International Shipbuilding Progress 60 (2013) 107–141
DOI 10.3233/ISP-130083
IOS Press

Verification and validation for marine applications of CFD

L. Eça [a,*] and M. Hoekstra [b]

[a] *Instituto Superior Técnico, Lisbon, Portugal*
[b] *Maritime Research Institute, Wageningen, Netherlands*

This paper discusses Verification and Validation for CFD applications. It emphasizes the differences between the mathematical problem of "solving the equations right" (Verification) and the science/engineering activity of "solving the right equations" (Validation). A clear distinction is made between Code Verification (error evaluation) and Solution Verification (error estimation) and procedures based on grid refinement studies are presented and discussed for both activities. The paper presents examples of Code Verification for Reynolds-Averaged Navier–Stokes solvers using the Method of the Manufactured Solutions; Solution Verification exercises including the KVLCC2 tanker at model and full scale Reynolds number and two examples of the application of the ASME V&V 20 Validation procedure. This paper is written to stimulate a conscientious approach to CFD in marine applications.

Keywords: Verification, validation, CFD, grid refinement, uncertainty quantification

1. Introduction

Since the 1960s the role of Computational Fluid Dynamics (CFD) in marine hydrodynamics has been gaining importance continuously. Starting with inviscid flow models, we have passed from solving boundary-layer equations to RANS equations and in these days recourse is sometimes taken to Detached Eddy Simulation (DES) or Large Eddy Simulation (LES). This is reflected in the papers submitted to this series of conferences, confirming that CFD has undeniably conquered a solid position, next to experimental work.

Since none of the mathematical models mentioned lend themselves to analytical solution, numerical techniques are needed. Algorithms of various kind have been developed and improved significantly in the course of time. Important here is to recognize that in CFD there is next to a mathematical model (equations plus boundary conditions) always a numerical method to solve the model.

The attentive reader may see a parallel with experimental fluid dynamics (EFD). There the test facility and the (usually scaled) test object form the physical model, while instrumentation and test procedures are needed for data acquisition.

*Corresponding author. E-mail: eca@marine.ist.utl.pt.

In EFD, uncertainty quantification became a mandatory issue [17] with procedures acknowledging the distinction between the measuring instruments and the physical modelling [20]. But if an experimentalist is supposed to be concerned with the accuracy of his measurements, can a CFD practitioner stay behind and produce numbers without caring about quality and uncertainties?

The problems to be handled by CFD have a huge variety and cover a wide range of complexity. With the growing capabilities of hardware and numerical techniques the manageable complexity increases. Problems which were out of reach a couple of years ago can now be taken up. But, as observed by Roache [34], the trend is still that CFD practitioners (particularly in the academic world) focus on qualitative simulation of the next more difficult problem, more than aiming for quantitative accuracy of the previous, somewhat simpler problem. This paper is written to stimulate a conscientious approach.

There are good reasons to take quality of numerical simulations seriously. Some scientific journals have a publication policy saying that papers are accepted only if due consideration is given to numerical (or experimental) uncertainty [18,19]. Moreover, in commercial CFD applications some kind of quality assessment may be demanded. Additional arguments to pursue quality assessment come from the engineering world. After all, if decisions in design optimisation are partly or solely(!) based on CFD, one must have an idea of the uncertainty of the results in order not to run the risk of being misled.

The question is now: how can quality assessment for CFD be accomplished? The answer to this question can only be provided through "Verification and Validation", i.e. making a clear distinction between the deficiencies of the mathematical model (Validation) and the numerical errors of the techniques used to solve the selected model (Verification) (N.B. although the words verification and validation are practically synonyms, in the present context they are by agreement distinct technical concepts; in an attempt to avoid misunderstanding we indicate that these technical concepts are meant rather than the general "dictionary meaning" of the words by writing them with capitals, following Roache [34,36]). As elegantly described by Boehm [3] and Blottner [2], Verification checks if we are "solving the equations right", whereas Validation deals with "solving the right equations".

Verification is a purely mathematical/numerical exercise that aims at quantifying the numerical error/uncertainty (the distinction between both will be given below). Therefore, it does not require the knowledge of any experimental data, but it may require exact solutions of the selected mathematical model. On the other hand, Validation intends to determine how well the mathematical model represents the physical world; it requires measures of the physical world (experimental data) and of the mathematical model (CFD simulation) that naturally should correspond to the same set-up. Since both these quantities are not exact, a proper Validation exercise requires the knowledge of the experimental and numerical uncertainties. Therefore, Verification must precede Validation.

A CFD code (or any code that performs numerical flow simulation) contains a discretised version of a set of continuum partial differential equations which can be solved in a selected domain with suitable boundary conditions. With regard to Verification there are two different problems that must be addressed: the correctness of the implementation of the model in the code (no bugs); the acceptability of the estimated error/uncertainty of a given application. Verification consists therefore of two separate actions: Code Verification and Solution Verification. The first one requires error evaluation and so the exact solution must be known, whereas the latter aims at estimating the error/uncertainty of a numerical solution for which the exact solution is usually unknown. It is also important to point out that Code Verification is done once and it does not need to be repeated if the code remains unchanged. On the other hand, Solution Verification is an ongoing activity that should preferably be performed for every application of the code.

In order to illustrate the significant differences between Verification and Validation and the meaning of "validated", we present two examples:

(1) Suppose we try to compute the wake field in the propeller plane of a single-screw ship with a potential-flow code, i.e. the Laplace equation for a velocity potential is our mathematical model. The Verification process could then be completed with success by showing that the results converge with the correct order (usually second order) upon grid refinement and the uncertainty of the fine grid solution is marginal. The Validation exercise would show however that the potential-flow model is inadequate for the wake prediction. In short: very accurate results (correct numerical solution), but with serious modelling errors (wrong mathematical model). (This example can also serve as an illustration of the distinction between accuracy and reliability.)

(2) By coincidence (error cancelling) one might have a numerical solution with perfect agreement in a specific aspect of an experimental result, possibly obtained by calibration of one or more free parameters in the mathematical model (turbulence model, boundary condition, fluid properties, ...) or in the numerical solution (grid density, discretisation coefficients, ...). The proud conclusion could be that the results are validated. True as that conclusion may be, "validated" means here nothing else than that the exercise did not produce any evidence that the computational model used is unsuitable for the present flow problem. But it says nothing about the quality of the numerical result until in addition the uncertainty bands of both experiment and simulation have been found to be sufficiently small.

