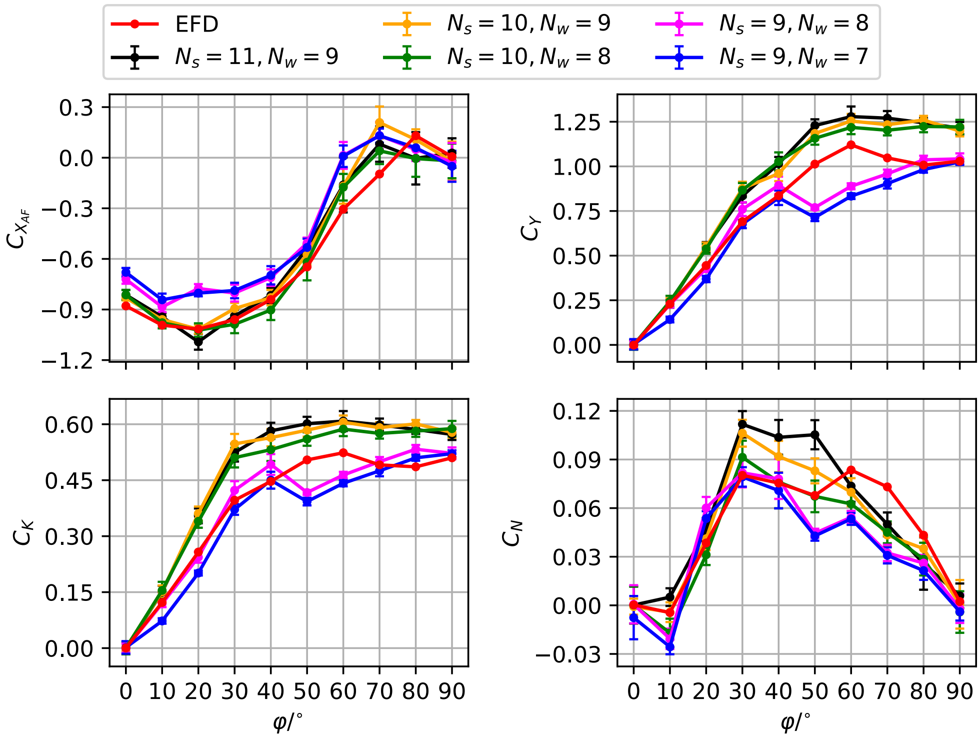


Figure 44 – Wind coefficients for the wall mounted rectangular prism ($L/B = 4$) as a function of the wind angle: influence of refinement settings on the solution.

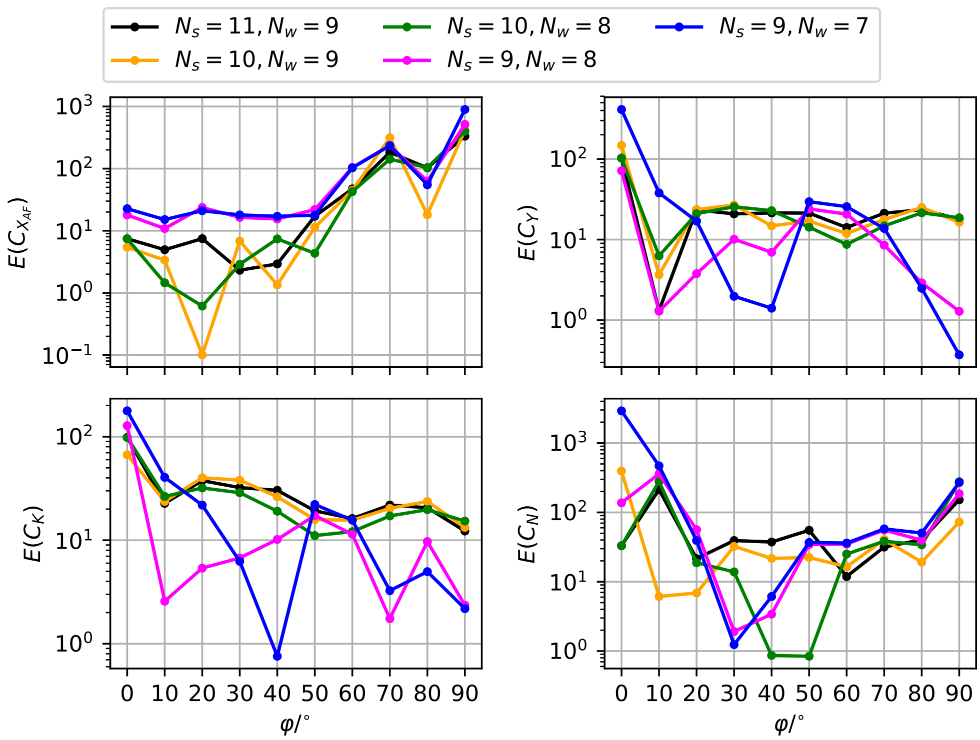


Figure 45 – Relative errors of the results presented in Fig. 44.

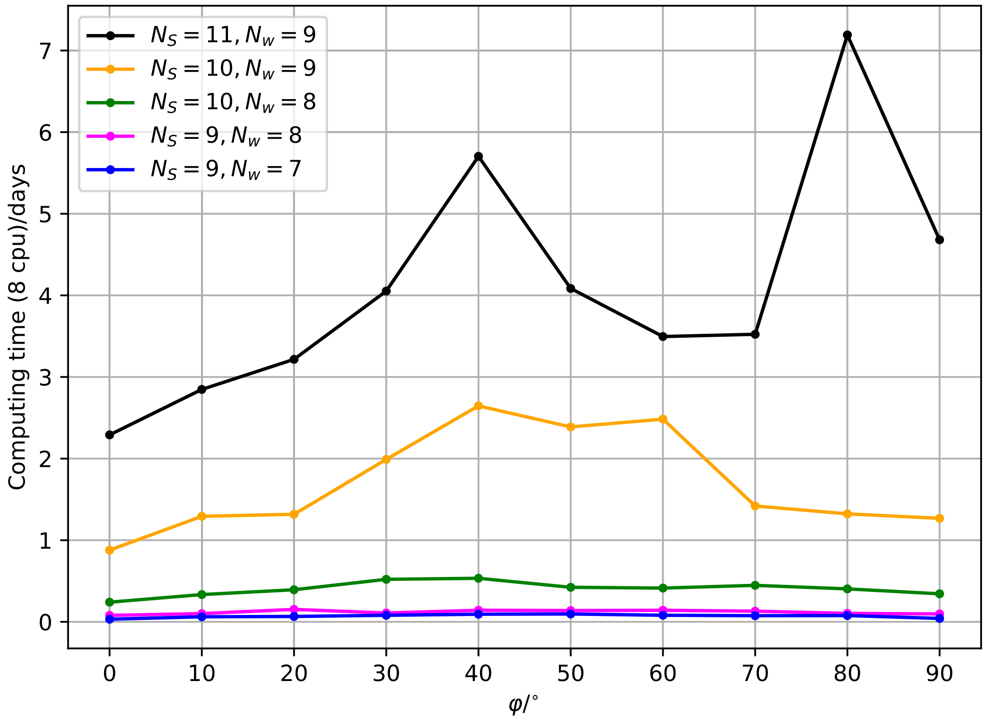


Figure 46 – Computing times of the results presented in Fig. 44.

unprocessed[26] data is available (Andersen, 2007, p. 234 (Bilag O1)). The unprocessed data for case 01-01-02 is shown in Fig. 47, including the roll moment C_K. Due to issues with the measurement equipment, roll moment values were not included in the post-processing by Andersen, only the longitudinal and lateral force components and the yawing moment are available for all tested configurations. For the current validation, the roll moment will not be included.

The biggest difference with the previous cases of Blendermann is that for this specific experimental case, the natural boundary layer of the wind tunnel was used to subject the ship model to a non-uniform velocity profile. This means that a post-processing step is required where the CFD results (obtained in a uniform flow field) are adapted to account for the vertical velocity profile. The correction functions of Blendermann (2013) will be used for this purpose. Alternatively, these corrections can also be used to convert the experimental results to a uniform flow field.

[26]With *unprocessed data* is meant the data before corrections for both blockage and reference pressure are applied.

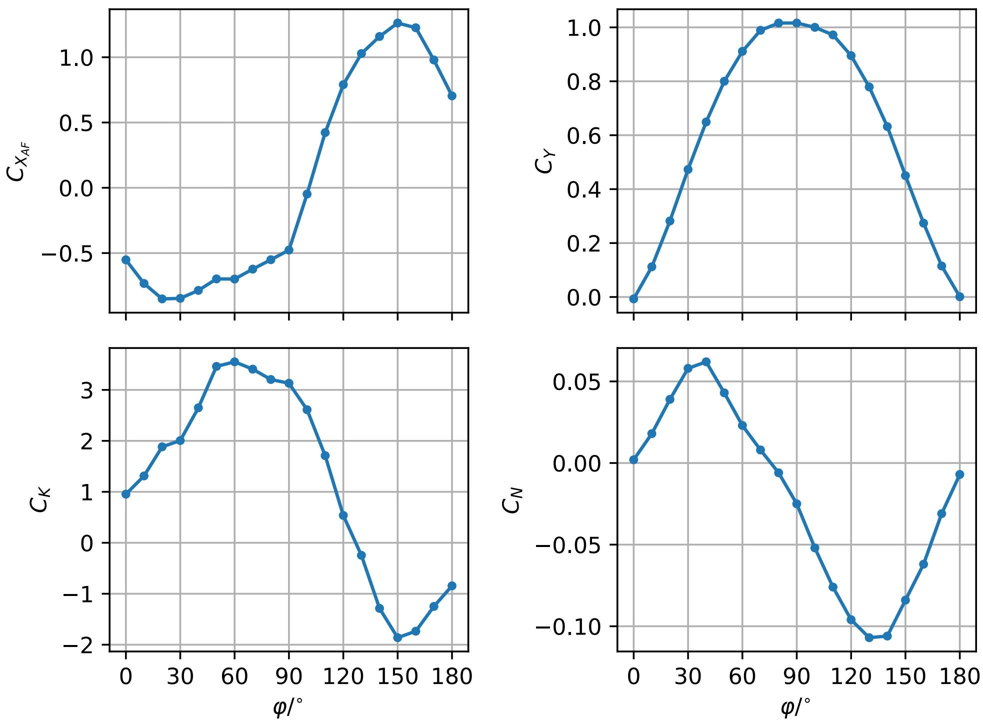


Figure 47 – Unprocessed coefficient values from Andersen (2007) for configuration 01−01−02.

4.2.2 Velocity profile and dynamic pressure

The velocity profile as measured at the forward testing point (where the ship model was mounted in the wind tunnel) is shown in Fig. 48 together with a scaled silhouette of the frontal projection of the ship[27]. For comparison, a logarithmic profile that has the same velocity as the measured velocity profile at the scaled reference height (horizontal black line) of 10 m is added in addition to the profile (in red) assumed by Andersen, the 1/7 power profile (in green) as assumed by Blendermann for wind tunnel boundary layers, and a 1/5 power profile (in orange) that gives a better fit of the boundary layer profile below $z = 0.35$ m. One can deduce from this figure that the upper 60% of the ship model was subject to a uniform flow, while for the lower 40% a gradient is present in the velocity field. Also, the velocity field assumed by Andersen gives a reasonable match to the complete measured vertical velocity profile, although for the boundary layer (roughly below $z = 0.4$ m) other profiles show a better agreement.

It should be noted that the coefficient values shown in Fig. 47 were converted to non-dimensional form by dividing them by the dynamic pressure as measured in the wind tunnel[28], not the dynamic pressure based on the velocity at the model reference height ($= 10$ m$/450 \approx 0.022$ m). This means that for the CFD computations (where uniform velocity is assumed), the resulting coefficients (especially for the lateral force at $90°$) will likely be larger than the ones shown in Fig. 47. Integration of the dynamic pressure over the height of the ship results in a force of 5.45 N/m while the assumption of uniform velocity over the ship height with a magnitude of 9.94 m/s results in a force of 6.08 N/m. It is estimated that as a result of this difference, the (lateral) force coefficients computed from CFD will be approximately $10\,\%$ higher than the coefficient values obtained by Andersen (2007).

Following Blendermann (2013), correction factors are computed to convert the coefficients of Andersen (2007) to a uniform flow field. At model scale, the mean height H of the lateral plane equals $H = A_L/L_{oa} = 0.066\,14$ m. This is higher than the thickness of the boundary layer ($\delta \approx 0.04$ m), which means that the ref-

[27]The height of the ship model tested in the wind tunnel is approximately 0.10 m.
[28]And obviously, the appropriate reference area.

erence dynamic pressure q_{ref} for the longitudinal force requires no correction, since it uses q_H as reference height. For the lateral force, the mean dynamic pressure over the mean height is found by numerically integrating the dynamic pressure from $z = 0$ to $z = H$ and dividing that result by H which gives $\bar{q}_H = 50.15$ Pa. The dynamic pressure at the mean height equals $q_H = 59.03$ Pa. Then, the reference dynamic pressure for the lateral force is computed,

$$q_{ref} = k_q \bar{q}_H + (1 - k_q) q_H = 53.70 \text{ Pa.} \tag{47}$$

The correction factor for the lateral force coefficients is found from

$$q_{corr} = \frac{q_0}{q_{ref}} == \frac{\frac{1}{2}\rho U_{pitot}^2}{q_{ref}} = 1.107, \tag{48}$$

which is a correction (increase) of approximately $11\,\%$, almost the same as the difference in the integral of the dynamic pressure over the ship height as mentioned before. The value for q_0 is the (reference) dynamic pressure as obtained by Andersen(Andersen, 2007) in the wind tunnel (with $\rho = 1.204$ kg/m^3 and $U_{pitot} = 9.94$ m/s).

4.2.3 Computational setup

The CAD model used in section 3.3.1 is scaled by a factor $2.266\,667$ (hence $13\,\%$ larger than the largest size that was used in the blockage investigation) such that its length corresponds to the length of the model as used in the wind tunnel (≈0.75 m). The cross section of the domain is a square with side one, while the wind tunnel cross section used by Andersen is 1 m wide and 0.7 m high. The blockage is thus lower in the CFD computations than in the experiments. It should be noted that the chamfers in the four corners of the cross section of the wind tunnel have been left out in the CFD computations, similar to Janssen *et al.* (2017).

Due to the size of the vessel compared to the domain, it would be located too close to the inlet. For this reason, the domain is enlarged by adding two boxes aft of the two default ones and the ship is moved to the centre of the second box. For three wind angles ($\varphi = 0°,\ 40°$ and $90°$), a top view of the domain with the ship model is shown in Fig. 50 where the velocity magnitude is displayed on the bottom plane.

For the computations, the inlet velocity is set to 1 m/s, while for the experiments, it was 45 m/s. Since the geometric scaling factor equals one (the CAD model is as large as the experimental model) and the velocity scale factor is 45, one second of computational time corresponds to $\frac{1}{45} = 0.0222$ s dimensional time. By setting the end time of the computations to 20, the physical time period equals 0.444 s. For the reference frontal and lateral areas, scaled values from Table 5 are used.

Computations are configured for 19 angles from $0°$ to $180°$ where $N_s = 8$ and $N_w = 7$ and the wake refinement is restricted to a cylindrical volume centered at the ship with a radius slightly larger than the length of the vessel.

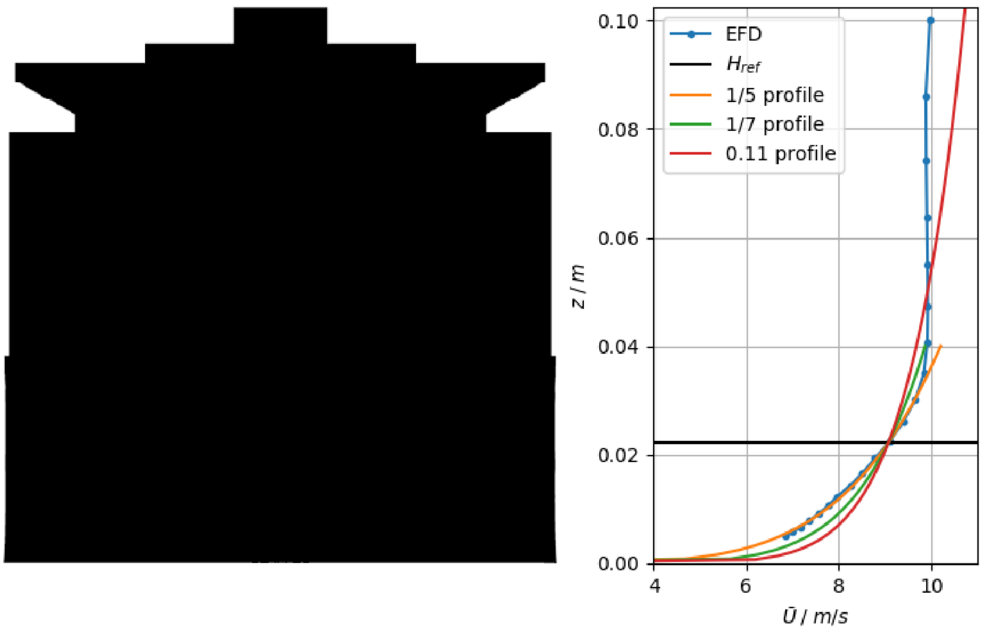


Figure 48 – Model ship silhouette and velocity profile (to scale) with power law fits.

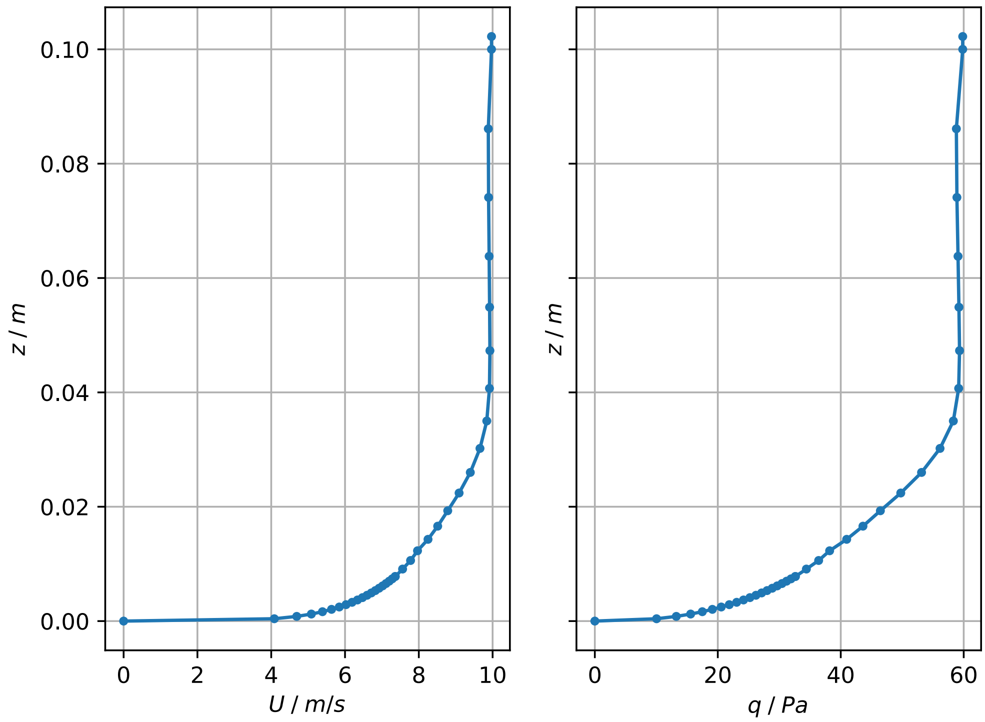
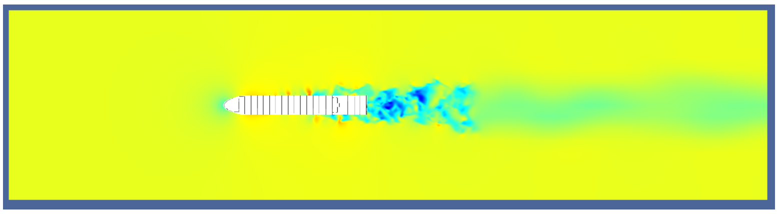
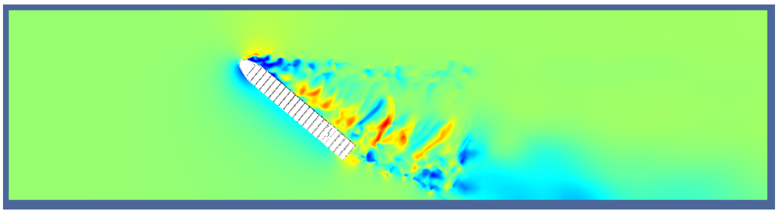


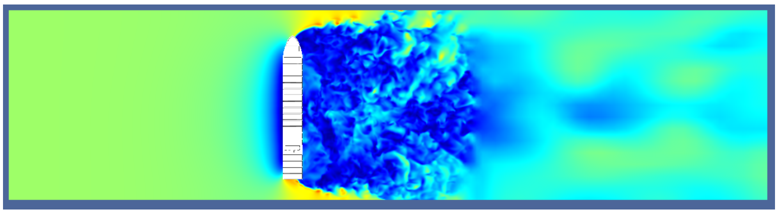
Figure 49 – Wind tunnel velocity and dynamic pressure profiles.



(a) $\varphi = 0°$.



(b) $\varphi = 40°$.



(c) $\varphi = 90°$.

Figure 50 – Top view of the computational domain showing the location and size of the ship model with respect to the domain size and the magnitude of the velocity on the bottom plane.

4.2.4 Results

The four coefficients $C_{X_{AF}}$, C_Y, C_K and C_N are shown in Fig. 51 as a function of the wind angle φ. The experimental data from Andersen reduced to uniform flow is shown as well. The relative errors of the numerical values are displayed in Fig. 52. Both force components follow the experimental reference curves well. Not much can be said of the roll moment, as the reference data clearly has issues (as was noted by Andersen (2007)). The agreement for the yawing moment is not as good, although the overall shape does show correct trends. Due to the significantly lower maximum value of the first extremum, the zero-crossing occurs earlier around $45°$ while for the experiment, it is located near $75°$. From $120°$ onwards, both the trend and the magnitude of the values show reasonable agreement.

Given that the forces do show a good agreement with the experimental data, it is possible that the longitudinal position of the reference point is located incorrectly. A careful inspection of the lateral view of the ship and the CAD model (see Fig. 53) revealed that the reference point of the CAD model is indeed too far forward (by $\approx 2\,\%$).

Due to the proximity of the model to the domain sides, it was decided to adapt the geometry and rerun the computations with the same settings as used for the results shown before in Fig. 51. In addition, wind coefficients are computed for the simplified geometry of the VLCC without gaps between the container stacks as discussed in § 2.3 (also with corrected reference point location). Both the original and new results are shown in Fig. 54, while the relative errors are displayed in Fig. 55. It should be noted that in these figures, the reference values for the roll moment have been left out. The graph with label *CFD conf. A* is the original geometry, the graph with label *CFD conf. B* is the result for the geometry with the corrected reference point, and *CFD conf. C* is the simplified geometry with corrected reference point. As was mentioned at the end of § 3.2, the lateral force coefficient is indeed significantly higher for configuration C: the uninterrupted lateral surface area results in a disproportionally larger pressure force as compared to the result for the lateral surface area with gaps between container stacks. The reference point location has almost no influence on the forces acting on the ship, only the yawing moment shows a significant difference between configuration A and B (as expected).

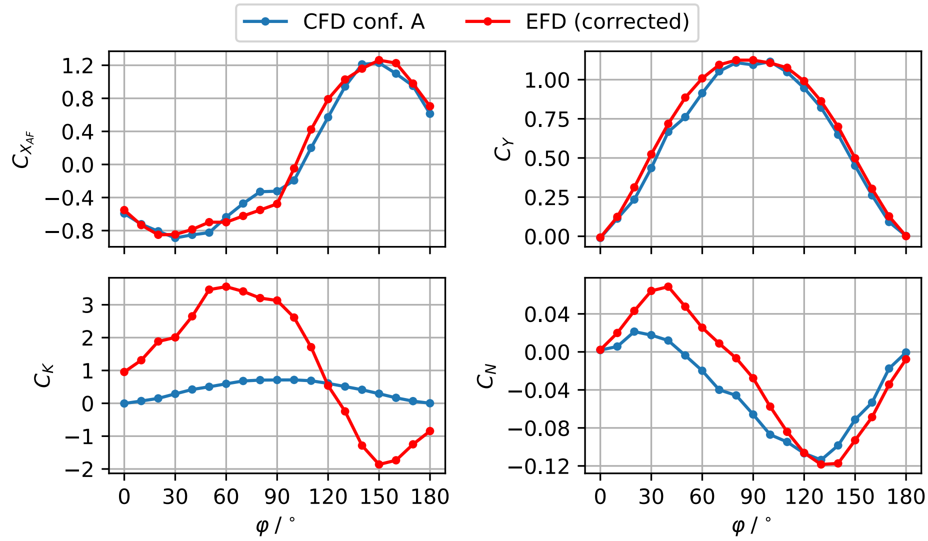


Figure 51 – Comparison of CFD results with EFD data for configuration 01−01−02 of Andersen (2007). The EFD data is reduced to uniform flow following the procedure described in Blendermann (2013).

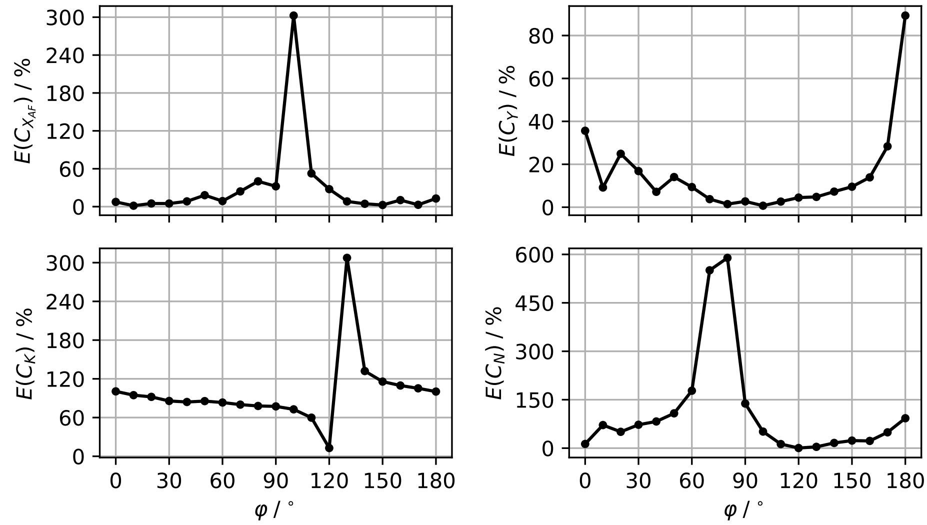


Figure 52 – Relative errors of the results shown in 51.

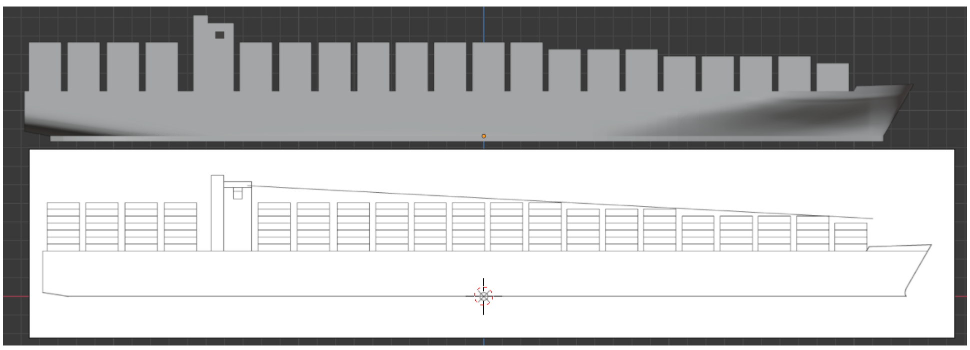


Figure 53 – Longitudinal offset of the ship geometry between the reference drawing (bottom) from Andersen (2007) and the initial ship geometry (top). In these views, the reference points are aligned.

For configuration C, the yawing moment has significantly increased in value and its extrema surpass the experimental extrema. Removing the gaps between the container stacks has a profound influence on the longitudinal force, which now shows a trend that contains a component proportional to $\sin(4\varphi)$ while with separate container stacks, the highest frequency component is proportional to $\sin(2\varphi)$. This can also be observed in the results of the simplified configuration tested by Janssen *et al.* (2017). Another observation that can be made from these results is related to the peak in the value at $\varphi = 50°$ for C_Y (and C_K) for configuration C that is absent for both the other configurations and the reference. For this, first the lift and drag coefficient of the hull are computed from $C_{X_{AF}}$ and C_Y:

$$C_L = C_{X_{AF}} \frac{A_F}{A_L} \sin\varphi + C_Y \cos\varphi, \tag{49}$$

$$C_D = -C_{X_{AF}} \frac{A_F}{A_L} \cos\varphi + C_Y \sin\varphi, \tag{50}$$

where it should be noted that in order to combine $C_{X_{AF}}$ and C_Y in one equation, the same reference areas must be used: the former one is scaled such that both can be added together. Here also, the experimental data is corrected for the deficiency in the dynamic pressure for the lateral force. The results are shown in Fig. 56. The extrema of the lift (i.e. the force component perpendicular to the oncoming flow) are almost twice as high for configuration C as compared to the extrema of the experimental data and configurations A and B: by removing the gaps between the container stacks, the pressure fields on the sides of the hull can reach more extreme values, while with gaps, the pressure fields between both sides are equalised at each gap. The gaps between the container stacks result in slightly higher drag at wind angles near $\varphi = 0°$, while for angles near $\varphi = 90°$, the drag is increased considerably. Overall, the agreement of the CFD results with the experimental results is best for the first two configurations (with gaps between container stacks, corresponding to the scale model tested by Andersen).

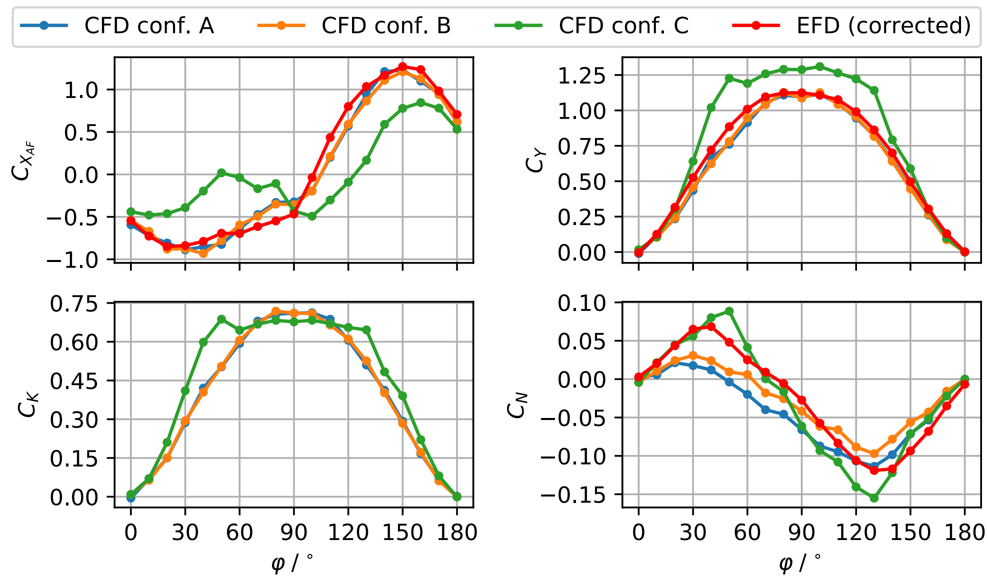


Figure 54 – Comparison of predicted wind coefficients for three alternative geometries for configuration 01–01–02 of Andersen (2007). *CFD conf. A* is the hull geometry with gaps between container stacks and the incorrect location of the reference point, *CFD conf. B* has the container stack gaps and the correct location for the reference point and *CFD conf. C* has the correct reference point location but lacks the container stack gaps.

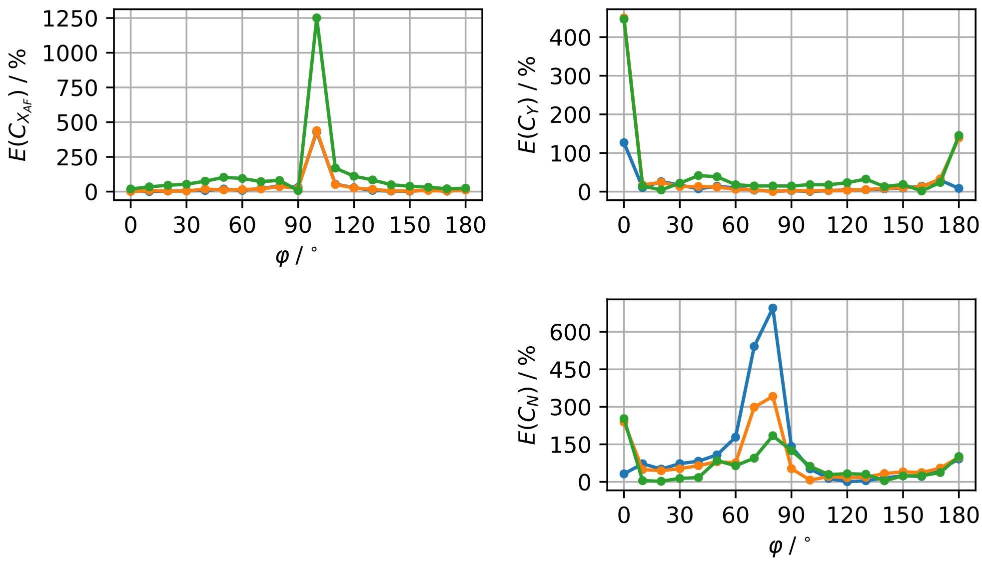


Figure 55 – Relative errors of the coefficient values shown in Fig. 54. Relative errors for C_K are not computed.

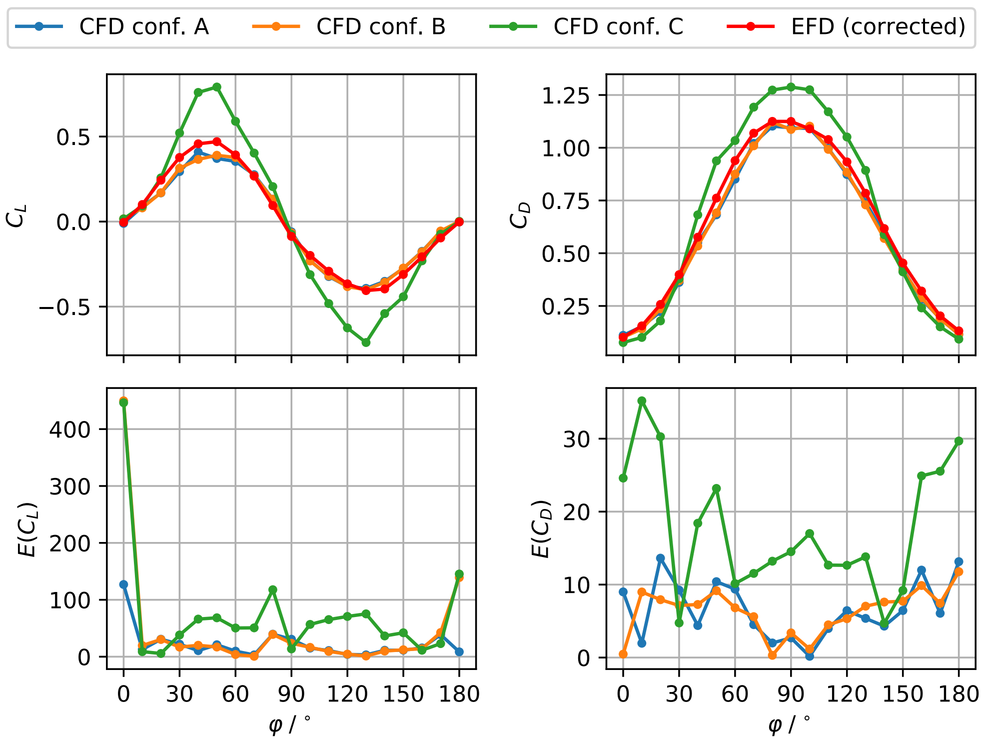


Figure 56 – Lift and drag coefficient values (and relative errors) of the hull of the VLCC of Andersen (2013) for 01−01−02 as a function of wind angle.

4.3 Results for additional configurations of Andersen's VLCC

Due to the encouraging agreement for the forces on the hull of the container ship tested by Andersen in combination with the relatively short computing times to obtain a complete wind coefficient set, additional computations are executed for this experimental data set with different container configurations[29]. The configuration number, the lateral and frontal reference areas and a side view of the ship are gathered in Table 8. For the configurations with containers on deck, the full height of the stacks is used, similar to the initial configuration.

Computational settings are exactly the same as in the previous section, with the only difference that the geometry file was changed. All 95 computations ran on four processors of either the Navier or Stokes queue, and all of them finished within 24 hours after they were launched[30].

It is expected that for configuration 01-01-01, the drag will be smaller than measured in the experiment. Since for this case, the majority of the hull is a (relatively) smooth slender object, it is expected that the percentage-wise contribution of friction to the drag (or longitudinal force) is higher than for example for configuration 01-01-02.

Table 8 – Details of additional configurations of the VLCC of Andersen.

configuration	A_L/m^2	A_F/m^2	side view
01-01-01	$0.030\,426\,65$	$0.008\,457\,92$	
01-02-01	$0.040\,755\,3$	$0.009\,477\,6$	
01-02-03	$0.047\,344$	$0.009\,477\,6$	
01-02-07	$0.041\,374\,2$	$0.009\,477\,6$	
01-03-01	$0.034\,938$	$0.009\,477\,59$	
01-03-02	$0.046\,219\,6$	$0.009\,477\,6$	

Results are shown in Figs. 57 to 62. Trends for the force coefficients are in general in good agreement with the experimental values, although for two cases (01−01−01 and 01−03−01), the shape of the lateral force curve C_Y near $\varphi = 90°$ shows a sharp peak that is not present in the experimental data. The exact reason for this is not known, but these two cases do correspond to the two configurations tested that where a large part of the ship hull has no containers on deck. Interestingly, for these two cases, the roll moment C_K does not show the sharp peak near at $\varphi = 90°$. As was predicted before the computations were executed, for configuration 01−01−01 (Fig. 57), the CFD prediction of the drag force $C_{X_{AF}}$ is consistently lower in magnitude than the experimental case. As was the case with the rectangular cuboid, the yawing moment shows the correct overall trends, but the prediction of the first peak magnitude (between $0°$ and $90°$) is in some cases (such as 01−02−01 and 01−02−07, respectively Fig. 57 and 60) significantly underpredicted.

[29] Another argument is to verify that the validation presented in the previous section is not just a result of getting *lucky*.

[30] The Navier and Stokes queues do not have sufficient cores available to run all 95 computations at the same time.

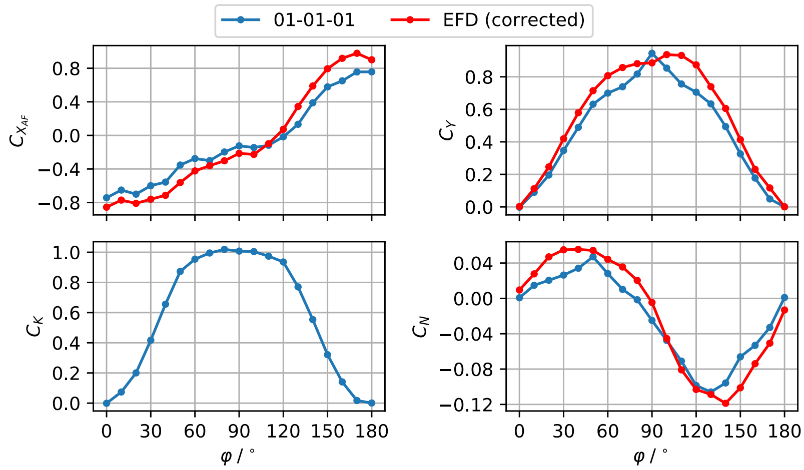


Figure 57 – Configuration 01−01−01 of Andersen (2007): comparison of numerical and experimental wind coefficients.

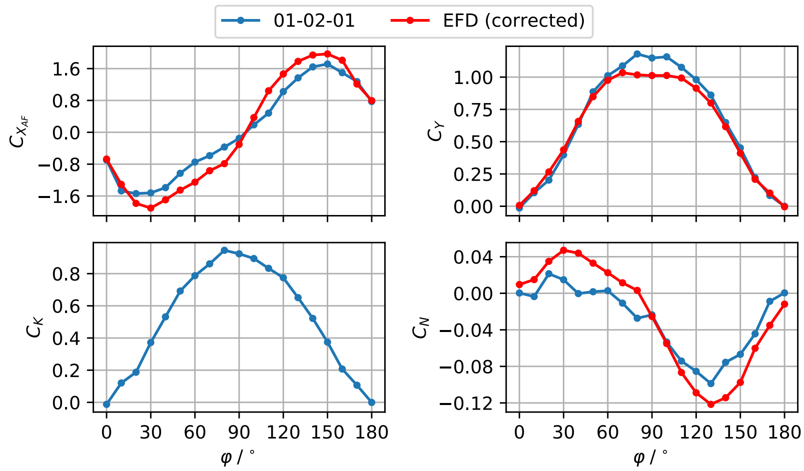


Figure 58 – Configuration 01−02−01 of Andersen (2007): Comparison of numerical and experimental wind coefficients.

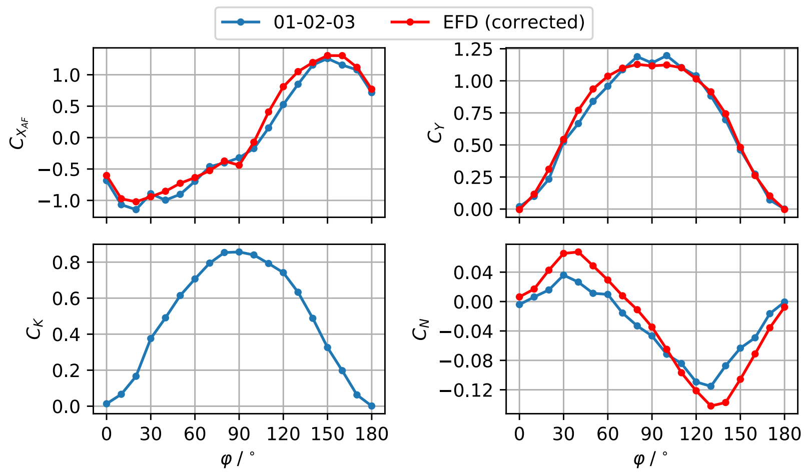


Figure 59 – Configuration 01−02−03 of Andersen (2007): comparison of numerical and experimental wind coefficients.