In the remaining sections of this paper, we present an overview of our experience in Code and Solution Verification from the last 13 years, which is primarily (but not exclusively) related with the Reynolds-averaged Navier–Stokes (RANS) equations as a mathematical flow model. Validation is also addressed, but in that case we will only discuss the procedure proposed in the ASME V&V20 report [1], which is not the only available in the open literature (see for example [37,38]).

2. Error classification

2.1. Error versus uncertainty

In the distinction between errors and uncertainties, we follow the classification given by Roache in [34]: an error is defined as the difference between a given solution and its "true/exact" value. It has a sign and it requires the knowledge of the "truth". As we will discuss below, the "truth" depends on which type of error we want to evaluate. An uncertainty defines an interval that should contain the "true/exact" value with a certain degree of confidence. It is defined with a $\pm$ sign, i.e. it signifies an error band for the numerical result so that the true solution can be expected to be within that error band (usually with 95% confidence). In the present paper we will estimate uncertainties as the error estimator multiplied by a factor of safety [34].

2.2. Modelling errors

Physical modelling errors are a consequence of the representation of the physical world by a mathematical model. Their detection and quantification is the goal of Validation.

Physical modelling errors are not only related to the set of equations chosen but also to the selected domain and to the boundary conditions. Calculations are carried out in a domain of finite size and exact boundary conditions are not always available. For instance, if "far-field" boundary conditions are applied while the computation domain is so small that at the borders the "far-field" has not been reached a modelling error in the boundary conditions appears.

2.3. Programming errors

The discretisation of the selected set of partial differential equations and the procedure to solve the resulting equations numerically is incorporated into a computer code. Programming errors (bugs), often simple typos in the code, are likely to appear. Code Verification aims at detecting and removing such type of errors.

2.4. Numerical errors

It is commonly accepted (see for example [34] or [32]) that there are three different contributions to the numerical error:

- Round-off errors as a consequence of the finite precision of the computers.
- Iterative errors that stem from the non-linearity of the mathematical equations solved by CFD.
- Discretisation errors which are a consequence of the approximations made (finite-differences, finite-volume, finite-elements, ...) to transform the partial differential equations of the continuum formulation into a system of algebraic equations.

2.4.1. Round-off errors

The reduction of the round-off error is related to the number of digits available and to the numerical procedure applied to obtain the solution. For smooth solutions, the use of double-precision is usually sufficient to obtain a negligible contribution of the round-off error to the numerical error. However, there are ill-conditioned problems where the round-off error may be the leading contribution to the numerical error. A simple and relevant example for CFD is high-order three-dimensional polynomial interpolation in grids with high aspect ratio cells, typical of viscous flow solutions. The determination of the coefficients of a polynomial of order n leads to a matrix with diagonal coefficients ranging from 1 to $(\Delta x)^n \times (\Delta y)^n \times (\Delta z)^n$. High aspect-ratio cells combined with large values of n lead to severe round-off error problems.

It is interesting to observe that the "truth" for the determination of the round-off error is never available because it would require a machine with infinite precision. However, a simple comparison between solutions obtained with different precision (simple and double or double and quadruple) is usually sufficient to assess the influence of the round-off error.

Obviously, even in smooth solutions, there is a limit to the amount of grid refinement imposed by the round-off error.

2.4.2. Iterative errors

The iterative error is related to the non-linearity of the system of partial differential equations solved in CFD. However, in a CFD code for the solution of the RANS equations there are several sources of non-linearity:

- The convective terms. The usual linearisation procedures are Picard or Newton methods, see for example [21], which imply an iterative solution.
- Deferred corrections in the discretisation schemes of the continuity and momentum equations. The velocity derivatives of the convective terms are often discretised with high-order schemes, but with implicit first-order approximations and explicit higher order contributions.
- The turbulence closure. For example, one and two-equation eddy-viscosity models have non-linear convective terms and non-linear production and dissipation terms. Also, the turbulence model equations are often solved segregated from the continuity and momentum equations.

Furthermore, the linear system of algebraic equations obtained from the discretisation of the linearised partial differential equations is rarely solved with a direct method. Therefore, the solution process includes an extra iterative cycle originating from the way in which the linear systems of equations is solved. In most flow solvers, no clear distinction is made between the various iterative cycles.

In principle, the iterative error may be decreased as far as the machine accuracy (round-off error) permits. However, in complex turbulent flows it is not guaranteed that one achieves that level of convergence. Furthermore, the c.p.u. time required to attain such level of iterative error may be significantly higher than that required for

achieving an "acceptable" level of the iterative error. For this reason it is unusual in industrial applications of CFD to converge to machine accuracy, which brings up the need to develop reliable techniques for estimating the iterative error.

In [6,9], we have shown that the iterative error is usually much larger than any norm of the normalised residuals or changes in the last iteration performed. Furthermore, the iterative error should be two to three orders of magnitude smaller than the discretisation error to have a negligible influence on the estimation of the numerical error. This means that the contribution of the iterative error to the numerical error is often incorrectly ignored or underestimated.

It is also important to point out that the determination of the iterative error of a given numerical solution does not require the knowledge of the exact solution. The "truth" for this type of numerical error can be approximately determined (to the level of the round-off error) by converging the numerical solution to machine accuracy.

2.4.3. Discretisation errors

The discretisation error is usually the main contributor to the numerical error of a CFD solution. Its determination requires the knowledge of the exact solution and so in many practical applications we can only estimate it.

Although several techniques have been proposed in the open literature (see for example [34]) to make such estimate, we shall here focus on what is currently the most general and reliable technique: the use of systematically refined grids.

Unlike the other two sources of numerical error, the relative importance of the discretisation error decreases with the grid refinement.