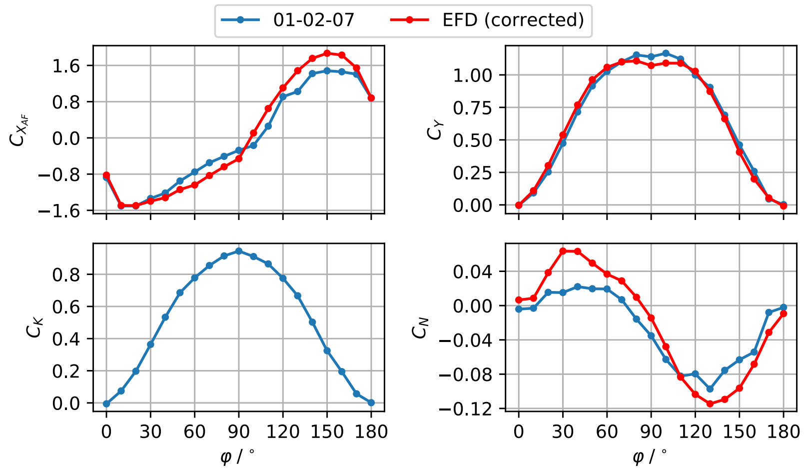


Figure 60 – Configuration 01−02−07 of Andersen (2007): comparison of numerical and experimental wind coefficients.

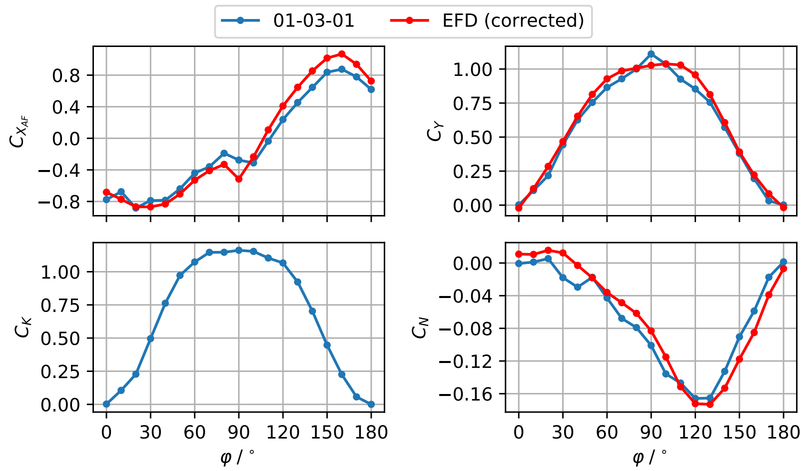


Figure 61 – Configuration 01−03−01 of Andersen (2007): comparison of numerical and experimental wind coefficients.

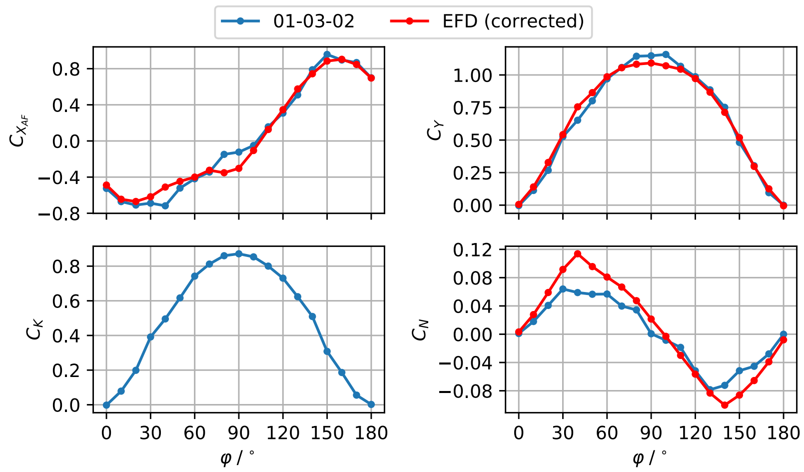


Figure 62 – Configuration 01−03−02 of Andersen (2007): comparison of numerical and experimental wind coefficients.

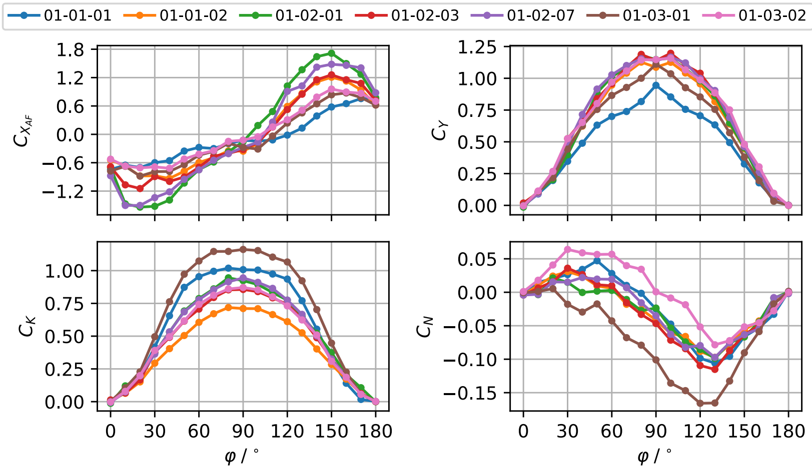


Figure 63 – Wind coefficients for all configurations listed in Table. 8.

5 Conclusions and Outlook

5.1 Parameter variations and validation

The Gerris Flow Sover has been evaluated for the computation of forces on large bluff bodies such as container ships subject to a uniform wind field. The computational setup of the flow solver is explained, experimental reference cases used for the validation are introduced and boundary and initial conditions are detailed. One of the key features of the solver is its ability to adapt the grid to the flow in an automatic manner. A downside to this is that a large wake refinement region combined with a high refinement level in said region can slow down computations considerably because the grid count can quickly become significantly larger than initially anticipated. This means that one should be careful when configuring computations or extrapolating computing time estimates based on a single computation. Before performing a validation against experimental data from literature, a number of parameters have been altered to determine their influence on the results. These parameters are the blockage, the maximum refinement in the wake and the length of the wake refinement region. The parallel efficiency and scalability of Gerris on the cluster at FHR has been investigated as well.

The first validation case concerned a wall-mounted rectangular prism with various length to width ratios tested in uniform flow by Blendermann (2013). The second validation case was a container ship tested in the natural boundary layer of a closed-circuit wind tunnel by Andersen (2013). A review of the experimental setup for this second case showed that the upper $60\,\%$ of the hull experiences uniform flow. To facilitate a comparison with the numerical results obtained with Gerris in a uniform flow field, the experimental values are corrected with the empirical relationships determined by Blendermann (2013). In both cases (the wall mounted cube and the VLCC model), good agreement with experimental data is found for the drag and lateral force as a function of the wind angle. The prediction of the yawing moment shows larger differences: the trends are captured, but the peak values are in some cases significantly underpredicted. The exact cause of this is not known.

When performing a validation against experimental data (or when computing coefficients for use in the simulator), care should be exercised when recreating the geometry used in the experiments: both the level of detail included in the geometry (such as the gaps between container stacks) and the location of the reference point have been shown to affect the results. The gaps between container stacks have a significant impact on the predicted forces and moments: without gaps, the predicted lateral forces as a function of wind angle is overpredicted by a significant margin. The shape of the longitudinal force component deviates a lot from the reference as well: without container stack gaps, the C_X curve is proportional to sin(4φ), while with gaps, it is proportional to sin(2φ), which confirms similar findings in literature. This does mean that computations executed in the past with FINE/Marine for the estuary vessel *Tripoli* (Van Hoydonck *et al.*, 2015b) (where the gaps between container stacks were neglected), may have to be revisited.

It is concluded that Gerris can be used to compute wind coefficients of ships in an efficient manner and that the results can be used as such in the mathematical models used in the ship simulators of FHR.

5.2 Gerris

At the moment, Gerris is unmaintained (bugs are not fixed, no new releases) by its developers because the focus of development has shifted towards improving its successor Basilisk[31]. This means that the issue regarding

[31] *Basilisk* is the name of Basiliscus basiliscus, which is also known as the Jesus-Christ lizard while *Gerris lacustris* is the Latin name of the common water strider, two animals that can walk on water.

the bad performance of the parallelisation (with computations running on more than 8 processors) that has been observed during this research will not be solved.

The development focus of Basilisk has in part been steered by weak points in maintaining and using Gerris: the scripting language used for the configuration of computations adds a significant amount of code that must be maintained. In Basilisk, actual C programs must be written that directly use the functions defined in the Basilisk library. The parallel implementation has been simplified which results in a significant improvement of parallel scalability. As a drawback, the (current, at the time of writing this report) implementation restricts the domain size to a single cubic box when used in combination with adaptive grid refinement (the multigrid implementation in Basilisk does not have this restriction). This restriction makes it basically impossible to execute wind computations where the vessel has a large blockage: the distance between the inlet and the object would become too small for lateral blockages higher than 50%. This would only be an issue for specific cases where a large blockage is simulated such as the validations described in Chapter 4 using the wind tunnel results of Andersen (2013). For use in the simulator, results are required without influence of blockage, hence it is recommended to invest time in getting familiar with Basilisk in order to verify that it indeed improves upon the efficiency and scalability of Gerris.

5.3 Future work

At the moment, the wind coefficients in the simulator only account for the forces in the horizontal plane and the moment around the vertical axis, which stems from the three DOF manoeuvring model as originally implemented in the simulator. An effort is underway to extend the manoeuvring models to six DOF; wind coefficients of existing and new mathematical ship models should be upgraded to include (at least) the roll component. Although not deemed as important as roll, the addition of the vertical force and pitching moment due to wind can be added at no additional cost when the coefficients are computed using CFD. Computing wind coefficients of a ship subject to uniform flow is a good start (and Gerris is an effective tool for that), but more work is required to implement more realistic conditions (such as accounting for the effects of an atmospheric boundary layer (vertical gradient) and partial sheltering behind other objects (horizontal gradient)). By dividing the ship hull in Gerris in multiple parts (both longitudinally and vertically) and computing the forces on the parts separately, a set of partial wind coefficients is obtained (as opposed to a single set of coefficients as is the case in the simulator right now). The total force on the hull is then found by summing the contributions for each of the hull parts – where an appropriate location must be used for the reference velocity – automatically accounting for gradients in the wind field. On the side of Gerris, the required changes in the configuration files have been implemented to output partitioned wind coefficients. Partial wind coefficients must be computed for some relevant ship geometries (preferably geometries for which coefficients are available from either experiment or numerical means where the wind profile is known) and then tests have to be executed to verify that the method works as intended. This will be undertaken as part of project 21_001 (Modelleren van schepen in wind en stroming).

A1 Sample Gerris configuration file

In this section, a sample configuration file is shown that contains all necessary features to execute computations similar to the ones discussed in this report. Features that are not strictly necessary to compute pressure forces and moments on the object placed in the computational domain (such as averaging of field variables, output of images for flow visualisation) have been left out. Comments are added in the configuration file to explain the purpose of the different statements. An explanation for the different objects that can be defined in a Gerris configuration file (regarding syntax, optional arguments, ...) can be found online at `https://gfs.sourceforge.net/wiki/index.php/Object_hierarchy`.

The `Define` statements at the top of the configuration file have the same function as *Defines* in C. At every occurrence of the `Define` in the configuration, its value is substituted. This increases the flexibility of preparing multiple configuration files where (for example) only the wind angle is altered. The configuration file can be copied into multiple directories and the angle changed with a `sed` command, after which the computation can be prepared for a parallel run and added to the queue.

Listing 2 – Basic Gerris configuration file to determine pressure forces on a solid wall-mounted object in a uniform flow field.

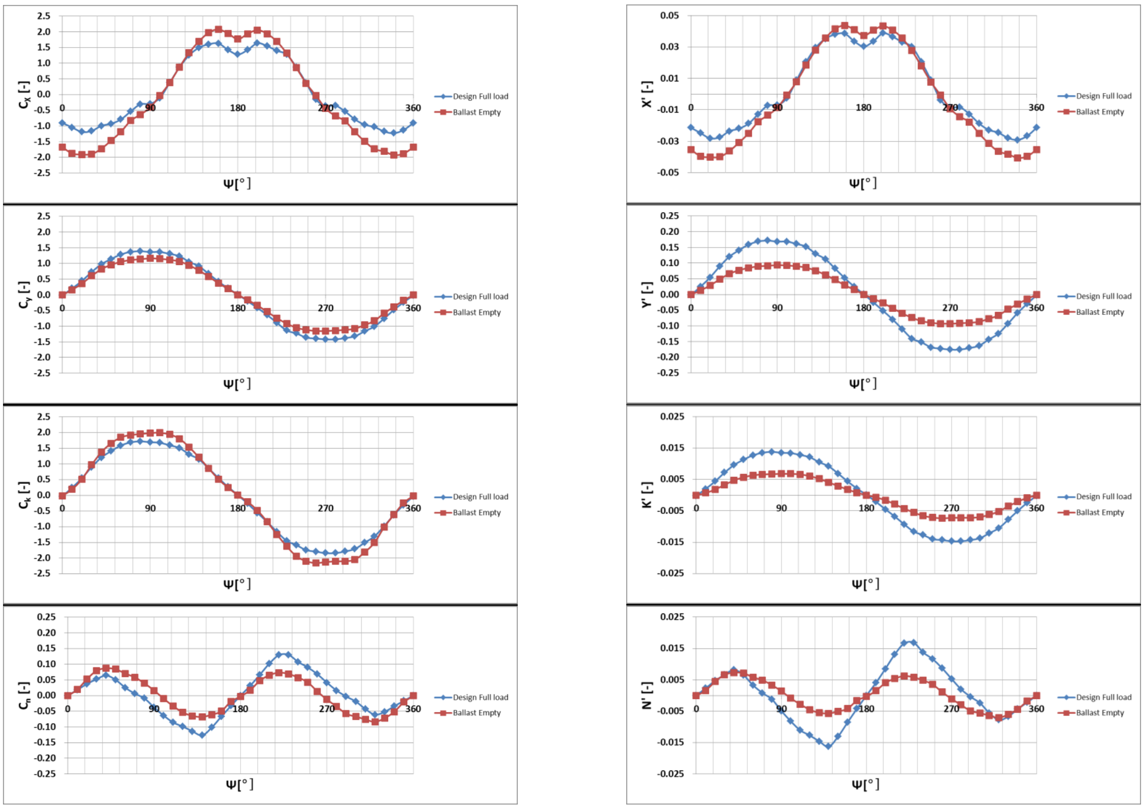
```
# wind angle of ship
Define ANGLE 30.0
# maxium refinement on ship
Define MAXREF 8
# maximum refinement in wake
Define WAKEMAXREF 7
# end time of the computation
Define TEND 20.0
# scaling of ship cad geometry: this value for validation with Andersen's
# VLCC data due to the size of the ship, the domain consist of 4 boxes
# with the ship located in the second box
Define SHIPSCALE 2.266667
# position of ship cad model: TZ = -0.5 -> move ship 0.5 down to domain bottom
Define TZ -0.5
# position of ship cad model: TX = 1.0 -> move ship to center of second box
Define TX 1.0
# square of the cylinder radius within which to refine the wake
Define CYLREFRAD 0.889
# different output files
Define OUTPUT out-ANGLE-MAXREF-WAKEMAXREF.std
Define FORCES forces-ANGLE-MAXREF-WAKEMAXREF.dat
Define FIRST_STEP first_step-ANGLE-MAXREF-WAKEMAXREF.gfs
Define LAST_STEP last_step.gfs
# simulation with 4 boxes connected with 3 'edges'
4 3 GfsSimulation GfsBox GfsGEdge {} {
  Time { end = TEND }
  # Insert the solid boundary defined explicitly by the
  # triangulated surface contained in the GTS file vlcc_anderson_01-01-02.gts
  Solid ./relative/path/to/stl_ship_geometry_converted_to_gts_format.gts {
    scale = SHIPSCALE # scale ship
    tx = TX # move to box 2 to increase space between inlet and ship
    tz = TZ # move to bottom of domain
    rz = -ANGLE # axes system has same orientation for gfs and f/m
  }
  # refine in domain 5 times, on solid to MAXREF
  Refine 5
```

```
  RefineSolid MAXREF
  # uniform initial velocity field
  Init {} { U = 1. }

  # refine the wake based on the angle of the ship (longer wake for 0 deg, shortest
  # wake for 90 deg. The coarse resolution near the domain outlet acts as an efficient
  # "sponge" layer to dampen any eddy before it exits the domain. refinement volume
  # is a vertical cylinder centered at the ship location.
  AdaptVorticity { istep = 1 } { maxlevel = ((x-TX)*(x-TX) + y*y < CYLREFRAD  ? WAKEMAXREF :
   ↪  0) cmax = 1e-2 }

  # output computation statistics
  OutputSolidStats { istep = 1 } OUTPUT
  OutputTime { istep = 1 } OUTPUT
  OutputBalance { istep = 1 } OUTPUT
  OutputProjectionStats { istep = 1 } OUTPUT

  # compute force on embedded solid
  OutputSolidForce { istep = 1 } FORCES

  # balance the number of cells for each processor
  EventBalance { istep = 1 } 0.1

  # output field variables at start and end of simulation
  OutputSimulation { istart = 1 } FIRST_STEP
  OutputSimulation { start = end } LAST_STEP
  # output timing
  OutputTiming { start = end } OUTPUT
}
# definition of domain: first box with on left side uniform inflow BC
GfsBox {
    left = GfsBoundaryInflowConstant 1
}
# box 2 and 3: nothing special
GfsBox {}
GfsBox {}
# last box with outflow on right hand side
GfsBox {
    right = GfsBoundaryOutflow
}
# connection of boxes: 2 on right side of 1, 3 on right side of 2, etc
1 2 right
2 3 right
3 4 right
```

Ã—

A2 Computation of reference parameters based on coefficient values

For cases where coefficient values as a function of the wind angle are provided for X, Y, K and N both in terms of Eqs. 10 to 13 and Eqs. 15 to 18 (as shown in Fig. 64), the parameters A_F, A_L, H_S and L_{pp} can be determined numerically. This procedure is outlined below.



(a) Area-based coefficients (b) L_{pp}-based coefficients

Figure 64 – Wind coefficients of a VLCC for two loading conditions.

Equating Eqs. 10 and 15, an expression for the frontal projected area A_F in terms of L_{pp} is obtained,

$$A_F = \frac{X'}{C_{X_{AF}}} {L_{pp}}^2. \tag{51}$$

Similarly, an expression for the lateral projected area A_L results from Eqs. 11 and 16,

$$A_L = \frac{Y'}{C_Y} {L_{pp}}^2, \tag{52}$$

while an expression for H_S follows from Eqs. 12 and 17,

$$H_S = \frac{K'}{C_K} \frac{{L_{pp}}^3}{L_{oa}}. \tag{53}$$

The equations for the yawing moment coefficients (Eqs. 13 and 18) result in a second expression for A_L,

$$A_L = \frac{N'}{C_N} \frac{{L_{pp}}^3}{L_{oa}}. \tag{54}$$

By eliminating A_L from Eqs. 52 and 54, an expression for L_{pp} is found that only contains known values,

$$L_{pp} = \frac{Y'}{C_Y}\frac{C_N}{N'}\frac{1}{L_{oa}}. \tag{55}$$

This value can then be used to estimate values of A_F, A_L and H_S with the aid of Eqs. 51, 52 and 53. Estimates for these four parameters are shown in Figs. 65 and 66. Note that at zero-crossings of the coefficients for certain wind angles, specific values of the parameters may deviate significantly from the average. These outliers were be eliminated before averaging. The mean values for the parameters thus obtained are L_{pp} = 384.16 m, A_F = 3489.36 m^2, A_L = $18\,083.45$ m^2, H_S = 24.38 m.

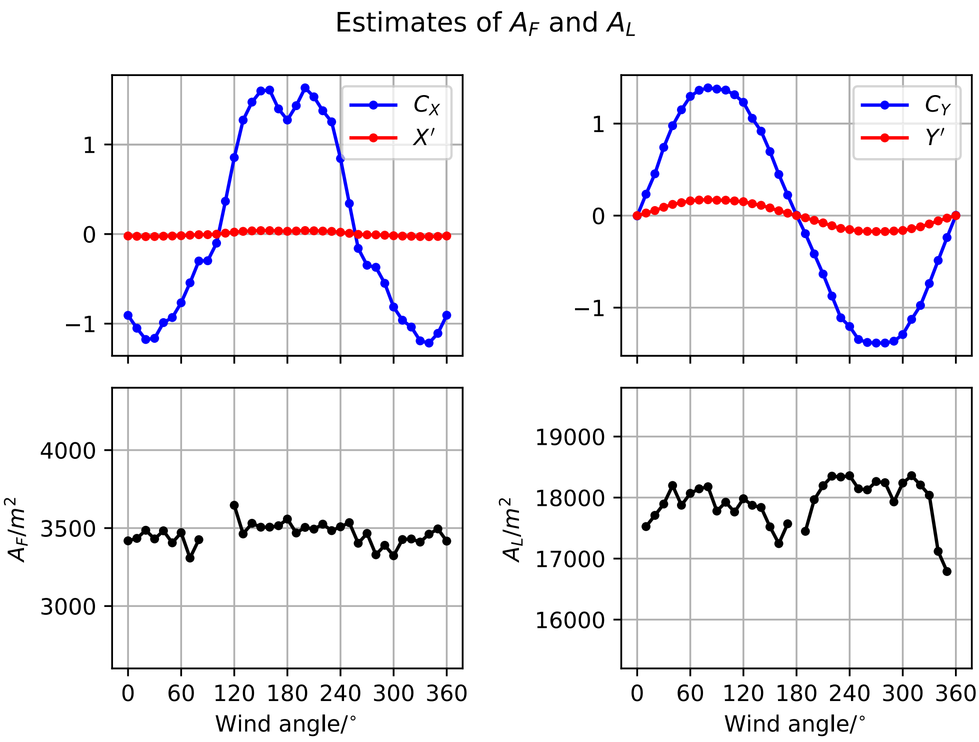


Figure 65 – Wind tunnel coefficient values for longitudinal and lateral force and estimates of A_F and A_L.

The lever arm of the lateral force can be computed by dividing the values of K by Y,

$$\frac{K}{Y} = \frac{C_K q A_L H_S}{C_Y q A_L} = \frac{C_K H_S}{C_Y}, \tag{56}$$

or

$$\frac{C_K}{C_Y} = \frac{K}{Y H_S}, \tag{57}$$

which is the lever arm of the lateral force relative to the geometric centre of gravity of the lateral projected area H_S. If this value is (significantly) larger than one, the application point of the lateral force is located above H_S, which can only happen if the vertical velocity profile in which the model was tested resembles an atmospheric profile where the upper half of the vessel experiences significantly higher dynamic pressures than the lower half. For the wind coefficients shown in Figs. 64, the lever arm of the lateral force is shown in Fig. 67. The average value is approximately 1.27, which means that the model tested in the wind tunnel was subject to an atmospheric profile.

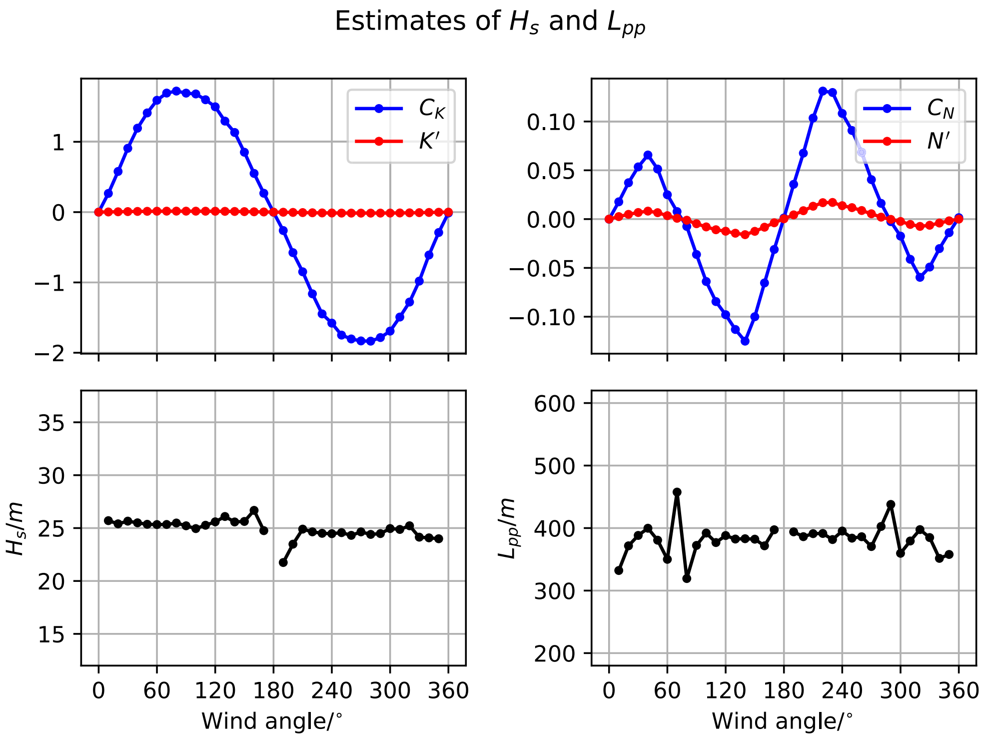


Figure 66 – Wind tunnel coefficient values for overturning and yawing moment and estimates of H_S and L_{pp}.

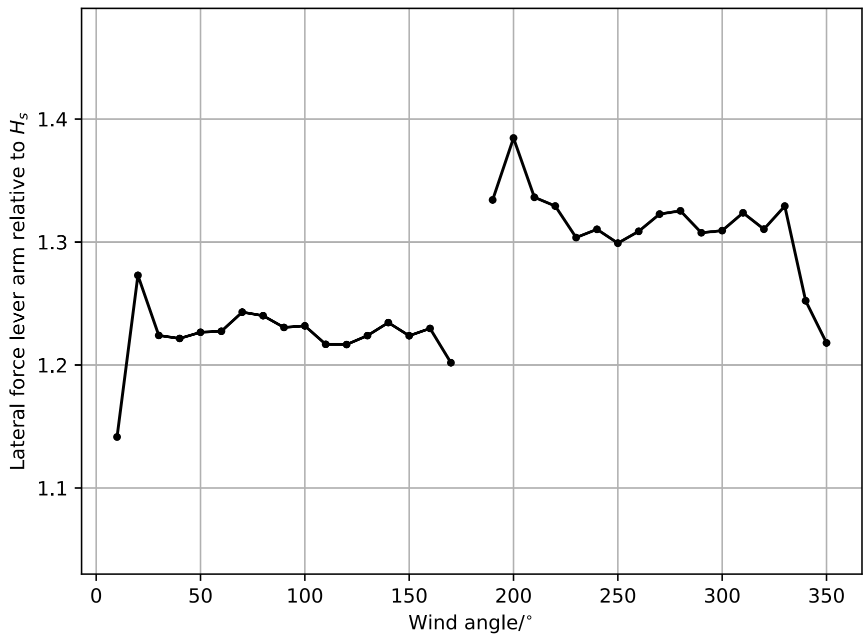


Figure 67 – Lever arm of the lateral force relative to the geometric centre of gravity of the lateral projected area.

A3 Additional results for the bottom-mounted rectangular block

For $\varphi = 0°$, $20°$, $30°$, $50°$, $60°$, $70°$, $80°$ and $90°$, the time histories of the four coefficients $C_{X_{AF}}$, C_Y, C_K and C_N (and their cumulative moving averages) are shown in Figs. 68 to 75. The time histories for $\varphi = 10°$ and $40°$ were shown in section 4.1.2.

Visualisations of the flow field in a plane parallel to the bottom halfway between the bottom and top of the cube for $\varphi = 60°$ to $90°$ are shown in Fig. 76.

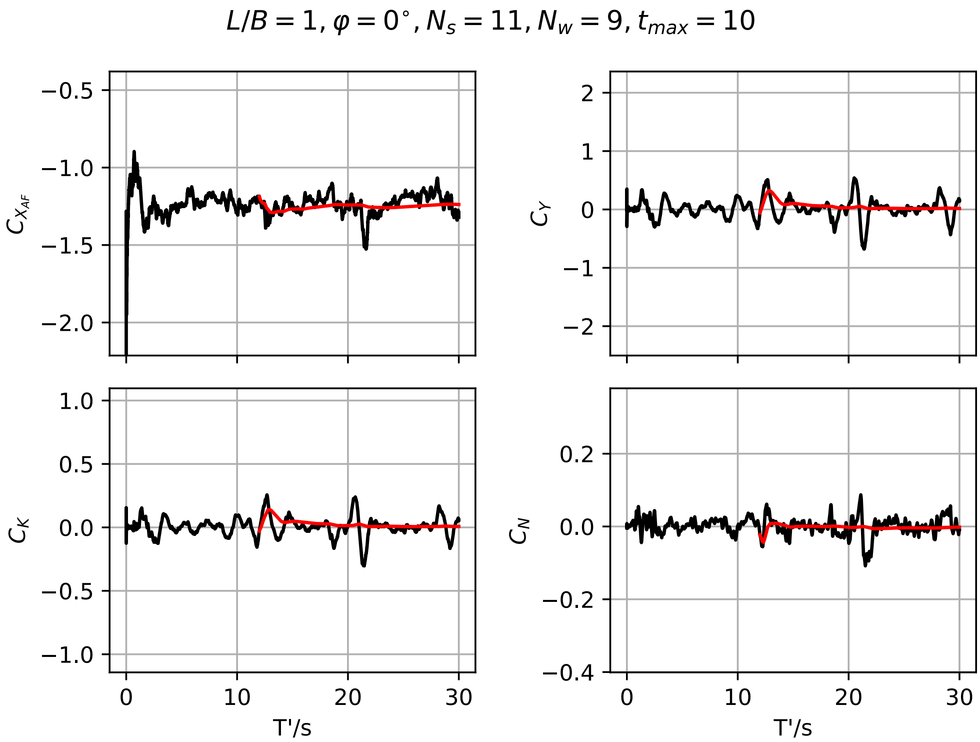


Figure 68 – Time histories and cumulative moving averages of the four coefficients for the wall-mounted rectangular block ($L/B = 1$) for $\varphi = 0°$.

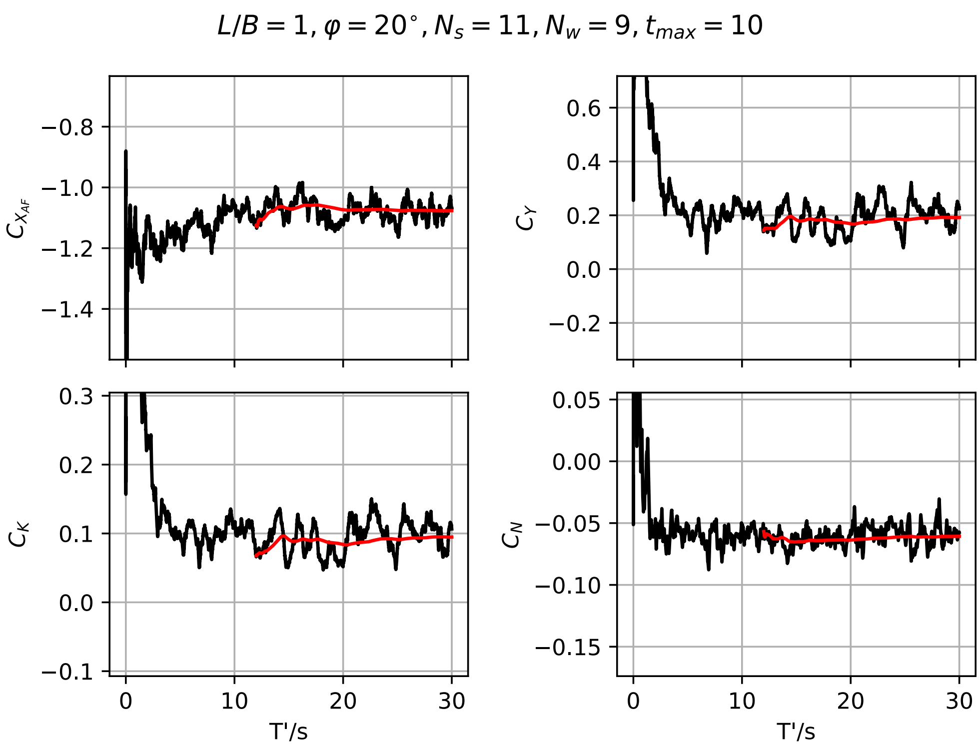


Figure 69 – Time histories and cumulative moving averages of the four coefficients for the wall-mounted rectangular block $(L/B = 1)$ for $\varphi = 20°$.

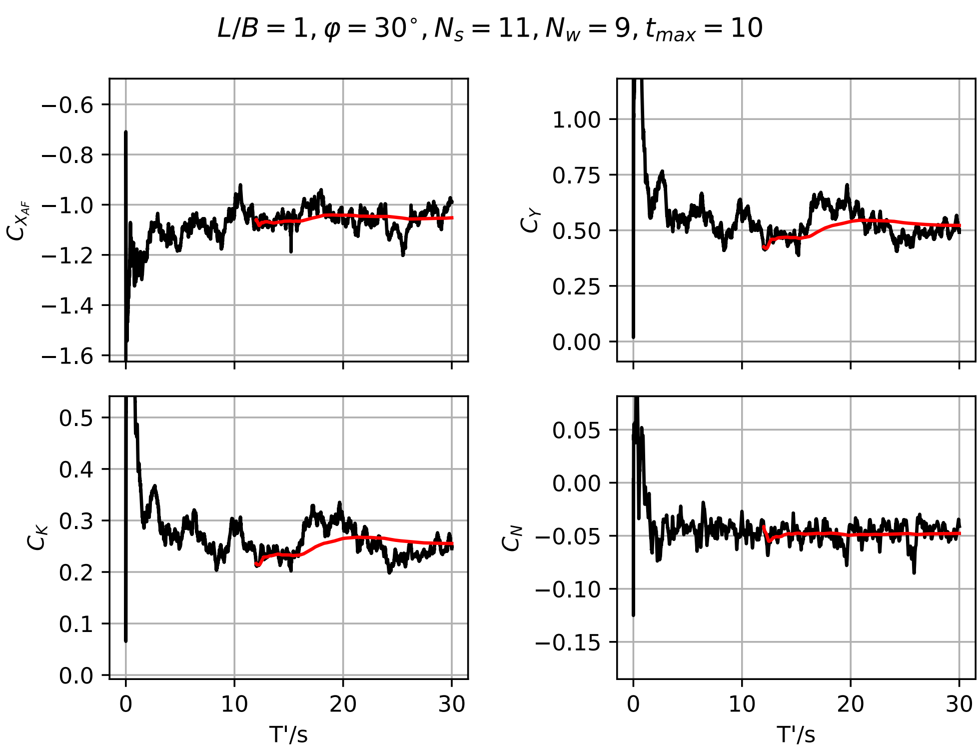


Figure 70 – Time histories and cumulative moving averages of the four coefficients for the wall-mounted rectangular block $(L/B = 1)$ for $\varphi = 30°$.

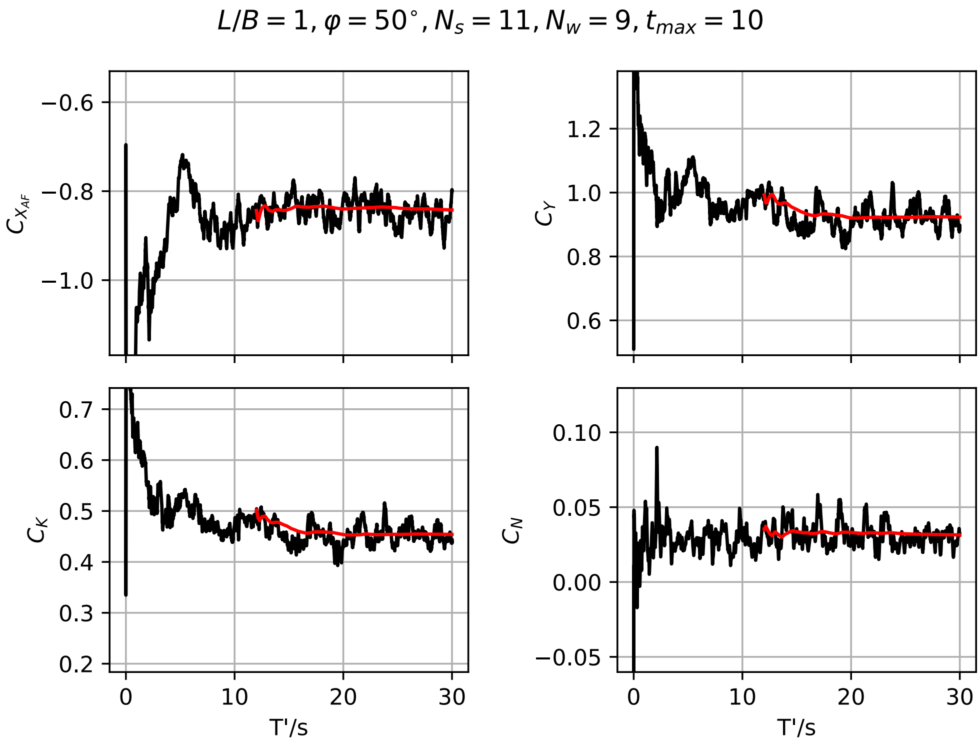


Figure 71 – Time histories and cumulative moving averages of the four coefficients for the wall-mounted rectangular block $(L/B = 1)$ for $\varphi = 50°$.

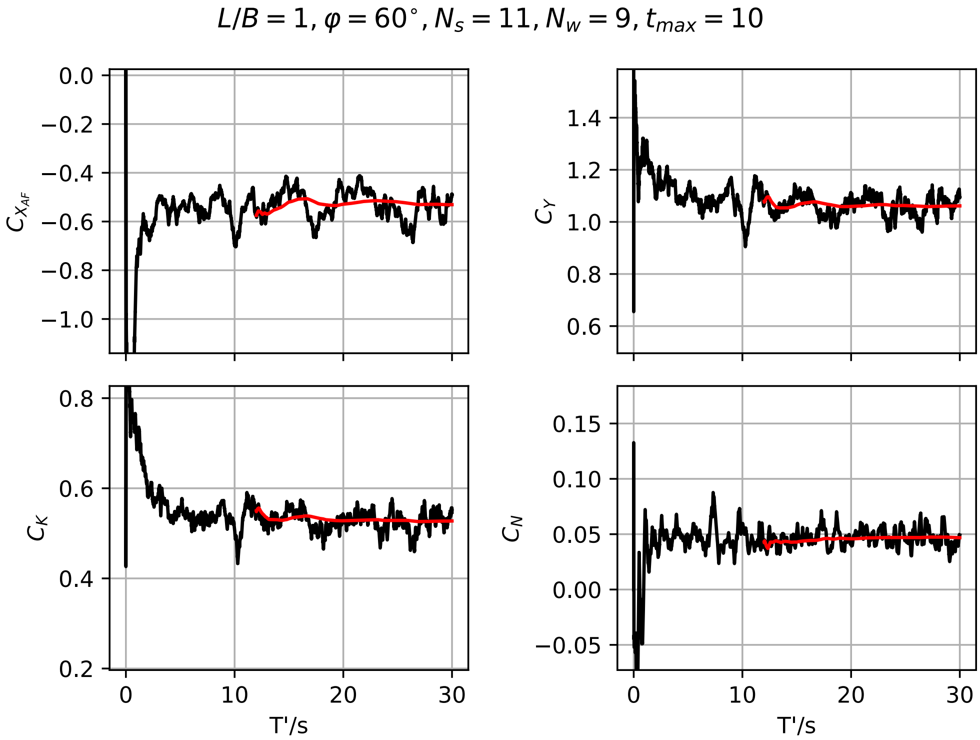


Figure 72 – Time histories and cumulative moving averages of the four coefficients for the wall-mounted rectangular block $(L/B = 1)$ for $\varphi = 60°$.

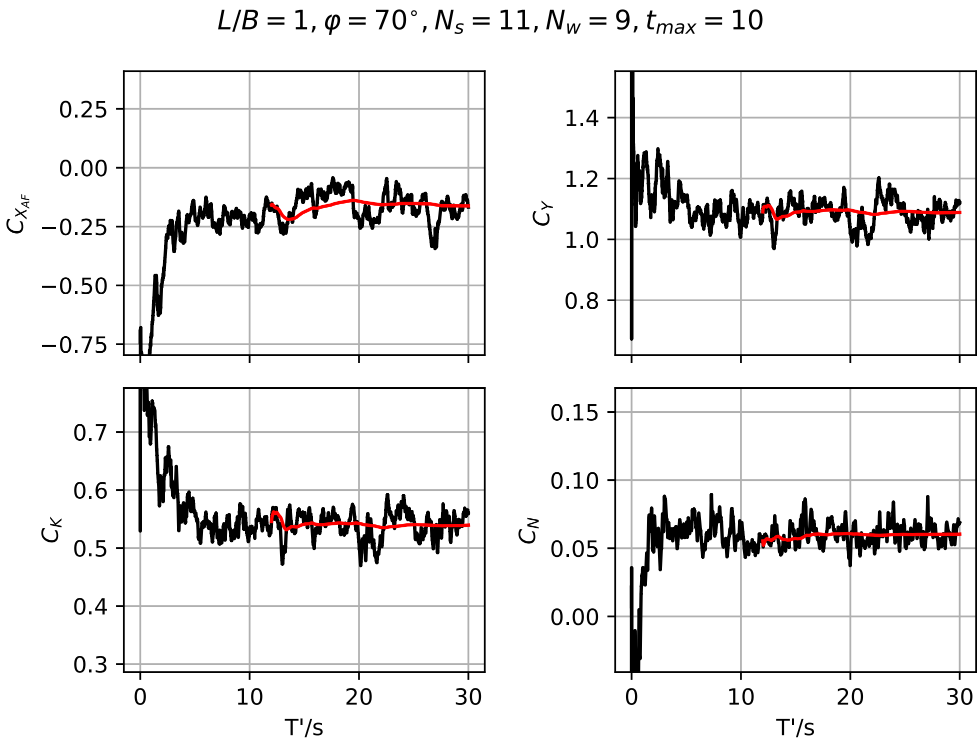


Figure 73 – Time histories and cumulative moving averages of the four coefficients for the wall-mounted rectangular block $(L/B = 1)$ for $\varphi = 70°$.