3. Discretisation error representation by power series expansions

One the most common approaches to describe the behaviour of the discretisation error with grid refinement studies is the use of power series expansions [34]. The basic equation to estimate the discretisation error ϵ_ϕ of a variable ϕ is:

$$\epsilon_\phi \simeq \delta_{RE} = \phi_i - \phi_o = \alpha h_i^p, \tag{1}$$

ϕ_i stands for any integral or other functional of a local flow quantity, ϕ_o is the estimate of the exact solution, α is a constant to be determined, h_i is the typical cell size of grid i and p is the observed order of accuracy.

Equation (1) can actually be used for the description of the error of any flow quantity, $e(\phi) = \phi - \phi_{\text{exact}}$, as a function of the typical cell size. In that case, $e(\phi_o)$ should be equal to zero and the observed order of accuracy should match the theoretical order of the discretisation. Furthermore, if the error for cell size zero is negligible, the variation of the discretisation error with the typical cell size in logarithmic scales is a straight line.

$$\log\big(e(\phi)\big) \simeq p\log(h_i) + \log(\alpha). \tag{2}$$

Therefore, Eq. (1) can actually be used for Code Verification (error evaluation) and Solution Verification (error estimation). However, the goals of the two exercises are different: showing that $e(\phi_o) \simeq 0$ (i.e. of the order of the round-off error) and (if possible) that p is equal to the theoretical order of the method for Code Verification and obtaining δ_{RE} for Solution Verification.

The use of power series expansions to estimate discretisation errors relies on two simple requirements: the data are obtained in sufficiently refined grids to guarantee that the first term of the power series expansion is dominant, i.e. the data are in the "asymptotic range"; the grid refinement ratio is constant for the complete computational domain, in order to allow a single parameter (the typical cell size) to be defined to characterize the level of grid refinement.

We address the latter requirement next and will come back to the first one in practical examples of Solution Verification.

3.1. Definition of the typical cell size

The typical cell size h_i is a length measure, say the edge length of a representative grid cell. In a set of geometrically similar grids the ratio of the typical cell sizes of two grids of the set, h_i/h_j, is then uniform over the complete computational domain [39].

In order to clarify this concept, we consider a simple one-dimensional example where the "cell size" is just the distance between grids nodes. A geometrically similar grid set is obtained when the non-dimensional distance between grid nodes

$$s = \frac{x - x_{\min}}{x_{\max} - x_{\min}}$$

is uniquely defined as a function of a parametric coordinate defined by

$$\xi = \frac{i-1}{N_x - 1}, \tag{3}$$

where i is the node counter and N_x is the number of grid nodes. This means that all grids of a given set must respect the same function $s = f(\xi)$. As discussed in [39], this definition of geometrically similar grids can be easily extended to two and three dimensions by generalizing the one to one invertible mapping between the physical domain (x) and the parametric space (ξ) to two and three dimensions.

In geometrically similar grids, the grid refinement ratio (h_i/h_1) can be easily defined by

$$\frac{h_i}{h_1} = \left(\frac{(N_{\text{cells}})_1}{(N_{\text{cells}})_i}\right)^{1/n} = \left(\frac{(\Lambda_1)_i}{(\Lambda_1)_1}\right)^{1/n} = \left(\frac{(\Lambda_2)_i}{(\Lambda_2)_1}\right)^{1/n}, \tag{4}$$

where n is the space dimension (1, 2 or 3) and Λ_1 and Λ_2 are the average and root mean square of the cell size Λ (length, area or volume)

$$\Lambda_1 = \frac{\sum_{i=1}^{N_{\text{cells}}} \Lambda_i}{N_{\text{cells}}} \quad \text{and} \quad \Lambda_2 = \sqrt{\frac{\sum_{i=1}^{N_{\text{cells}}} \Lambda_i^2}{N_{\text{cells}}}}.$$

3.2. A simple 1D example

Geometrical similarity is achievable in structured grids, but seems hard to reconcile with the concept of unstructured grids. Therefore, it is important to know what are the consequences of using any of the definitions presented above in a grid set that does not respect geometrical similarity. To this end, we present the solution of the simple differential equation

$$\frac{\mathrm{d}^2\phi}{\mathrm{d}x^2} = a\phi \quad \text{for } 0 < x < 1 \tag{5}$$

with the boundary conditions

$$\begin{aligned} &\phi(0) = 1, \\ &\phi(1) = 0. \end{aligned} \tag{6}$$

The exact solution of the problem is given by

$$\phi_{\text{exact}} = C_1 \mathrm{e}^{-\sqrt{a}x} + C_2 \mathrm{e}^{\sqrt{a}x}, \tag{7}$$

where

$$\begin{aligned} C_1 &= -\frac{\mathrm{e}^{\sqrt{a}}}{\mathrm{e}^{-\sqrt{a}} - \mathrm{e}^{\sqrt{a}}}, \\ C_2 &= \frac{\mathrm{e}^{-\sqrt{a}}}{\mathrm{e}^{-\sqrt{a}} - \mathrm{e}^{\sqrt{a}}}. \end{aligned} \tag{8}$$

In this example we will use $a = 5$.

Equation (5) is discretised with second-order central-difference approximations and the resulting linear system of equations is solved in double-precision with a direct method. Therefore, this simple example has no iterative errors and the round-off error is more than six orders of magnitude below the discretisation error.

Figure 1 presents the four stretching functions tested: equally-spaced (Equal), cosine distribution (Cos), two-sided stretching function proposed in [43] (Stret) and distribution based on a geometrical progression for a fixed number of grid nodes

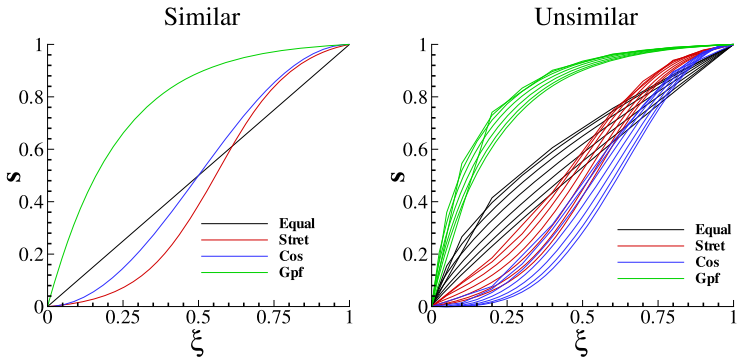

Fig. 1. Stretching functions of grid sets for one-dimensional example. (Colors are visible in the online version of the article; http://dx.doi.org/10.3233/ISP-130083.)