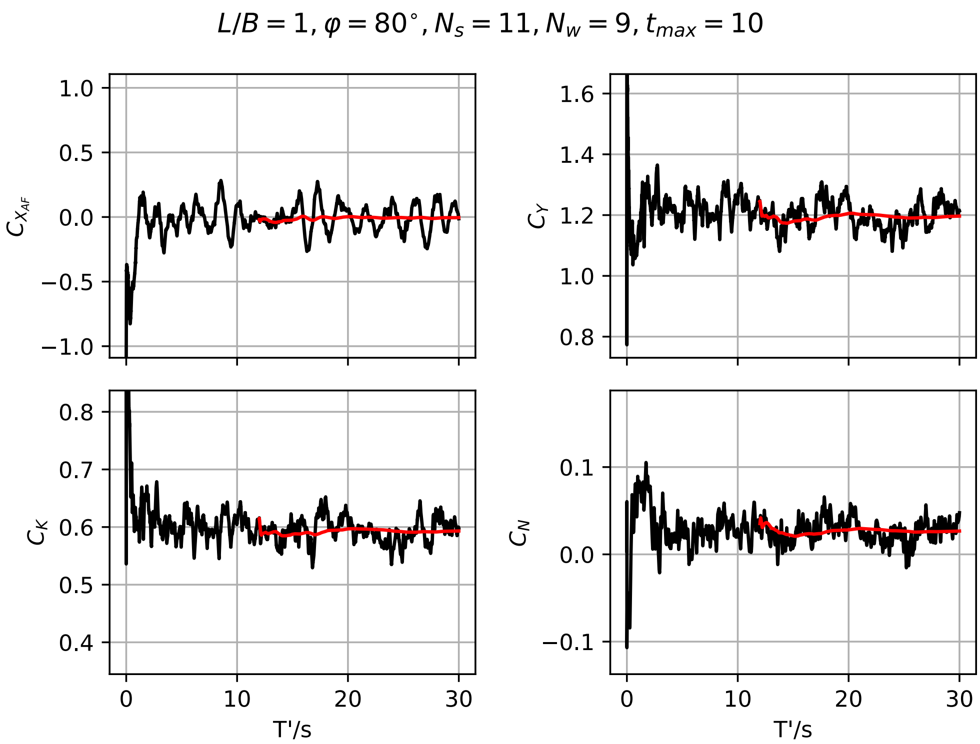


Figure 74 – Time histories and cumulative moving averages of the four coefficients for the wall-mounted rectangular block $(L/B = 1)$ for $\varphi = 80°$.

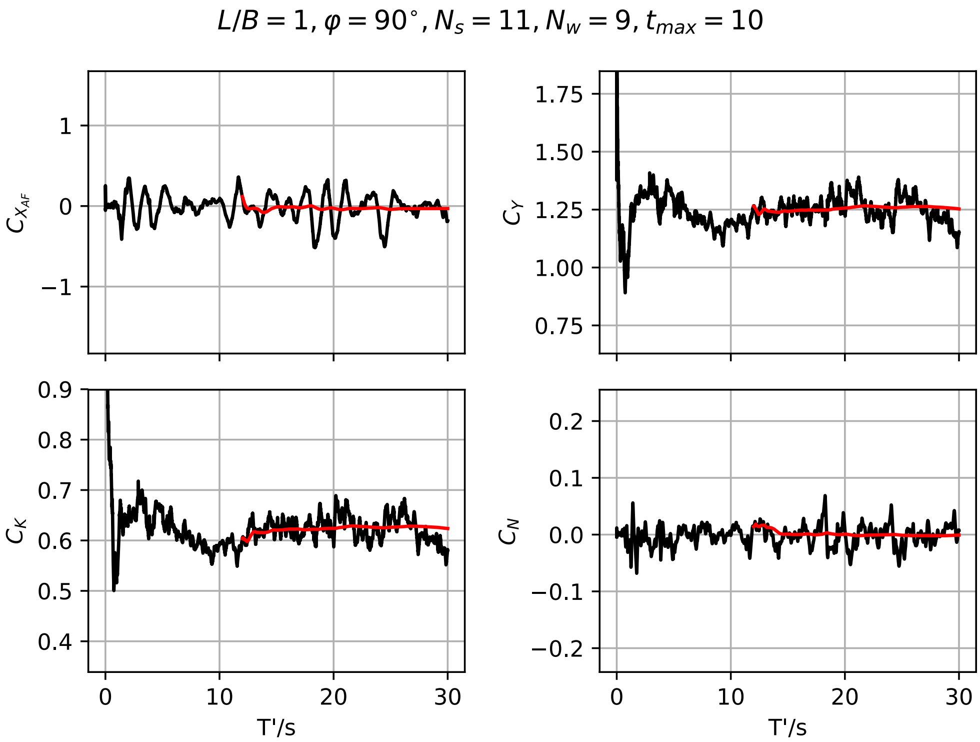


Figure 75 – Time histories and cumulative moving averages of the four coefficients for the wall-mounted rectangular block $(L/B = 1)$ for $\varphi = 90°$.

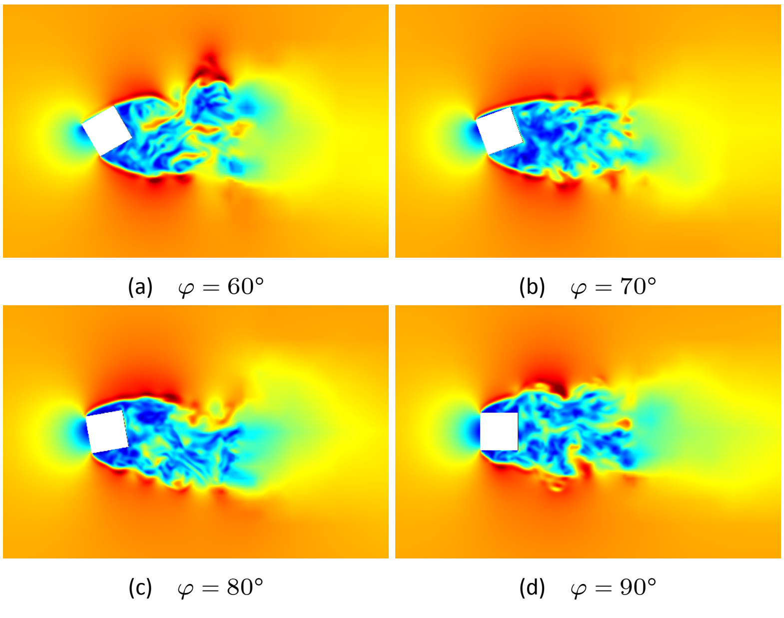


Figure 76 – Snapshot of velocity magnitude in a horizontal plane halfway between the bottom and top of the cube.

A4 Influence of inlet boundary condition on the solution

Figure 77 shows a vertical cross section in the computational domain at $y = 0$ that shows the magnitude of the vorticity for a simulation where the inflow boundary condition was set with the generic `GfsBoundary`. The cells in this cross section are outlined with a black boundary. This view shows five spurious jets emerging from the inlet in front of and slightly above the ship. The location of these jets (and whether or not they appear at all) seems to be random. If they occur in front of the the hull, they can affect the long-term convergence of the forces on the vessel. When using the more specific `GfsBoundaryInflowConstant` boundary condition (which sets Dirichlet boundary conditions for the three velocity components), the spurious jets are not present.

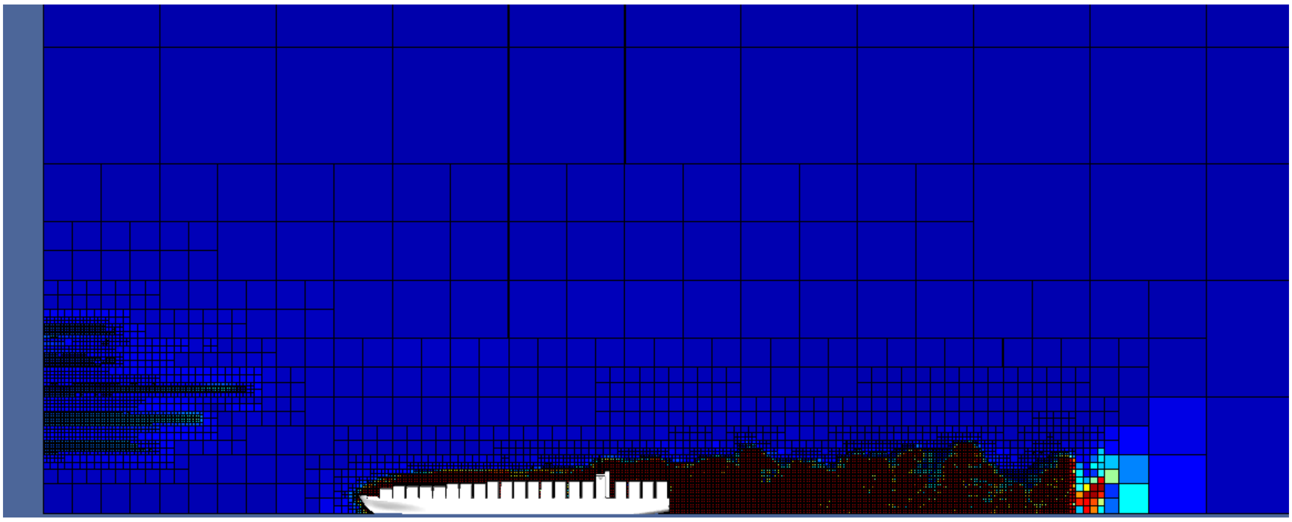
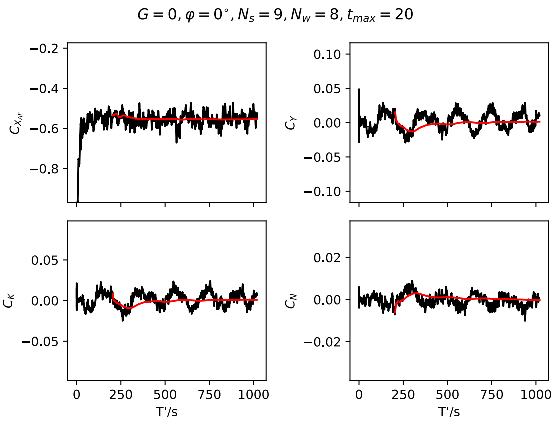


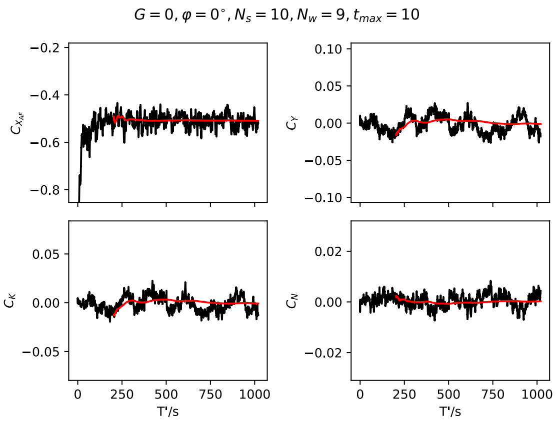
Figure 77 – Snapshot at $t = 10$ of the magnitude of vorticity in a vertical plane at $y = 0$ for the VLCC of Andersen (2007) at $\varphi = 0°$ that shows spurious jets originating from the inlet (left) ahead of the vessel.

A5 Influence of ship scale w.r.t. domain size: time histories

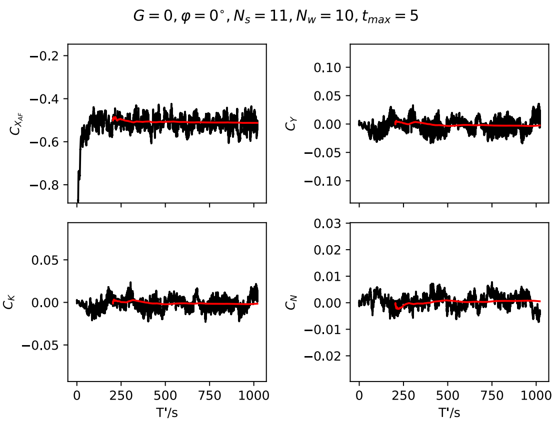
Figures 78 to 81 show the time histories of the force and moment coefficients together with the cumulative moving averages for the three cases of the VLCC of Andersen where the influence of blockage on the solution is investigated.



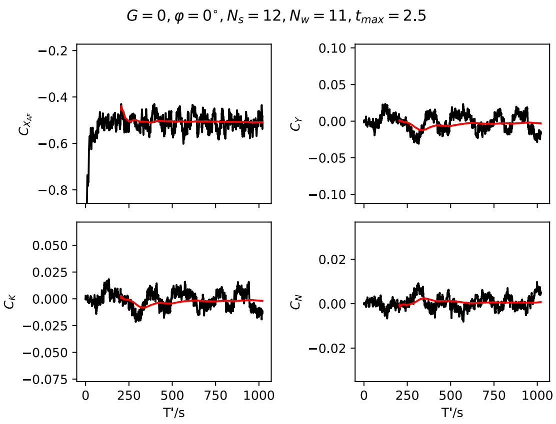
(a) L1



(b) M1



(c) S1



(d) S2

Figure 78 – Time traces of force and moment coefficients including cumulative moving averages on the VLCC hull for four length scaling values L_{ref} for $\varphi = 0°$.

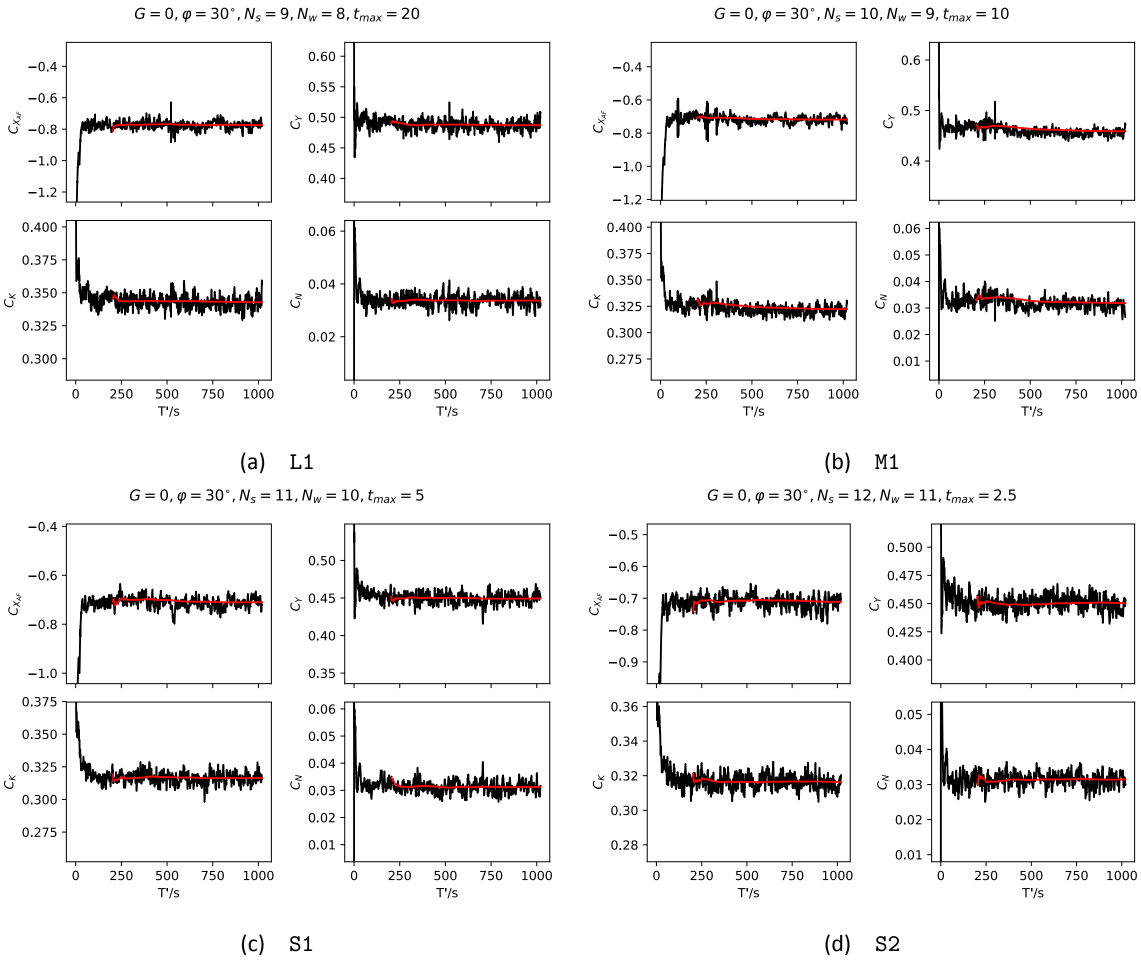


Figure 79 – Time traces of force and moment coefficients including cumulative moving averages on the VLCC hull for four length scaling values L_{ref} for $\varphi = 30°$.

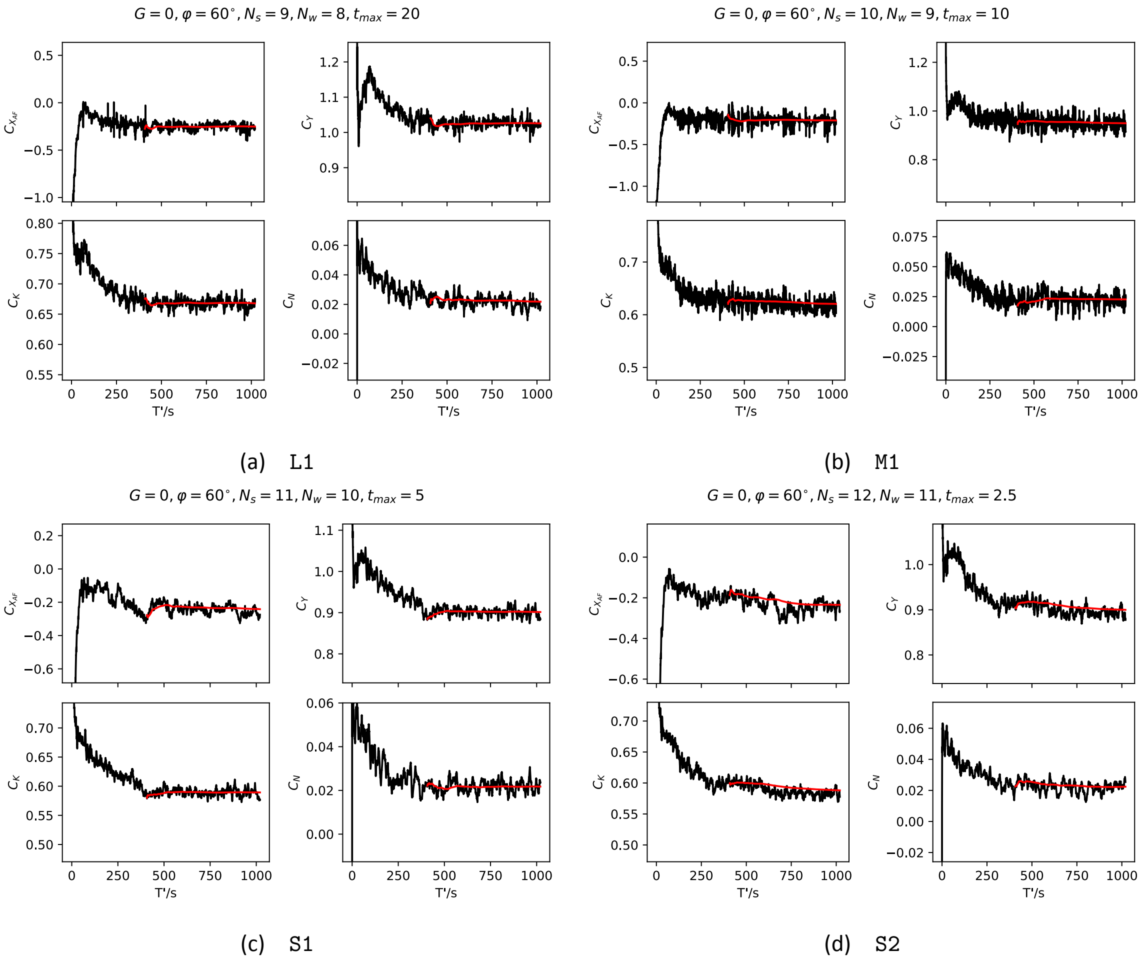
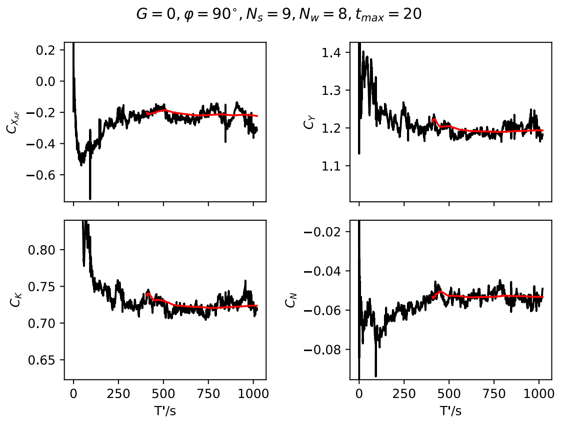
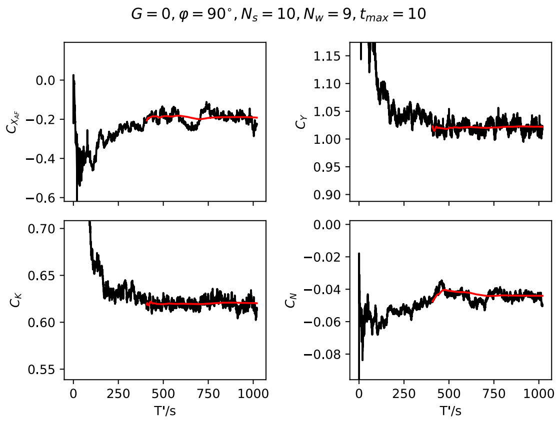


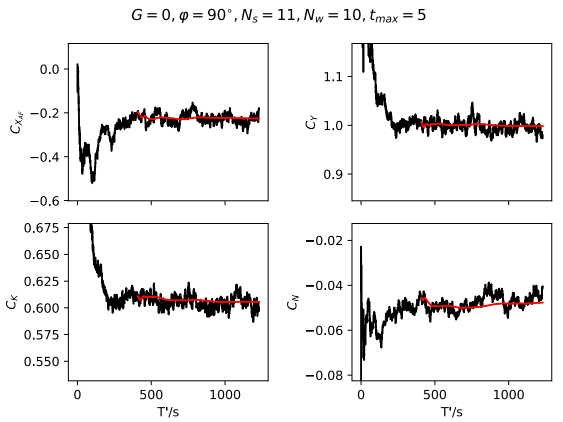
Figure 80 – Time traces of force and moment coefficients including cumulative moving averages on the VLCC hull for four length scaling values L_{ref} for $\varphi = 60°$.



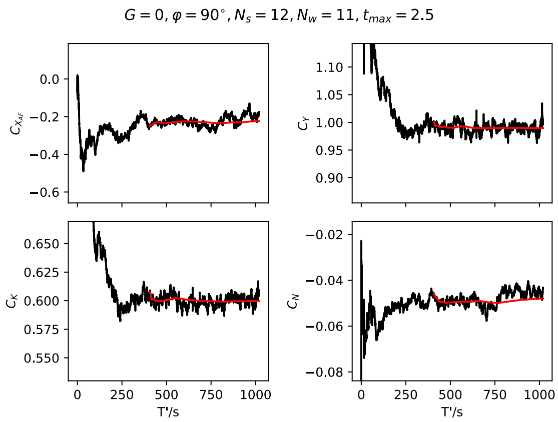
(a) L1



(b) M1



(c) S1



(d) S2

Figure 81 – Time traces of force and moment coefficients including cumulative moving averages on the VLCC hull for four length scaling values L_{ref} for $\varphi = 90°$.

A6 Review of Janssen *et al.* (2017) and Andersen (2013)

A6.1 Introduction

Andersen (2013) presents an investigation on the influence of the container configuration on the deck of a post-panamax container ship through a series of wind tunnel tests on a 1/450 scale model. Measurements were corrected for the effects of the boundary layer and wind tunnel blockage. The presented results show the dependence of the wind coefficients on the container configuration. Andersen (2013) is based on the Master Thesis report by the same author (Andersen, 2007, (in Danish)) however, the original thesis is missing in the reference list of Andersen (2013).

Janssen *et al.* (2017) present CFD simulations of wind loads on a container ship and validate the method using the experimental data of Andersen (2013). In the research of Janssen *et al.* (2017), the commercial solver Fluent 15 is used and the 3D steady RANS equations are solved where the realizable $k-\epsilon$ turbulence model is used for closure of the governing equations. The computational domain has the same dimensions as the wind tunnel and the results obtained thus are compared with computational results obtained in a very large domain (one where no blockage corrections must be applied). The former results are corrected with the same formulae as used for the experimental results (Andersen, 2013). From the good agreement between the corrected EFD data and the corrected CFD results obtained in the small domain, Janssen *et al.* conclude that the ESDU bluff body correction formula employed by Andersen severely underestimates the maximal wind load on the vessel due to the significant difference between the CFD results obtained in the large domain and the blockage-corrected CFD results obtained in the small domain.

A condensed version of this appendix was submitted to *Ocean Engineering* but rejected on the basis that a comment paper should be received within three months of the online publication of the paper under discussion, which was not the case.

A6.2 Wake expansion factor m

The bluff body blockage correction follows the approach in Engineering Sciences Data Unit (1998),

$$C_c = C_s \cdot \left(1 - \frac{mS}{A}\right), \tag{58}$$

where C_c is the corrected coefficient, C_s the original coefficient, S the projected area perpendicular to the flow direction, A the cross-section of the wind tunnel and m an expansion factor for the wake. Note that in the publication of Andersen, Eq. 58 is erroneously shown without brackets as follows[32]:

$$C_c = C_s \cdot 1 - \frac{mS}{A}. \tag{59}$$

According to Janssen *et al.* (2017), Anderson used a value of 3.83 for the wake expansion factor m. Andersen (2013) indeed contains a value of 3.83. However, in the original MSc. Thesis of Andersen a value of 2.83 is used. The computation of the numerical value of m is detailed in Andersen (2007) and repeated here:

$$m = 3.2 - 0.05\frac{H}{b}. \tag{60}$$

[32]The Master thesis of Andersen contains the formula with brackets, hence this seems to be an unfortunate typographical error.

where H and b are the dimensions of a flat plate. According to Andersen, H corresponds to L_{oa} (336.7 m) and b to the ship's height (45.6 m). Substituting these values leads to

$$m = 3.2 - 0.05\frac{336.7}{45.6} = 2.83. \tag{61}$$

Given that both b and H in Eq. 60 are linear dimensions and thus always have positive values, the upper limit for m is 3.2.

Andersen (2007) contains for a single container configuration (01-01-02) the uncorrected measurements of forces and moments. It was verified that the corrected values can only be obtained by using a value of $m = 2.83$ for the wake expansion factor. Hence, it appears that the value of 3.83 as published in Andersen (2013) is a typographical error that was copied by Janssen *et al.* (2017).

A6.3 Reference area

Janssen *et al.* uses a different method to compute the projected area for the vessel from the method used by Andersen (2013). Janssen *et al.* computes the projected area perpendicular to the flow direction from the frontal and lateral area using the following equations:

$$S_x = A_f|\cos\varphi| \tag{62}$$
$$S_y = A_s|\sin\varphi| \tag{63}$$

where φ is the wind angle on the ship and A_f and A_s are the frontal and lateral projected areas of the vessel. For the yawing moment the sum of the two is used,

$$S_v = A_f|\cos\varphi| + A_s|\sin\varphi|. \tag{64}$$

Andersen (2007) and Andersen (2013) computes the projected area S of the vessel without the absolute values:

$$S = A_f\cos\varphi + A_s\sin\varphi \tag{65}$$

and always uses the value computed as such in Eq. 60.

The absolute value for the cosine and sine terms in Eqs. 64 is required because the cosine term is negative between $90°$ and $270°$. Without the absolute values, the reference area for $180°$ would be negative and equal to $-A_f$. It is assumed that the omission of the absolute values in Eq. 65 is only a typographical error.

Table 1 in Janssen *et al.* (2017) contains the numerical values of the correction factors that were used to correct the CFD results for blockage in the domain corresponding to the wind tunnel geometry. The data is shown in Fig. 82. Some remarkable conclusions can be drawn from this graph:

- when $\varphi = 90°$, the longitudinal force coefficient C_X is not corrected for blockage;
- when $\varphi = 0°$ and $180°$, the lateral coefficient C_Y is not corrected for blockage;
- the moment coefficient C_N is the only one that is always corrected, irrespective of the relative wind angle.

This is remarkable since Andersen (2013) clearly states that all coefficients were corrected for their associated blockage: a single correction value should be used at any wind angle for each of the three components instead of three different values. This is also stated in Engineering Sciences Data Unit (1998).

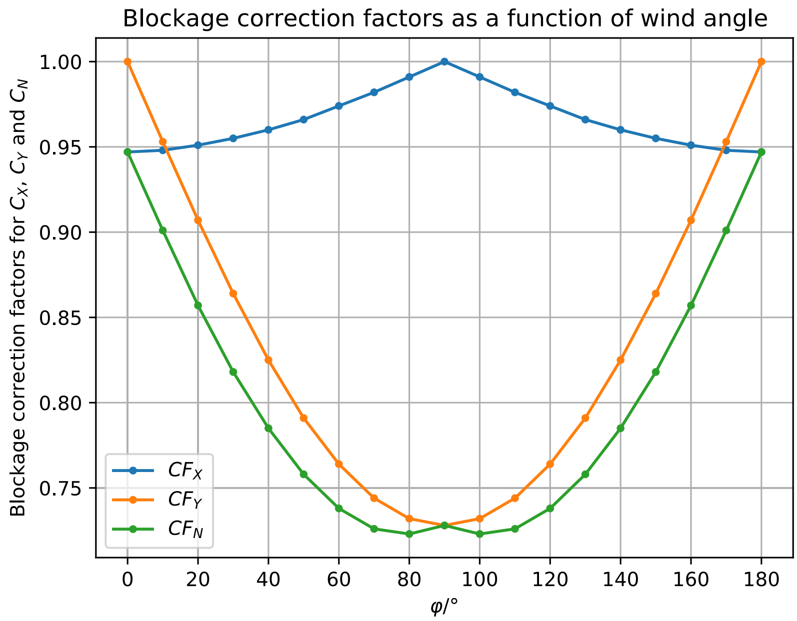


Figure 82 – Correction factors used by Janssen *et al.* (2017) to account for blockage.

A6.4 Verification

To verify that the value of $m = 3.83$ is not only a typographical error in Janssen *et al.* (2017), but that this value was in fact used to correct the CFD results for blockage, the value of m in Eq. 60 is computed using the values of C_c and C_s from Fig. 12 in Janssen *et al.* The data of this graph was digitised and is shown here in Fig. 83. The blockage-corrected CFD results and the blockage-corrected EFD results show excellent agreement for wind angles from $60°$ to $120°$. The agreement deteriorates somewhat for wind angles outside this range.

Solving Eq. 58 for parameter m gives:

$$m = \frac{A}{S}\left(1 - \frac{C_c}{C_s}\right). \tag{66}$$

Fig. 84 shows the value for the wake expansion factor m computed using Eq. 66 with the data shown in Fig. 83. The large errors for the small and large wind angles are likely caused by the process of digitising the original graph. In any case, in the current investigation only the lateral forces are investigated (similar to the validation of Janssen *et al.*), and for a large range of lateral wind, the estimated values of m are close to 3.83.

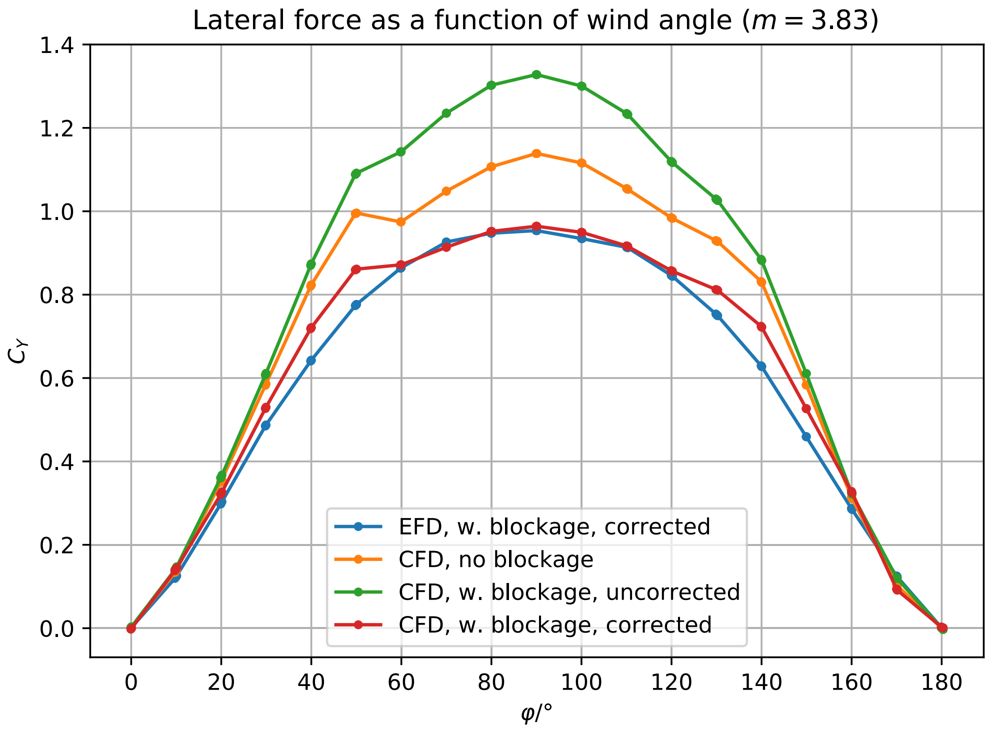


Figure 83 – Original data of Fig.12 of Janssen *et al.* (2017).

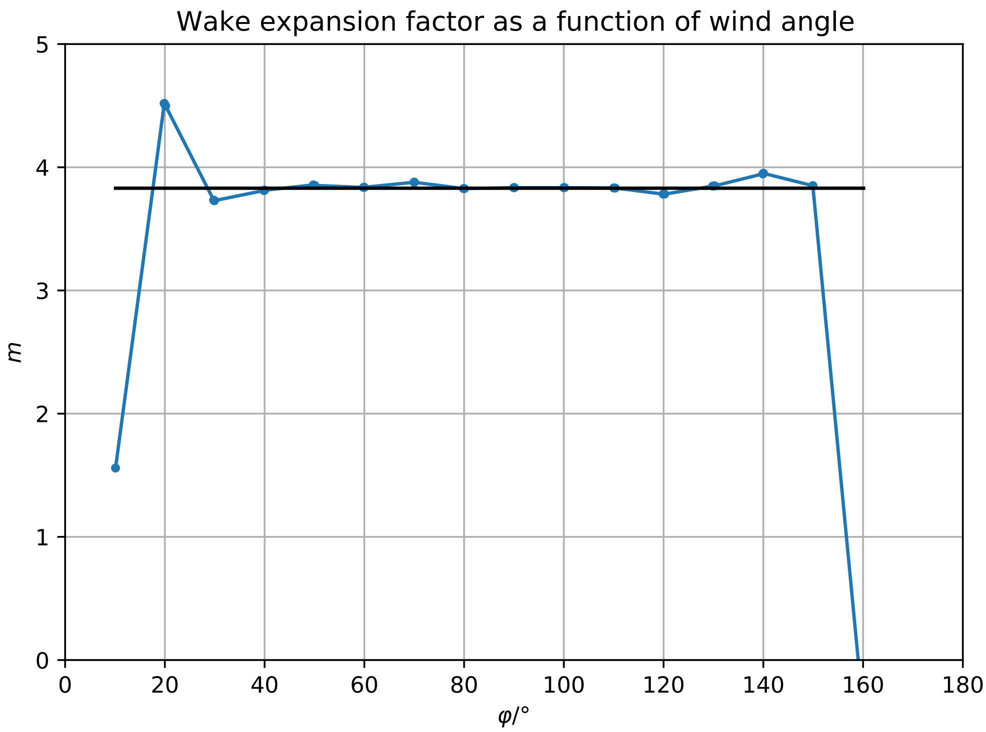


Figure 84 – Wake expansion factor m deduced from Fig 83.

The blockage correction factors as a function of wind angle using $m = 2.83$ and a single, consistent projected surface area are shown in Fig. 85 together with the original values as used in Janssen *et al.* (2017). The difference between the original and new blockage correction factors is largest for the longitudinal force, where differences as much as $20\,\%$ will occur. For the lateral force, the original and new correction factors cross near $30°$ and $150°$, for results between these bounds, the corrected values will be larger while for results beyond the bounds, corrected values will be smaller. For the yawing moment, all corrections will be smaller.

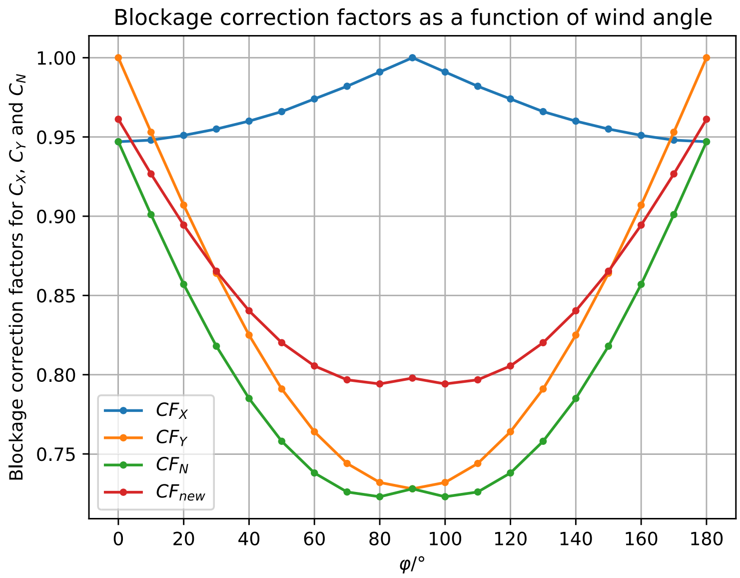


Figure 85 – Original blockage correction factors used by Janssen *et al.* (2017) and new values using the correct value for m and consistent projected surface areas.

If the CFD results obtained in the small domain are corrected with the new correction factors, there is a significant offset with the experimental data (Fig. 86). The differences between the CFD results in an unrestricted domain and the blockage-corrected results obtained in a restricted domain are now significantly smaller, indicating that the *ESDU 80024* correction performs reasonably well, although differences remain.

To get optimal agreement between the CFD results computed in the unrestricted domain and the results computed in the wind tunnel domain, a regression analysis was executed to find the optimal value of m under the assumption that a single value for the blockage correction factor can be used for the full range of wind angles (which, as discussed further, may not be fully warranted). Similarly, the optimal value for m was computed to get the best match with the corrected EFD results. For the former case, the optimum value is $m_{opt,CFD} = 1.84$ while for the latter case, it is $m_{opt,EFD} = 3.90$ (similar to the value used by Janssen). The results are presented in Fig. 87.

The conclusion is that Janssen *et al.* (2017) are too optimistic about their own results. Their numerical setup does not perform as good as they believe it does. This could have multiple issues:

- it was not verified by Janssen *et al.* that the velocity profile at the location of the ship was in fact the correct velocity profile as measured by Andersen: Janssen *et al.* only gives details about the velocity profile defined at the inlet, not how it evolves along the length of the computational domain;
- the use of a steady-state RANS setup for a problem that is inherently unsteady, may give results in a relatively short time period but those results will likely be suboptimal.

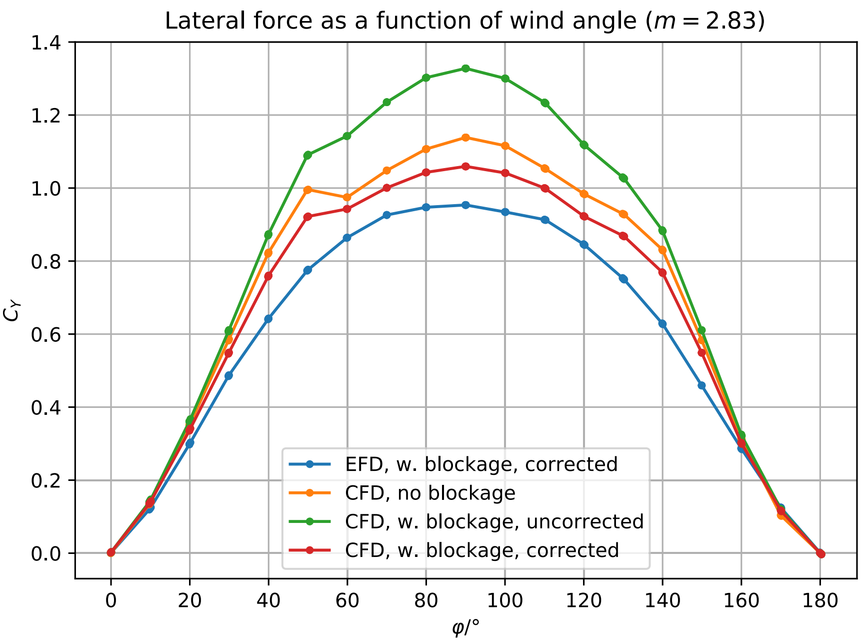


Figure 86 – Modified data of Fig. 12 of (Janssen *et al.*, 2017) with $m = 2.83$ instead of $m = 3.83$, using a consistent projected area.