(Gpf). For each distribution, we have generated a set of geometrically similar grids (left plot) and a set of unsimilar grids (right plot) by disturbing the function $s(\xi)$. All sets have a coarsest grid of 5 elements and a finest grid of 320 elements.

The convergence of the L_∞, L_1 and L_2 norms of the discretisation error are plotted in Fig. 2 as a function of the grid refinement ratio h_i/h_1 (defined from Λ_2). The lines plotted in Fig. 2 were obtained with least squares fits to the data of the six finest grids [4], which in this simple example are clearly in the "asymptotic range". The results show that geometrically similar grids are required to obtain the correct order of accuracy. However, all grids sets make a correct estimation of the error (the plots of figure contain only straight lines). Therefore, strict geometrical similarity is essential for obtaining the correct order of accuracy (second goal of Code Verification), but not for estimating the error (goal of Solution Verification).

4. Code verification

Code Verification is a fundamental step in the development of any CFD code. Its goal is to ensure that a given flow solver is correctly coded (no bugs). A typical way to demonstrate code correctness is by showing that in a series of systematically refined grids the discretisation error goes to zero. Furthermore, if the "asymptotic" order of accuracy can be correctly assessed, it should match the theoretical order of the method. This requires analytical solutions to allow the direct evaluation of the discretisation error.

There are no analytical, realistic-flow solutions available for the RANS equations. Fortunately, this does not prohibit Code Verification of RANS solvers. The Method of Manufactured Solutions (MMS) [24,33] provides an excellent framework for Code Verification of any numerical method. In the MMS, a continuum solution is first constructed, i.e. one specifies all unknowns (including any turbulence quantity

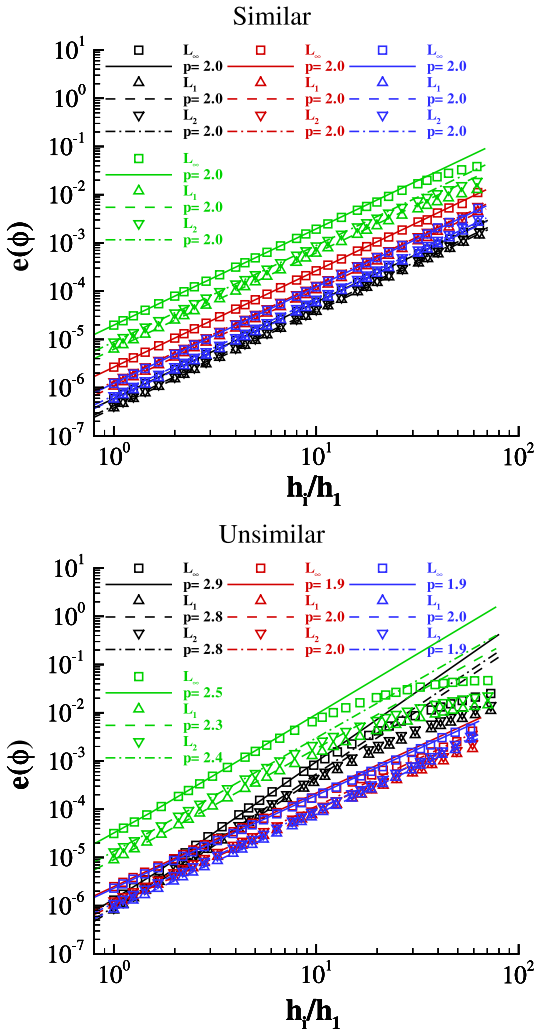

Fig. 2. Norms of the discretisation error as a function of the grid refinement ratio for one-dimensional example. (Colors are visible in the online version of the article; http://dx.doi.org/10.3233/ISP-130083.)

present in the turbulence model) by mathematical functions. In general, this constructed solution will not satisfy the governing equations (continuity, momentum and transport equations for turbulence quantities) because of the arbitrary nature of the choice. But by adding an appropriate source term, which removes any imbalance caused by the choice of the continuum solution, the governing equations are forced to become a model for the constructed solution.

In the MMS, the constructed solution need not have a physical meaning, since Code Verification is a purely mathematical exercise. Essential is that all terms in the

equations are activated (which in the end may require more than one MS). But we have found that it is very hard to do Code Verification for a RANS method with the turbulence model included in the trial if the MS has no resemblance with a near-wall flow at all [13,14]. A turbulence model with all its embedded empiricism may react in an awkward manner, so as to produce negative values for supposedly positive-definite turbulence quantities (like the turbulence kinetic energy), to give activation of limiters or damping functions where it is not desired, etc. All with catastrophic effects on the Verification exercise.

Recently, we have developed a new set of MS's for RANS solvers based on eddy-viscosity models that mimics the near-wall behaviour of the turbulence quantities [16]. The two main features of these MS's is the "skin friction" coefficient at the "wall" that follows a simple empirical formula for a flat plate turbulent boundary-layer and the use of expressions available from "automatic wall functions" [31] combined with an exponential decay in the outer region of the flow to determine the turbulence quantities. The proposed MS's have two and three-dimensional versions for incompressible, statistically steady flows that combine a smooth "boundary-layer" type flow with a perturbation flow that does not affect the shear-stress at the wall for a Reynolds number *Rn* that can range from 10^7 to 10^9.

Figure 3 presents three two-dimensional examples of the MS's proposed in [16], where MS1 corresponds to the "boundary-layer" type flow. Although the horizontal mean velocity profiles due not follow strictly the expected behaviour for a turbulence flow, as illustrated in Fig. 4, there is a viscous sub-layer and there is some resemblance with the log-law. Figure 5 presents the typical turbulence kinetic energy k, turbulence frequency ω and eddy-viscosity ν_t manufactured profiles for MS1. An alternative MS (MS1A) based on the approach followed in [13] is also plotted in Fig. 5. Although both MS's exhibit ω tending to infinity at the bottom (as in a real turbulent flow), in the "near-wall" region ν_t is much smaller in MS1A than in MS1. The profiles depicted in figures 4 and 5 use wall coordinates, i.e. $u^+ = u_x/u_\tau$, $\nu_t^+ = \nu_t/\nu$, $k^+ = k/u_\tau^2$ and $\omega^+ = \omega\nu/u_\tau^2$ as a function of $y^+ = u_\tau y/\nu$, where y is the distance to the wall, ν is the kinematic viscosity of the fluid and u_τ is the friction velocity.

Figure 6 illustrates a typical Code Verification result obtained with two completely different codes: PARNASSOS [22,41] and ReFRESCO [42]. PARNASSOS uses a finite-difference discretisation of the RANS and continuity equations written in Contravariant form and a coupled solution procedure, i.e. continuity (in its original form) and momentum equations are solved simultaneously. ReFRESCO uses a face-based finite volume discretisation of the RANS equations (volumes of arbitrary shape) written in strong conservation (divergence) form. The solution procedure is segregated and the continuity is enforced indirectly with a SIMPLE based pressure-correction method.

The calculations were performed in sets of 21 geometrically similar Cartesian grids with one-sided stretching functions [43] applied in the vertical direction to cluster grid nodes close to the bottom (the values of y^+ at the first grid node away

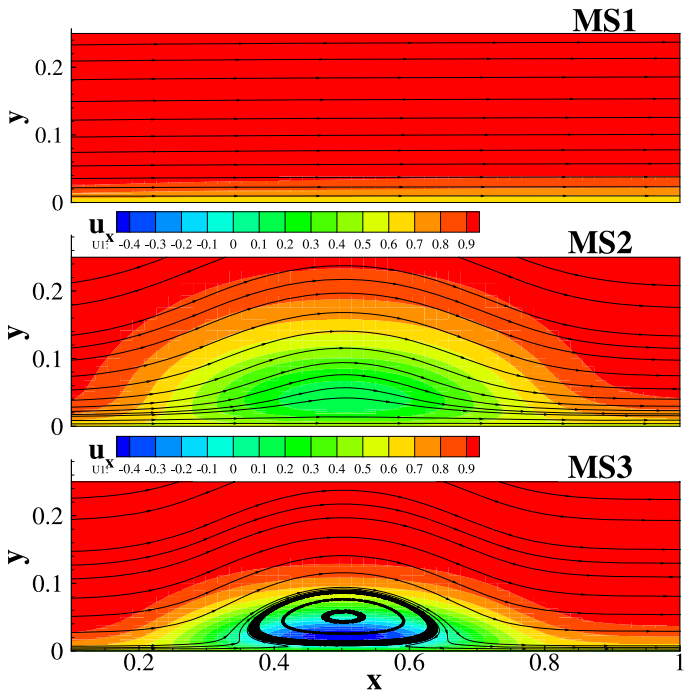

Fig. 3. Examples of two-dimensional manufactured solutions for incompressible flows. $Rn = 10^7$. (Colors are visible in the online version of the article; http://dx.doi.org/10.3233/ISP-130083.)

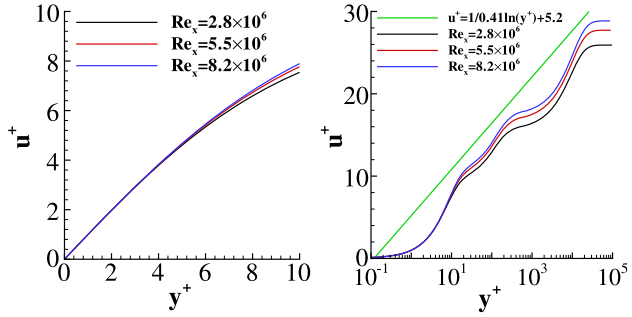

Fig. 4. Horizontal mean velocity profiles in wall coordinates for MS1 at three different locations. $Rn = 10^7$. (Colors are visible in the online version of the article; http://dx.doi.org/10.3233/ISP-130083.)

from the wall y_2^+ are given in Fig. 7, set A). The coarsest grid has 50×50 cells and the finest 800×800 cells, covering a grid refinement ratio of 16.

The data plotted in Fig. 6 were obtained with negligible round-off and iterative errors (at least four orders of magnitude below the discretisation error) and with the

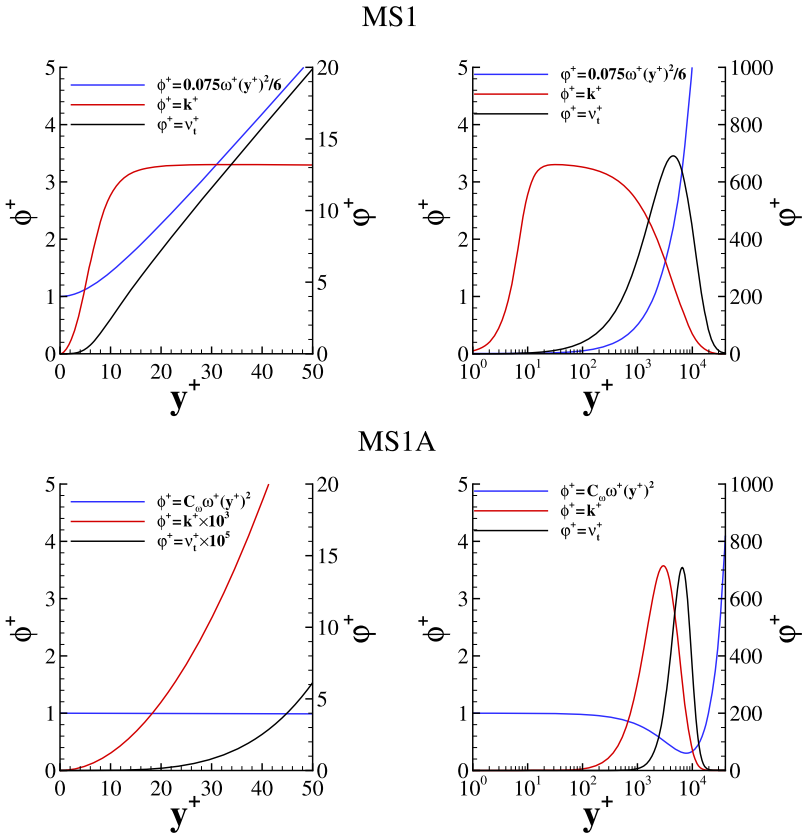

Fig. 5. Typical eddy-viscosity ν_t, turbulence kinetic energy k and turbulence frequency ω profiles in wall coordinates for MS1 and MS1A. $Rn = 10^7$. (Colors are visible in the online version of the article; http://dx.doi.org/10.3233/ISP-130083.)

manufactured eddy-viscosity field acting as the turbulence model. The L_∞ and L_2 error norms of the horizontal velocity component u_x show the expected behaviour with a negligible extrapolated error to cell size zero (straight lines in the plots) and an observed order of accuracy determined from the data obtained in the six finest grids matching the theoretical order of the two codes.