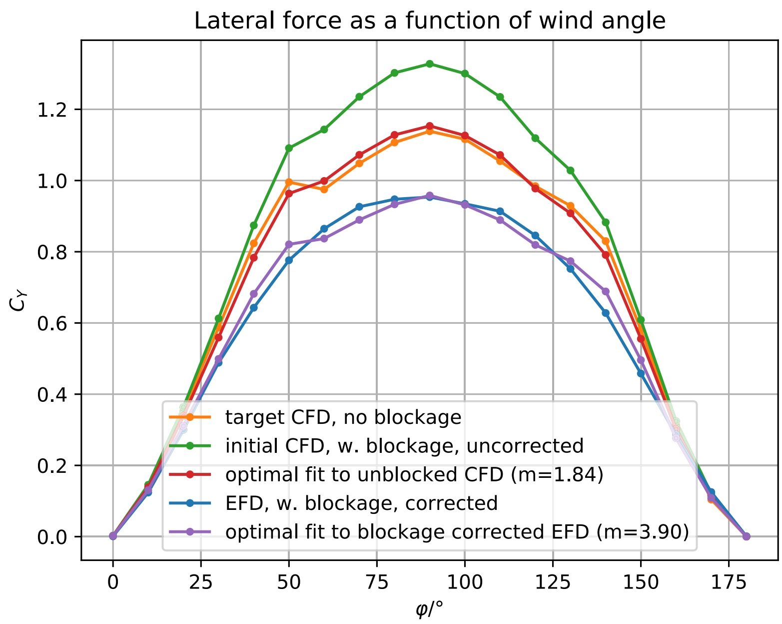
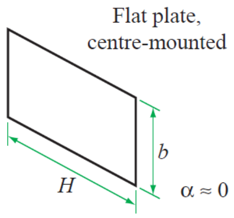
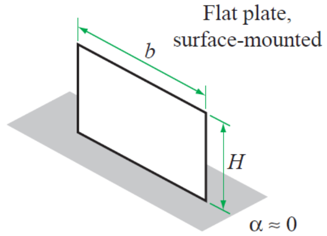
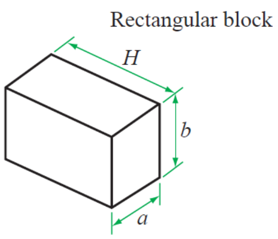


Figure 87 – Optimal fits of m for the best match of CFD results obtained in the small domain with CFD results obtained in the large domain and results obtained with EFD.

A6.5 Details of the *ESDU 80024* method

The blockage-corrected CFD results obtained by Janssen *et al.* (2017) correspond better to the CFD results obtained in the unrestricted domain for $m = 2.83$ than when $m = 3.83$ although, as noted before, differences remain. In this section, the applicability of the blockage corrections as presented in *ESDU 80024* and used by Andersen (2007) will be investigated. In Table 10.1.2 of *ESDU 80024* the blockage correction factors ε and m for three-dimensional flows are summarized. The values for a flat plate (both centre-mounted and surface-mounted) and for a rectangular block and the conditions under which they can be applied, are repeated in Table 9[33].

Table 9 – Blockage correction factors ε and m for three-dimensional flow (Engineering Sciences Data Unit, 1998).

Body shape	ε; m; maximum S/A
	$\varepsilon = 2.84 - 0.07H/b$; $m = 3.20 - 0.05H/b$ for $H/b \leq 20$; for $S/A \leq 0.10$.
	$\varepsilon \approx 2.37$; $m \approx 2.84$ for $1/3 \leq H/b \leq 3$; for $S/A \leq 0.10$. For $H/b > 3$ use value for centre-mounted plate of $2H/b$ aspect ratio.
	Use value for flat plate of same H/b provided flow reattachment does not occur, $a/b \leq 2.5$; for $S/A \leq 0.10$.

When flow separation occurs at or ahead of the maximum cross-sectional area of the body, it is recommended to use the method of Maskell (Engineering Sciences Data Unit, 1998, (§ 4.1)), which gives the blockage correction ratio as

$$\frac{C_{Ff}}{C_F} = \frac{1}{1 + \varepsilon C_D S/A}. \tag{67}$$

The alternative formulation of Cowdrey (Engineering Sciences Data Unit, 1998, (§ A2.3)) is convenient when measured values of C_D are not available,

$$\frac{C_{Ff}}{C_F} = 1 - mS/A, \tag{68}$$

where m is the wake expansion ratio which is approximately equal to (maximum separation bubble area)/(body reference area), B/S. This wake blockage correction underestimates the required correction for bodies with a length to width ratio of two or more. According to Engineering Sciences Data Unit (1998, p. 13), this can be compensated by adding a solid-body blockage correction (which is discussed in more detail in Barlow *et al.* (1999)).

[33]The original table contains additional data for near-circular and triangular flat plates and references to the data used for the derivation of the values.

For a rectangular block, it is advised to use the values for the centre-mounted plate, which is also what Andersen does in her Master thesis. The alternative formulation of Cowdrey (Eq. 68) is used in combination with Eq. 66 (Andersen, 2013).

Note that the flat-plate formulations are only valid when the plate is oriented perpendicular to the oncoming flow ($\alpha \approx 0$). Furthermore, for small wind angles (either near $0°$ or near $180°$), the restriction for $a/b \leq 2.5$ does not hold for (container) ships, where $a/b \geq 6$.

One could argue that using Eq. 60 with H the ship length and b the ship height is not entirely correct, as the formula holds for a centre-mounted flat plate and not a surface-mounted plate. For a surface-mounted flat plate, m is constant and ≈ 2.84, which is almost equal to Andersen's value of 2.83. However, the range of H/b values for which it holds is too restricted to be useful for ship models.

Gould (1970) notes that small wall-mounted rectangular plates where the height is smaller than three times the width all have the same wake-blockage factors provided the wall boundary layer is thin. This would mean that the value of m for the surface-mounted plate is valid outside the range as given in Table 9 if the wall boundary layer is thin as compared to the height of the wind tunnel model. This is not the case for the experiments of Andersen (which make explicit use of the naturally occurring wall boundary layer in the wind tunnel to simulate an atmospheric boundary layer).

Hence, no definitive conclusion can be drawn regarding a suitable wake blockage correction factor from Engineering Sciences Data Unit (1998) for the experiments. As such, the results presented by Andersen (2007) and Andersen (2013) are only useful as validation for CFD computations if the exact same domain geometry is used in combination with the same wake blockage correction. As noted in the beginning of this chapter, the experimental data of Andersen has been used for this purpose, but as shown, when the errors are corrected the conclusion is that the setup of Janssen *et al.* (2017) (steady RANS computations) cannot accurately predict the wind coefficients of Andersen (2013).

A6.6 More collateral damage

In a more recent paper, Ricci *et al.* (2020) build upon the work presented in Janssen *et al.* (2017) by simulating a cruise ship in the harbour of Rotterdam. The validation related to the chosen solver settings and geometry details are only briefly discussed (these appeared in more detail in Janssen *et al.* (2017)), but are summarized by stating that:

> Matching the Windtunnel (WT) test section and the computational domain and applying the same blockage corrections to WT and CFD results was important in order not to compromise the WT vs. CFD comparison.

Ricci *et al.* do note that (amongst others) the adopted steady RANS approach and the selected turbulence model reduce the accuracy of the result due to its inability to accurately predict separation, recirculation and unsteady vortex shedding in the wake. To improve this, a more advance approach such as Large Eddy Simulation (LES) or a hybrid RANS/LES should be used, at a much higher computational cost.

Similarly good results have been presented very recently by Prpić-Oršić *et al.* (2020), where also, the wind tunnel measurements of Andersen (2013) are used for validation of a computational setup using StarCCM+. In this publication, the authors also utilise a steady RANS solver where wall functions are used to resolve the boundary layers around the ship. The computational domain matches the wind tunnel geometry and hence, requires corrections for the blockage. For the same configuration as used by Janssen *et al.* (2017) (01-01-02 with full container stack), the corrected CFD results show an equally good match with the experimental results. Given that Prpić-Oršić *et al.* (2020) does not give details about the blockage corrections applied, it is assumed to be the same method as used by Janssen *et al.* (2017). Hence, the analysis in § A6.4 is not an isolated case.

In 2018, Saydam and Taylan have used the experimental data of Andersen (2013) for validation of their numerical setup to compute the wind loads on various vessel types with the commercial CFD package ANSYS CFX v14. Details on the validation are not given, and for five relative wind angles, a comparison is made. The conclusion of Saydam and Taylan is that their CFD setup is applicable to the problem of investigating the wind pressure on different ships [sic] types.

Majidian and Azarsina (2018) investigated the effect of different container stacking configurations on the wind resistance of a Post-Panamax 9000 TEU container ship in head wind. The computational results are compared with the experimental data of Andersen (2013). The draft of the ship model as used in these computations is different (and higher) than the draft of the model in the wind tunnel tests. The computational domain has a similar size as the wind tunnel tests but the blockage correction that was applied by Andersen was not used in the research of Majidian and Azarsina (2018). For the different configurations, a comparison with the experimental results shows that the numerical results underpredict the drag force as compared to the experiments by $10\,\%$ to $30\,\%$. Due to the different draft and the lack of blockage correction, it is difficult to judge the validity of the comparison as presented by Majidian and Azarsina (2018).

A6.7 Conclusions

Some conclusions can be drawn from this analysis:

- not many publications that use the data of Andersen (2013) include enough information in the publication to replicate the results (the publication of Janssen *et al.* (2017) is one notable exception);
- the use of steady RANS CFD computations for the computation of wind loads on ships can give significant errors when compared to experimental results obtained in wind tunnels despite the good agreements as reported in literature;
- to better mimic the physics of the experiments, one could or should resort to unsteady (second-order) time integration schemes, possibly combined with Detached Eddy Simulation (DES) or LES methods to resolve instead of model the massive turbulent flow in the wake of the ship's superstructure;
- Engineering Sciences Data Unit (1998) does not contain wake blockage corrections that are directly applicable to objects with dimensional proportions typical for ships. Using numerical (CFD) methods to derive blockage correction factors similar to the methods presented in Engineering Sciences Data Unit (1998) is left as a recommendation for future work.

A7 Simulation of an atmospheric boundary layer with FINE/Marine

A7.1 Introduction

In this appendix, research is documented related to the simulation of atmospheric boundary layers in FINE/Marine as a means to determine aerodynamic forces on a hull. Such results can be used as a reference case for computations executed with Gerris in uniform flow, when the hull is partitioned in sections to approximately account for an atmospheric boundary layer (as discussed in § 1.5.5). In the past, computations have been executed with FINE/Marine to determine new wind coefficients for the *Tripoli* estuary vessel at a prescribed draft for a certain container configuration (Van Hoydonck *et al.*, 2015b). One of the conclusions was that the new coefficients were an improvement over the original coefficients, but that the computations required too much engineering time. This appendix documents some findings related to the simulation of atmospheric boundary layers in RANS computations.

A7.2 Atmospheric boundary layer

For RANS computations of a neutral atmospheric boundary, three inputs are required at the inlet of the computational domain. These are a velocity profile $U(z)$, a turbulent kinetic energy profile $k(z)$ and a turbulent dissipation profile $\varepsilon(z)$. All of these profiles are a function of the vertical position above the ground (or water surface) z.

For the velocity profile, both logarithmic and exponential profiles are used in literature (Andersen, 2013; Blocken *et al.*, 2007). For a neutral atmospheric boundary layer, the profile is logarithmic:

$$U(z) = \frac{u^*_{ABL}}{\kappa} \ln\left(\frac{z + z_0}{z_0}\right), \tag{69}$$

where, u^*_{ABL} is the boundary layer friction velocity, κ is the von Karman constant ($\approx\ 0.41$) and z_0 is the roughness length. The associated turbulent kinetic energy profile is constant with altitude,

$$k(z) = \frac{{u^*}^2_{ABL}}{\sqrt{C_\mu}}, \tag{70}$$

where C_μ is a constant from the $k - \epsilon$ turbulence model with a standard value of 0.09. Lastly, the turbulent dissipation is given by

$$\varepsilon(z) = \frac{{u^*}^3_{ABL}}{\kappa(z + z_0)}. \tag{71}$$

To ensure that the only cause of modifications to the profile are due to the presence of an object in the computational domain, a number of 2D computations without objects have been executed in a domain long enough for the flow to stabilise before it reaches the outlet. If there are changes in the flow field along the domain, these can be attributed to the solver and its boundary conditions (and possibly to the grid). The setup and results of these computations are presented in the next two sections.

A7.3 Simulation setup

The computations are executed with FINE/Marine version 7.1 in steady mode (first-order time integration).

A side view of the domain is shown in Fig. 88. The height of the domain is 500 m, the length is $30\,000$ m and the width is 10 m. Initially, the domain length was only 5000 m, but this length was not enough for the flow to converge by the time the outlet was reached.

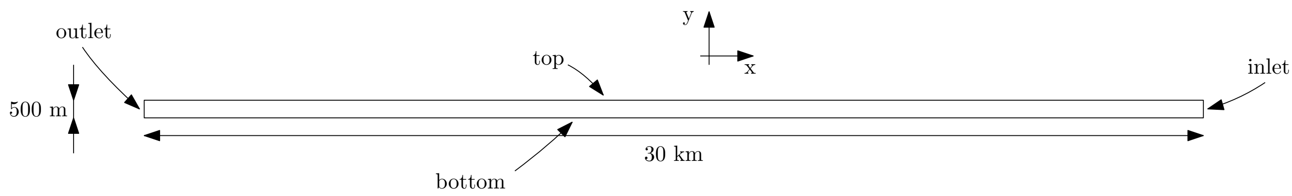


Figure 88 – Side view of the computational domain with boundary patch names.

A7.3.1 Boundary conditions

For the inlet, the *far field* boundary condition is used. This allows one to set the velocity, turbulent kinetic energy and turbulent dissipation profiles. For the outlet, both the *prescribed pressure* and *zero pressure gradient* boundary conditions have been tested. There does not appear to be a significant difference in the solution between the two, so the *prescribed pressure* boundary condition has been used for the computations. At the top boundary, the far-field boundary condition is used where the uppermost values of the profiles for the velocity, turbulent kinetic energy and turbulent dissipation rate applied at the inlet are prescribed. This ensures that no gradients can develop along the top boundary. For the bottom, a solid wall is used with wall functions. Computations have been executed with and without roughness activated for the bottom wall. *Mirror* conditions are applied at the front and back of the domain.

A7.3.2 Grid

Initially, the domain length was fixed at 5000 m. The grid generated for this domain was extruded to a length of $30\,000$ m. This feature has recently been added to HEXPRESS which significantly increases the workflow efficiency. The settings shown here pertain to the domain length of $30\,000$ m.

The initial Cartesian grid consists of 1500 cells: 300 cells with length 100 m in the x-direction, 5 cells vertically (y-direction) and one cell in the z-direction.

For a simulation of an atmospheric boundary layer, the cell height is refined in regions where large gradients in the inlet profiles occur. These are situated near the domain bottom. The maximum number of refinements is set to 9 and the refinement diffusion is increased to 5. The grid is refined by activating the bottom boundary for surface refinement. The target cell size is set to 110 m and 0.04 m for the x- and y-direction, respectively. Under the *Advanced* settings, the maximum aspect ratio is increased to 600. None of the boundaries are used for trimming. Wall functions are added at the bottom boundary to ensure that the $Y+$ is in the correct range (above 30 and below 300). Since the computation is based on a Reynolds number (with a reference length and reference velocity) and no reference length is available, this step was an iterative process. The final reference length was set to 400 m and the velocity was set to $10\,\mathrm{m/s}$[34]. With a first layer thickness of $9.484\,15 \times 10^{-3}$ m, the $Y+$ is in the correct range (between 75 and 130). With this value, 9 layers are added at the bottom. The

[34] At the time of writing this memo, it was found that the value for the kinematic viscosity was not changed from its default value of water to that of air.

total cell count for this 2D domain is $19\,800$. An overview of the vertical distribution of the cells near the inlet is shown in Fig. 89.

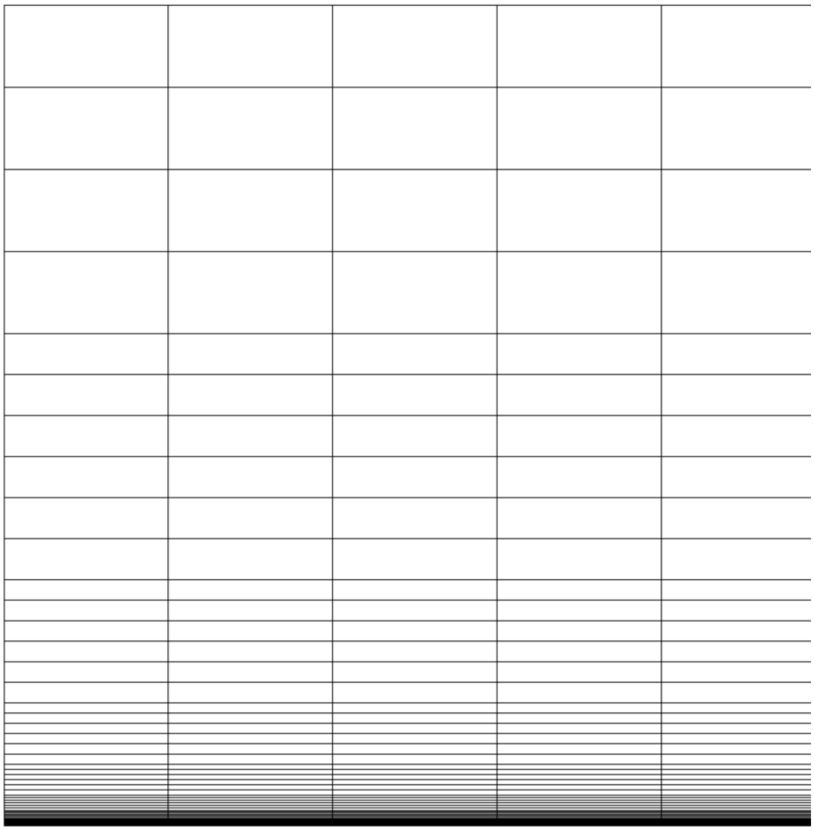


Figure 89 – Vertical distribution of cells in the domain.

A7.3.3 Computational setup

The configuration of these computations is fairly straightforward. The solver is run in *steady* mode with a single fluid with properties of air at 0 m ISA: density equals 1.225 kg/m^3 and the dynamic viscosity is set to 1.7894×10^{-5} Pas. The flow is assumed fully turbulent, and the $k\omega$-SST turbulence model is used. The reference length is set to 400 m and the reference speed equals 10 m/s.

A7.3.4 Boundary conditions

As already stated, a far field boundary condition is used at the inlet and top boundaries. For the top, values for the velocity, turbulent kinetic energy and turbulent dissipation are used that correspond to the uppermost values of the three profiles defined at the inlet. The values are: -18.4554 m/s, $2.630\,75$ m^2/s^2 and $0.003\,419\,49$ m^2/s^3 for the velocity[35], turbulent kinetic energy and turbulent dissipation. At the outlet, the pressure is fixed. The solution is initialised with a uniform velocity field of -10 m/s.

[35]The negative value for the velocity is a consequence of the orientation of the domain, air travels from the inlet to the outlet in the negative X-direction, see Fig. 88.

A7.3.5 Solver settings

Computations are run for $15\,000$ iterations and to ensure that the solver does not stop before this number, the convergence criterion is increased to 100 orders.

For the computations, the roughness is used as parameter, and varied between 0 (smooth) and 5, all values are listed in Table 10. As such, these values have no direct physical meaning: the roughness is solely varied to get an idea of its influence on the resulting velocity profile.

Table 10 – Roughness values used in the parameter variation.

roughness value
smooth
0.0001
0.001
0.01
0.1
0.5
1.0
5.0

At different locations along the domain length, the horizontal velocity component is recorded on a vertical line. The longitudinal positions are listed in Table 11. Some of these sections are special: the sections at 775 m and 1175 m are approximately located at the bow and the stern of three-dimensional computations executed with a ship geometry present.

Table 11 – Velocity section locations.

Distance from inlet, m
1.0
750.0
775
1175
$10\,000.0$
$20\,000.0$
$29\,999.0$

A7.3.6 Results

For the lowest 55 m in the domain, the profiles are shown in Fig. 90 and a zoom of these figures (for the lowest 10 m, is presented in Fig. 91.

Inspection of these figures shows that the inlet profile is not stable. This is not only a deficiency of FINE/Marine, it is a more general problem that is related to the implementation of roughness in wall functions (Blocken *et al.*, 2007). If a specific profile is used at an inlet, the actual profile encountered by a ship or structure downstream is different. Except for the computation with roughness height $k_s = 0.1$ m, all profiles converge to a single profile near the end of the domain. The computations have been re-executed with the EARSM turbulence model, and then, all velocity profiles converged, including the one with $k_s = 0.1$ m. For the higher values of the roughness length (which are more realistic for a (rough) sea surface), the converged profiles do not differ much.