The MMS framework also allows several important exercises to assess the convergence properties of RANS solvers. As an example, Fig. 7 presents the stretching functions applied to the vertical grid lines of six different grid sets that contain exactly the same number of grid nodes. The maximum and minimum values of y_2^+ are also plotted in Fig. 7. Sets B and C have larger values of y_2^+ than the other four sets that exhibit $y_2^+ < 1$ for all grids with $h_i/h_1 < 5$.

The L_∞ and L_2 error norms of the horizontal velocity component u_x obtained in these six grid sets are plotted in Fig. 8 as a function of the grid refinement ratio.

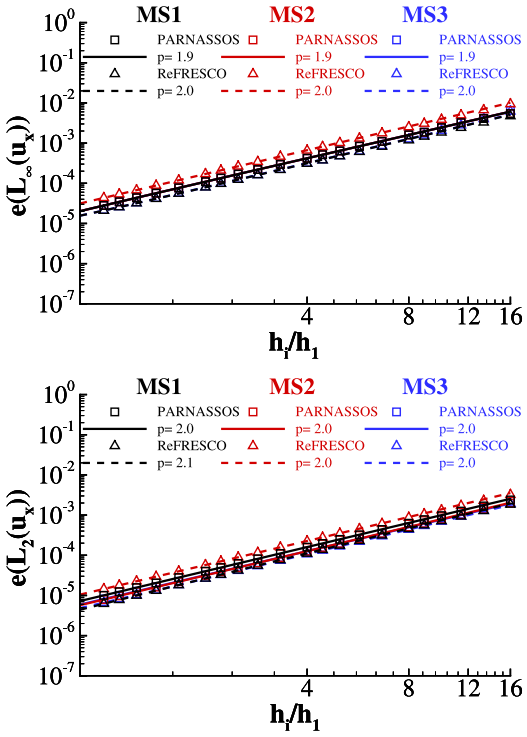

Fig. 6. L_∞ and L_2 norms of the error of the horizontal velocity component u_x calculated with PARNASSOS and ReFRESCO. $Rn = 10^7$. (Colors are visible in the online version of the article; http://dx.doi.org/10.3233/ISP-130083.)

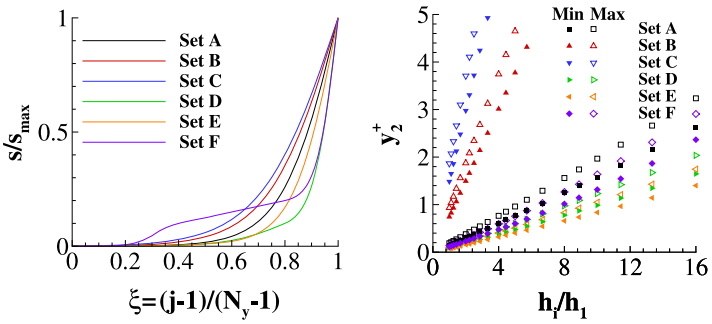

Fig. 7. Stretching functions and y_2^+ for six different grid sets. MS1 with $Rn = 10^7$. (Colors are visible in the online version of the article; http://dx.doi.org/10.3233/ISP-130083.)

The figure also contains the convergence of the friction resistance coefficient C_F of the bottom "wall" as a function of the grid refinement ratio. As for the previous results, the eddy-viscosity is determined from the manufactured solution and so there

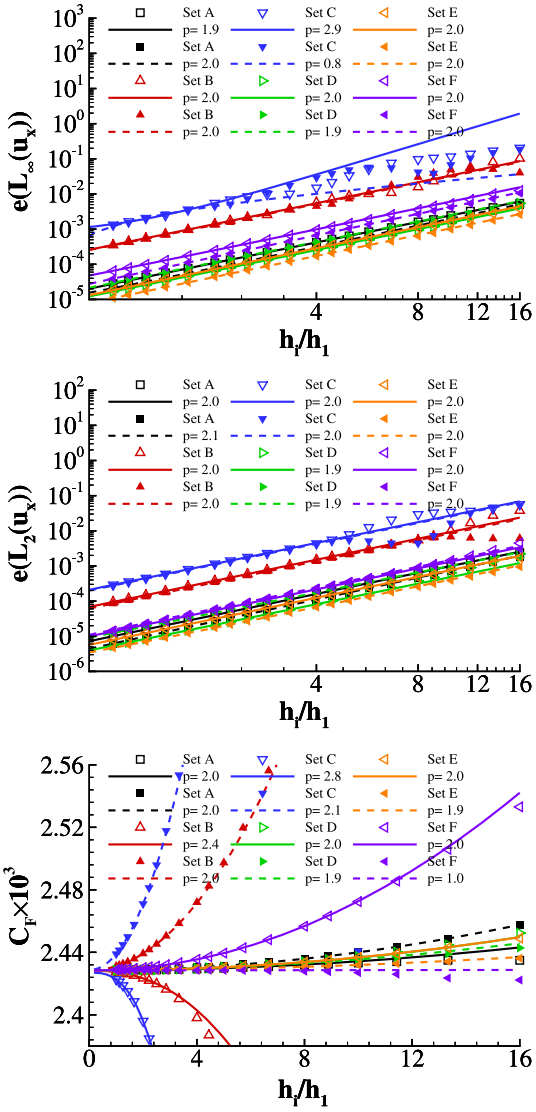

Fig. 8. L_∞ and L_2 norms of the error of the horizontal velocity component u_x and friction resistance coefficient of the bottom wall calculated with PARNASSOS (empty symbols and solid lines) and REFRESCO (filled symbols and dashed lines). MS1 with $Rn = 10^7$. (Colors are visible in the online version of the article; http://dx.doi.org/10.3233/ISP-130083.)

is no influence of any turbulence quantities transport equations in the convergence properties of the mean flow quantities. Nevertheless, there are several interesting features in the data.

As expected, the increase of y_2^+ (sets B and C) causes a deterioration of the convergence properties of the two codes, but interestingly the results of PARNASSOS increase to the correct value with the grid refinement whereas the data obtained with ReFRESCO show the opposite trend. Therefore, a comparison between the two codes for $h_i/h_1 = 4$ (a 200×200 cells grid) of sets B and C would suggest a misleading difference between the two codes that vanishes with grid refinement. The influence of y_2^+ is also visible in the convergence properties of the horizontal velocity component u_x. In particular, it is clear that for the L_∞ norm of the error of u_x most of the data of set C is outside the "asymptotic range" for both codes.

Although the data obtained for sets A, D, E and F (similar y_2^+) exhibit much less differences than those observed for sets B and C (largest y_2^+), it is clear that the selected stretching function influences the error level of C_F and u_x (constant α in Eq. (1)). In particular, set F leads to the largest grid dependency in C_F of these four sets for PARNASSOS and to the smallest with ReFRESCO, where the convergence is not monotonic (C_F increases for $h_i/h_1 > 3.33$ and it decreases for the finest grids), but the changes in C_F for the last six finest grids are smaller than 2×10^{-8}. Therefore, in this case, the observed order of accuracy of $p = 1$ is meaningless.

The next example is related to the one-equation model of Spalart & Allmaras [40], for which we have a manufactured undamped eddy-viscosity $\tilde{\nu}$ field [16] and the forcing source term in the $\tilde{\nu}$ transport equation. Figure 9 presents the convergence of the horizontal velocity component u_x and eddy-viscosity ν_t (which is similar to the undamped eddy-viscosity $\tilde{\nu}$) with the grid refinement. Results were obtained with PARNASSOS and ReFRESCO in grid set A with three different options in the discretisation of the convective terms of the $\tilde{\nu}$ transport equation: first-order upwind (TM1) and the third-order quick scheme [29] with (TML) and without (TM3) limiters.

The expected second-order convergence is observed for all TM3 solutions. In the MS2, the use of first-order upwind in the convective terms of the $\tilde{\nu}$ transport equations leads to first-order convergence for u_x in both codes. On the other hand, for the MS1 the effect in u_x is smaller than in the MS2 with the order of u_x dropping to values close to 1.5. This is not a surprising result, because MS1 has the largest gradients in the "near-wall" region which is essentially dominated by diffusion. On the other hand, the effect of the use of limiters (TML) is significantly different in the two codes: the results of ReFRESCO show an almost negligible effect of the limiters, whereas the data obtained with PARNASSOS are shifting from the TM3 to the TM1 solution with grid refinement.

The last example to illustrate the potential of Code Verification is the calculation of MS1 and MS1A with the turbulent/non-turbulent (TNT) version of the k–ω two-equation model [25]. Figure 10 presents the L_2 norm of the error of u_x, ν_t, k and ωy^2 as a function of the grid refinement. The figure contains a third plot that corresponds to calculations of MS1 with a fixed eddy-viscosity (ν_t taken from the MS). As mentioned above ω tends to infinity at the "wall" (ω is specified at the first node/cell

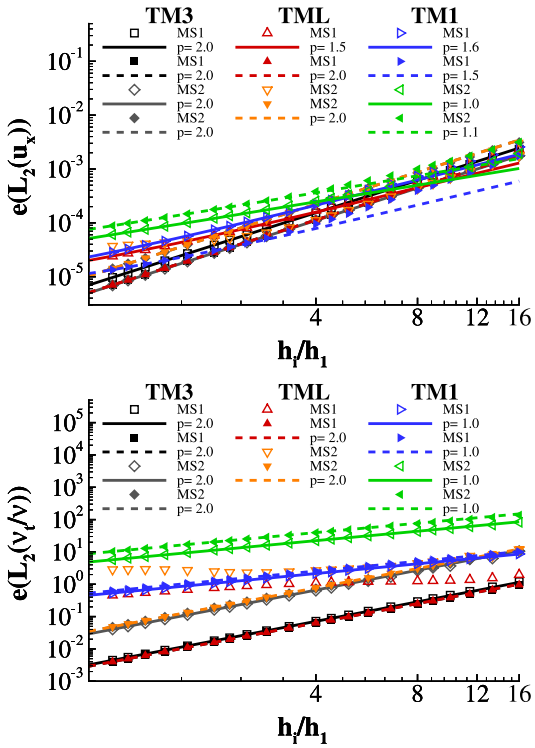

Fig. 9. L_2 norms of the error of the horizontal velocity component u_x and eddy-viscosity ν_t calculated with PARNASSOS (empty symbols and solid lines) and ReFRESCO (filled symbols and dashed lines). MS1 and MS2 with the Spalart & Allmaras model and $Rn = 10^7$. (Colors are visible in the online version of the article; http://dx.doi.org/10.3233/ISP-130083.)

centre away from the "wall") in both solutions and Fig. 10 contains therefore the error of ωy^2 that remains finite at the "wall". All the results plotted in Fig. 10 were obtained with TM3.

The results obtained with the two codes are very similar. The observed order of accuracy of mean and turbulence flow quantities depends on the selected MS. All flow quantities depicted in Fig. 10 exhibit p close to 1 for MS1, whereas p close to 2 is obtained for MS1A. The results obtained with fixed ν_t in MS1 show that k remains with p close to 1, whereas the other two variables show $p = 2$. However, it is likely that this result is a consequence of the singular behaviour of ω at the "wall".

Code Verification is in essence a one-time exercise. It is to be reiterated only if the code is modified. The MMS is a highly effective tool in Code Verification. It can detect even small errors, which can go unobserved for a long time in normal practice. With the examples given above we hope to have shown its power, but many more examples could be given. Try it!