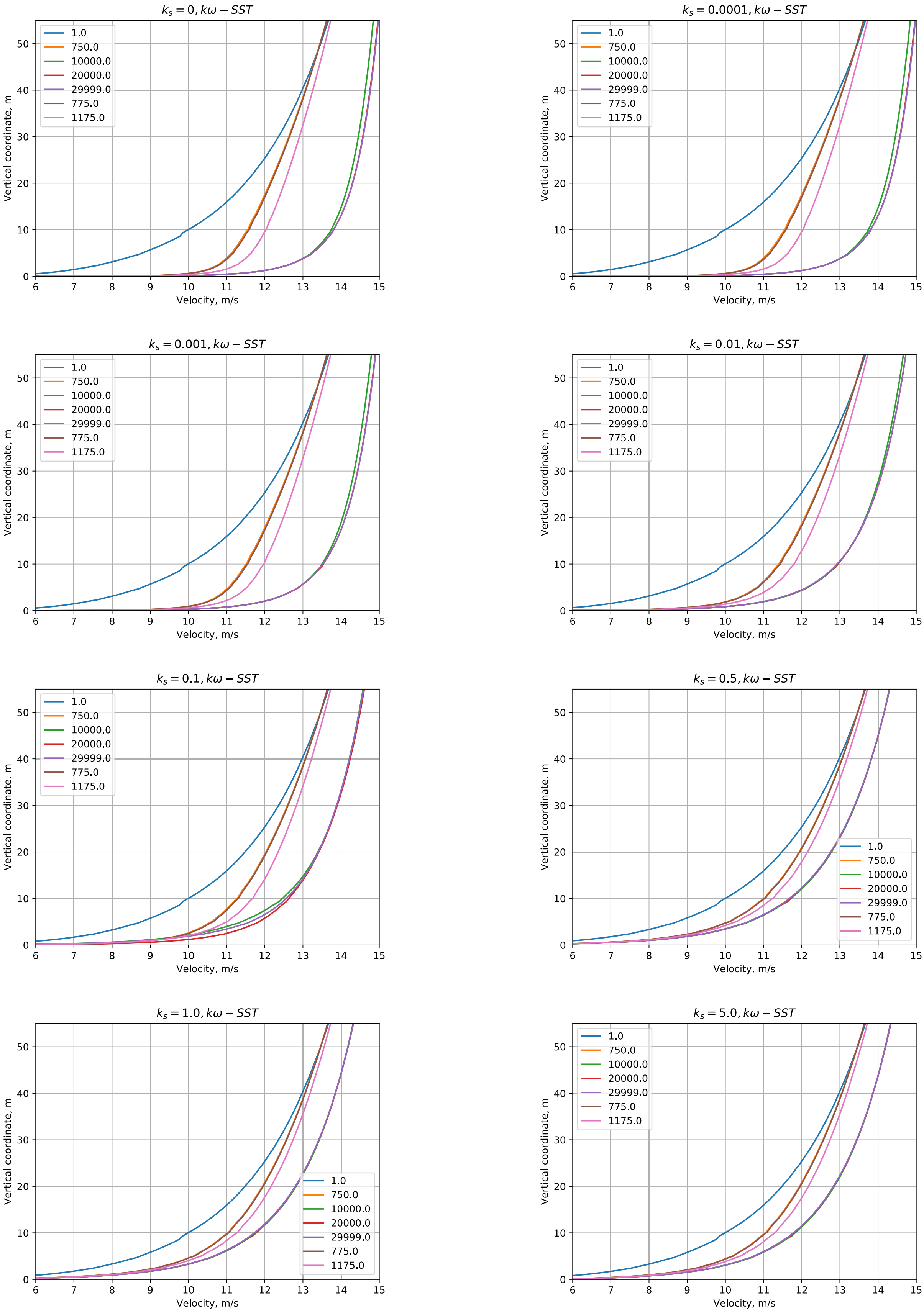


Figure 90 – Velocity profiles as function of longitudinal position in the domain below $z = 50$ m.

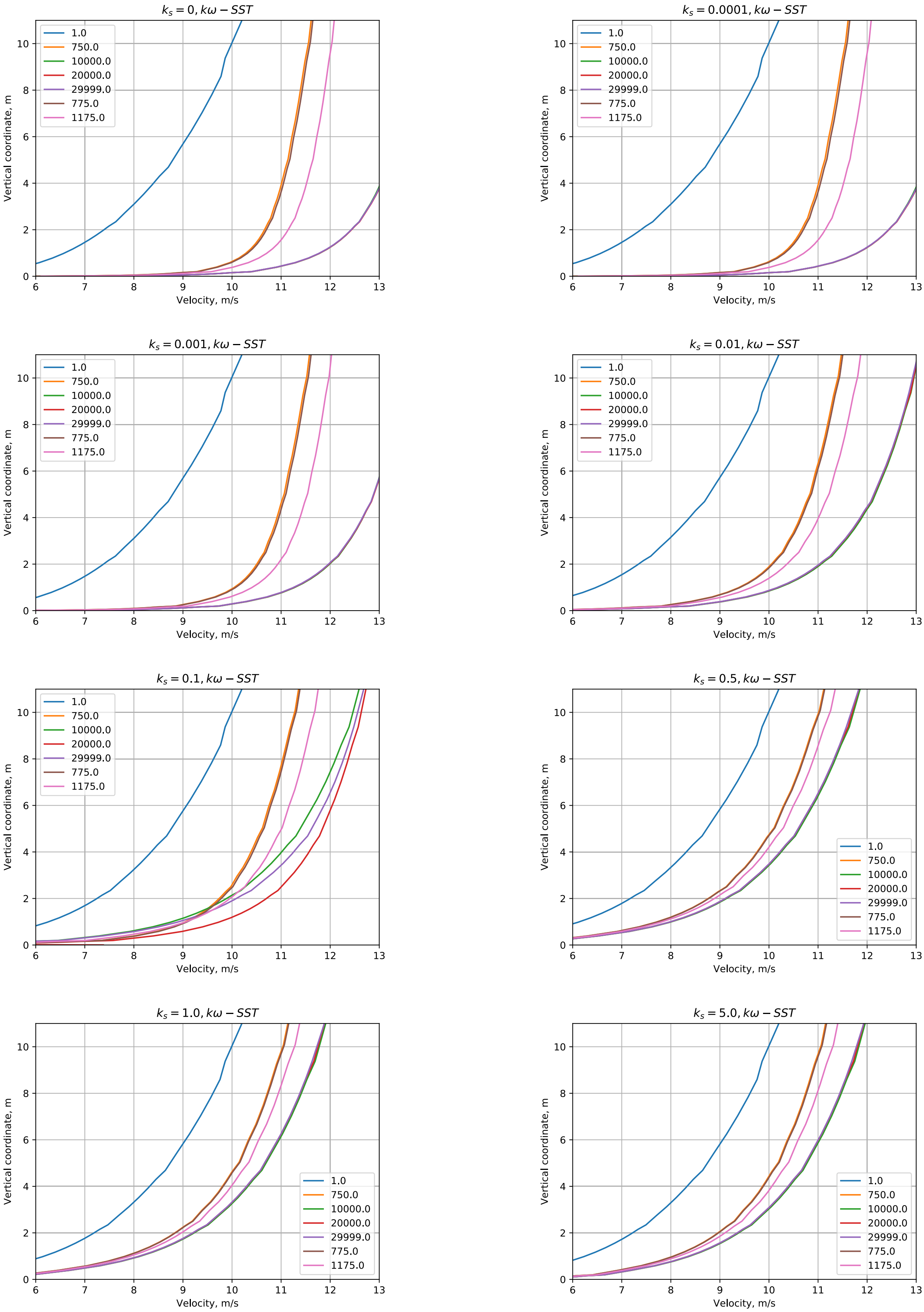
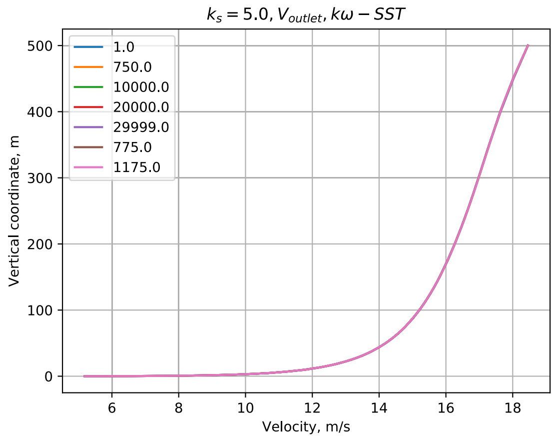


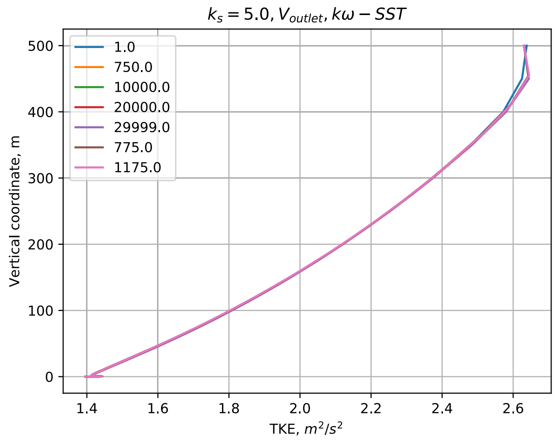
Figure 91 – Velocity profiles as function of longitudinal position in the domain below $z = 10$ m.

A7.3.7 Converged profiles

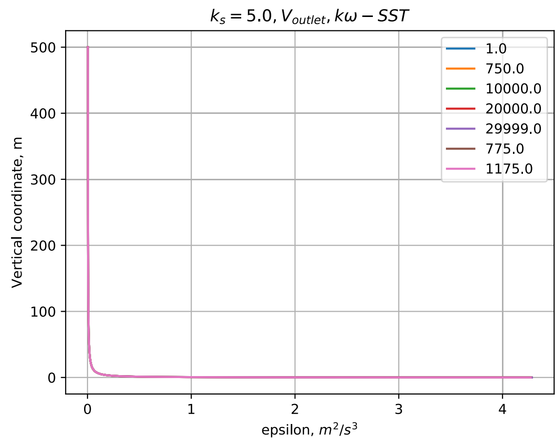
At the outlet, converged profiles are extracted for the three turbulence-related quantities for $k_s = 5.0$. These are used as input profiles for the inlet of an extra two-dimensional computation to verify that the profiles remain the same between the inlet and the outlet. For the three quantities, the profiles are shown in Fig. 92. Apart from a small deviation near the top of the domain for the turbulent kinetic energy profile, the profiles of the three quantities extracted from the domain at different longitudinal positions overlap. For this profile, the velocity at the standard reference height of 10 m is different: $V_{ref} = 11.8095$ m/s. At this height, the turbulent kinetic energy and turbulent dissipation equal $1.443\,95$ m^{2}s^2 and $0.065\,587$ m^2/s^3, respectively.



(a) Velocity



(b) Turbulent kinetic energy



(c) Turbulent dissipation

Figure 92 – Converged profiles for $k_s = 5.0$.

By means of the `lmfit` library[36] in Python, the converged inlet velocity profile is matched to an atmospheric profile. The original profile has three parameters: the reference height, the reference velocity at that height and the roughness length. For the current fit, the reference height and reference velocity have been fixed at the values that occur at 10 m height. The only parameter left is the roughness length z_0.

Fitting Eq. 69 to the experimental profile in a least-squares sense gives an optimal value for the roughness length $z_0 = 7.7403 \times 10^{-4}$ m. The numerical data, initial guess and final profile are shown in Fig. 93.

[36] See `https://lmfit.github.io/lmfit-py/`.

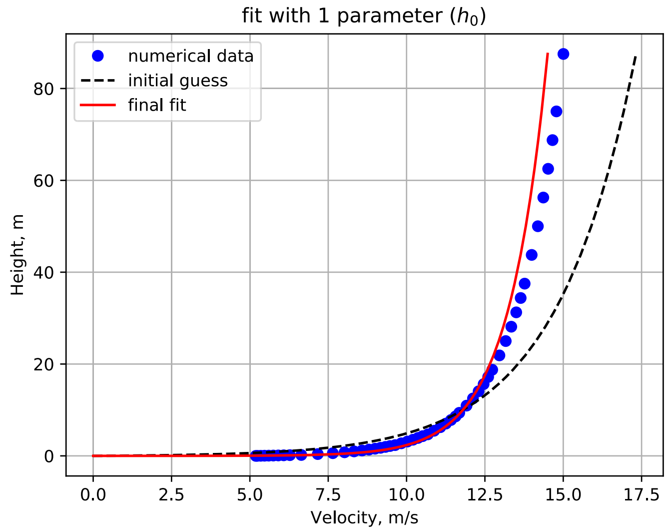


Figure 93 – Best fit of Eq. 69 to the converged velocity profile.

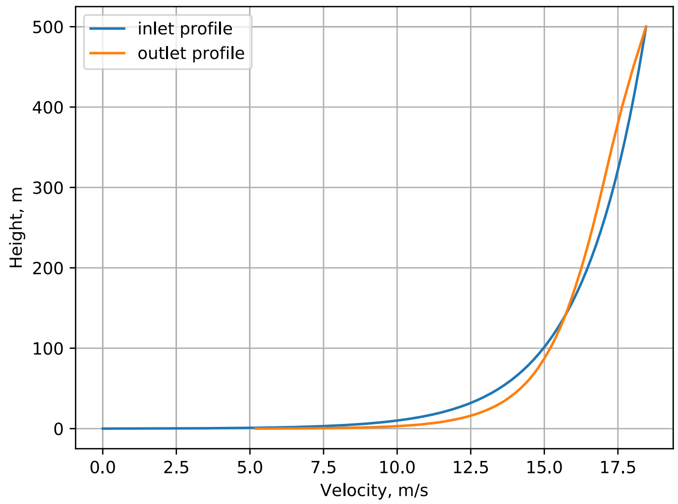


Figure 94 – Comparison of the velocity profiles at the inlet and outlet.

A7.4 Conclusions

The results of the computations reported here show that FINE/Marine lacks the ability to execute a simulation with an arbitrary atmospheric boundary layer profile specified at the inlet. The profiles for velocity, turbulent kinetic energy and turbulent dissipation assigned at the inlet transform to other profiles if the domain is long enough[37]. For the purpose of using FINE/Marine computations with an atmospheric boundary layer as reference for computations in Gerris executed in uniform flow, converged profiles obtained near the outlet of the domain should be used at the inlet of three-dimensional computations so that the influence of the profile on the forces experienced by an object can be compared to the forces experienced by this object when it is subjected to a uniform velocity field.

This also means that for computations executed in the past with FINE/Marine where an atmospheric profile

[37]This issues has been reported to Numeca.

was assigned at the inlet (such as Van Hoydonck *et al.*, 2015a,b, 2016), the actual profile at the ship is likely to be different.

References

Andersen, **I. M. V.** (2007). Vindkræfter på Containerskibe. (M.Sc. thesis). Institut for Mekanik, Energi og Konstruktion, Danmarks Tekniske Universitet

Andersen, **I. M. V.** (2013). Wind loads on post-panamax container ship. *Ocean Engineering 58*: 115–134. DOI: https://doi.org/10.1016/j.oceaneng.2012.10.008

Anderson, **J. D.** (1991). Fundamentals of aerodynamics. 2nd ed. McGraw-Hill Series in Aeronautical and Aerospace Engineering. McGraw-Hill: New York. ISBN: 0-07-100767-9

Armitt, **J.; Counihan**, **J.** (1968). The simulation of the atmospheric boundary layer in a wind tunnel. *Atmospheric Environment 2 (1)*: 49–62. DOI: 10.1016/0004-6981(68)90019-X

Barlow, **J.; W.H. Rae**, **J.; Pope**, **A.** (1999). Low-Speed Wind Tunnel Testing. 3rd ed. John Wiley & Sons: New York. ISBN: 978-0-471-55774-6

Blendermann, **W.** (2013). Practical ship and offshore structure aerodynamics. Hamburg University of Technology: Hamburg, Germany. ISBN: 978-3-89220-669-9

Blendermann, **W.** (1993). Schiffsform und Windlast-Korrelations- und Regressionsanalyse von Windkanalmessungen am Modell. 533. Hamburg, Germany

Blocken, **B.; Stathopoulos**, **T.; Carmeliet**, **J.** (2007). CFD simulation of the atmospheric boundary layer: wall function problems. *Atmospheric Environment 41 (2)*: 238–252. DOI: https://doi.org/10.1016/j.atmosenv.2006.08.019

Engineering Sciences Data Unit (1998). Blockage corrections for bluff bodies in confined flows. ESDU 80024. ISBN: 978-0-85679-307-3

Fujiwara, **T.; Nimura**, **T.** (2005). New estimation method of wind forces acting on ships on the basis of mathematical model. ***in****: Proceedings of the Fifteenth (2005) International Offshore and Polar Engineering Conference*. Seoul, Korea, 19th–24th June 2005. The International Society of Offshore and Polar Engineers. ISBN: 1880653648. pp. 82–89

Gould, **R.** (1970). Wake blockage corrections in a closed wind tunnel for one or two wall-mounted models subject to separated flow. *Aeronautical Research Council*, R. & M. No. 3649. Ministry of Technology: London

ITTC (1999). ITTC – Recommended Procedures and Guidelines - CFD, General CFD verification 7.5-03-01-04 (Revision 00). ITTC. p. 7

Janssen, **W. D.; Blocken**, **B.; van Wijhe**, **H. J.** (2017). CFD simulations of wind loads on a container ship: Validation and impact of geometrical simplifications. *Journal of Wind Engineering and Industrial Aerodynamics 166*: 106–116. DOI: https://doi.org/10.1016/j.jweia.2017.03.015

Majidian, **H.; Azarsina**, **F.** (2018). Aerodynamic simulation of a container ship to evaluate cargo configuration effect on frontal wave loads. *China Ocean Engineering 32*: 196–205. DOI: 10.1007/s13344-018-0021-1

Popinet, **S.** (2009). An accurate adaptive solver for surface-tension-driven interfacial flows. *Journal of Computational Physics 228 (16)*: 58380–5866. DOI: 10.1016/j.jcp.2009.04.042

Popinet, **S.** (2003). Gerris: a tree-based adaptive solver for the incompressible Euler equations in complex geometries. *Journal of Computational Physics 190 (2)*: 572–600. DOI: 10.1016/S0021-9991(03)00298-5

Prpić-Oršić, **J.; Valčić**, **M.; Čarija**, **Z.** (2020). A hybrid wind load estimation method for container ship based on computational fluid dynamics and neural networks. *Journal of Marine Science and Engineering 8 (7)*: 539. DOI: https://doi.org/10.3390/jmse8070539

Ricci, **A.; Janssen**, **W.; van Wijhe**, **H.; Blocken**, **B.** (2020). CFD simulation of wind forces on ships in ports: Case study for the Rotterdam Cruise Terminal. *Journal of Wind Engineering and Industrial Aerodynamics 205*: 104315. DOI: https://doi.org/10.1016/j.jweia.2020.104315

Richards, **P. J.; Norris**, **S. E.** (2012). Appropriate boundary conditions for a pressure driven boundary layer. ***in:*** Brandner, P. A.; Pearce, B. W. (Eds.). *Proceedings of the 18th Australasian Fluid Mechanics Conference*. Launceston, Australia, 3rd–7th December 2012. Australasian Fluid Mechanics Society. ISBN: 978-0-646-58373-0. p. 4

Saydam, **A. Z.; Taylan**, **M.** (2018). Evaluation of wind loads on ships by CFD analysis. *Ocean Engineering 158*: 54–63. DOI: https://doi.org/10.1016/j.oceaneng.2018.03.071

Ueno, **M.; Kitamura**, **F.; Sogihnara**, **N.; Fujiwara**, **T.** (2012). A simple method to estimate wind loads on ships. ***in****: The 2012 World Congress on Advances in Civil, Environmental, and Materials Research (ACEM' 12)*. Seoul, Korea, 26th August 2012–30th August 2005. International Association of Structural Engineering and Mechanics. pp. 2314–2322

Van Hoydonck, **W.; Delefortrie**, **G.; Peeters**, **P.; Mostaert**, **F.** (2015a). Computation of aerodynamic coefficients of ships using FINE/Marine: initial evaluation. 3.0. *WL Rapporten*, 12_106. Antwerp, Belgium

Van Hoydonck, **W.; Delefortrie**, **G.; Vos**, **S.; Peeters**, **P.; Mostaert**, **F.** (2015b). Computation of aerodynamic coefficients of ships using FINE/Marine: Wind coefficients for the Tripoli estuary container vessel at a draft of 3.5 m. 3.0. *WL Rapporten*, 12_106. Antwerp, Belgium

Van Hoydonck, **W.; Verwilligen**, **J.; López Castaño**, **S.; Mostaert**, **F.** (2021). Evaluation of Gerris flow solver for the computation of wind coefficients. Optimization of geometry preparation. 2.0. *FHR Reports*, 16_058. Antwerp, Belgium

Van Hoydonck, **W.; Vos**, **S.; Peeters**, **P.; Mostaert**, **F.** (2016). CFD computations of the wind climate behind a roro ship. Version 3.0. *FHR Reports*, 15_096. Flanders Hydraulics Research: Antwerp, Belgium

Wagner, **M.** (2005). Wind-Tunnel Tests with FSG NB 731/732. FORCE 2005031. FORCE Technology: Lyngby, Denmark

Walree, **F.; Willemsen**, **E.** (1988). Wind loads on offshore structures. ***in:*** Moan, T.; Janbu, N. *et al.* (Eds.). *Proceedings of the International Conference on Behaviour of Offshore Structures*. Trondheim, Norway, June 1988. Tapir Publishers

DEPARTMENT **MOBILITY & PUBLIC WORKS**
Flanders hydraulics Research

Berchemlei 115, 2140 Antwerp
T +32 (0)3 224 60 35
F +32 (0)3 224 60 36
waterbouwkundiglabo@vlaanderen.be
www.flandershydraulicsresearch.be