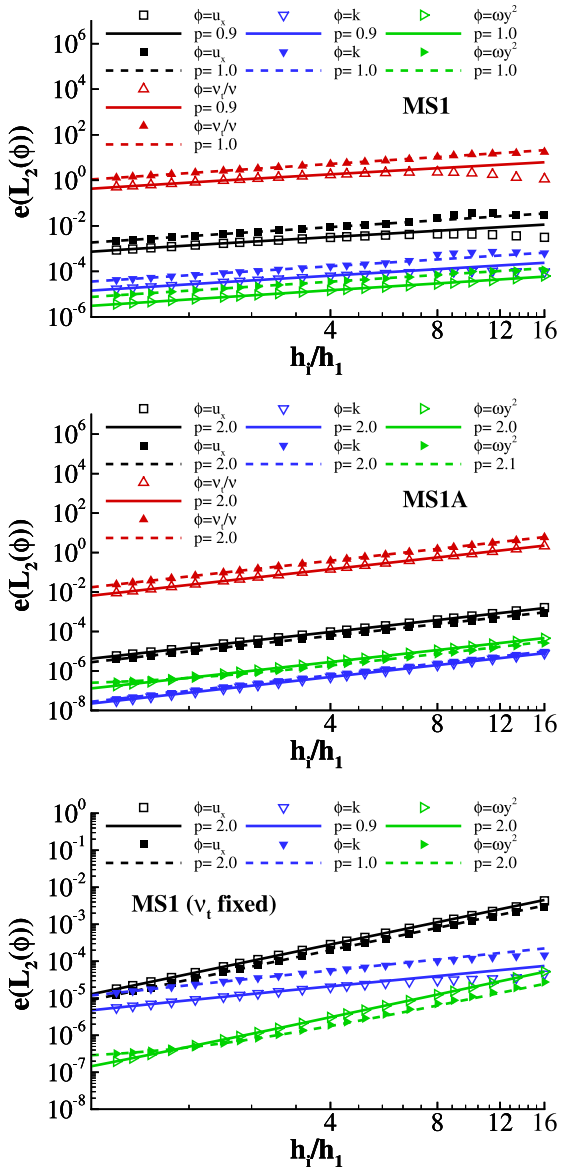

Fig. 10. L_2 norms of the error of the horizontal velocity component u_x, eddy-viscosity ν_t, turbulence kinetic energy k and turbulence frequency multiplied by distance to the "wall" squared ωy^2 calculated with PARNASSOS (empty symbols and solid lines) and ReFRESCO (filled symbols and dashed lines). MS1 (with and without fixed ν_t) and MS1A with the TNT k–ω model and $Rn = 10^7$. (Colors are visible in the online version of the article; http://dx.doi.org/10.3233/ISP-130083.)

5. Solution verification

The goal of Solution Verification is to estimate the uncertainty of a given numerical prediction U_ϕ (for which the exact solution is usually unknown), i.e. an interval that contains the exact solution with 95% confidence,

$$\phi_i - U_\phi \leqslant \phi_{\text{exact}} \leqslant \phi_i + U_\phi. \tag{9}$$

Therefore, unlike Code Verification, it is supposed to be done for all (if possible) numerical calculations.

As we mentioned above, the uncertainty is obtained from an error estimate times a safety factor. During the last 12 years we have developed and tested a procedure based on power series expansions and grid refinement studies to make reliable estimates of U_ϕ. A detailed description of the method is given in [11]. Here, we will only highlight its main options.

Equation (1) is the basic equation to obtain the error estimate (δ_{RE}). However, the calculation of δ_{RE} relies on the existence of a single parameter to characterize the grid density (the typical cell size) and (monotonically convergent) data in the "asymptotic range", which are assumptions really hard (if not impossible) to comply with in complex flows (turbulent flow) and complex geometries. The consequence is that scatter appears in the data [4]. A main contributor to noisy data is the lack of geometrical similarity of the grids [12]. While structured grids essentially allow geometrical similarity to be obtained, this is hardly true for unstructured grids. Other sources of scatter in the data are flux limiters, commonly used in the discretisation of convective terms, as well as damping functions and switches being part of many present-day turbulence models. Therefore, in complex flows it is an exception rather than a rule that the conditions required for the reliable use of Eq. (1) are met.

In order to be able to deal with the shortcomings of "practical calculations", we have added three other error estimators (assuming that the CFD code is theoretically second-order accurate):

$$\epsilon_\phi \simeq \delta_1 = \phi_i - \phi_o = \alpha h_i, \tag{10}$$

$$\epsilon_\phi \simeq \delta_2 = \phi_i - \phi_o = \alpha h_i^2 \tag{11}$$

and

$$\epsilon_\phi \simeq \delta_{12} = \phi_i - \phi_o = \alpha_1 h_i + \alpha_2 h_i^2. \tag{12}$$

These three alternatives are only used if the estimation with Eq. (1) is impossible or not reliable, i.e. the observed order of accuracy is either too small or too large ($p < 0.5 \wedge p > 2$). The first two options ((10) and (11)) are suitable for monotonically converging solutions only whereas the latter can be used as well with non-monotonic convergence.

The error estimators presented above require three grids (using Eqs (1) and (12)) or two grids (using (10) and (11)) to estimate an error. But error estimation based on three (or two) grids is not reliable for noisy data due to the extreme sensitivity of the determination of p to small perturbations in the data [4] (the use of the alternative error estimators depends on the estimated p). Therefore, it is virtually impossible to decide whether or not a given set of data (with the minimum number of grids required) is in the "asymptotic range" (in the presence of scatter, an observed order of accuracy equal to the formal order of accuracy may be fortuitously obtained and is not sufficient to label the data set as being in the "asymptotic range"). Furthermore, a single grid triplet gives only one instance of p, because Eq. (1) has three unknowns. Redundancy, and therefore the possibility of a quality check on the value of p, only occurs when the fourth grid is added! Therefore it is highly recommendable to use at least four grids when some scatter in the data is expected, i.e. for most engineering flow problems.

When more than three grids are available, the error estimators defined by Eqs (1), (10)–(12) may be solved in the least squares sense [11] to obtain an error estimation and an observed order of accuracy in the case of Eq. (1). Furthermore, the least squares approach also produces a measure of the quality of the error estimation: the standard deviation of the fit, σ.

In many "practical applications" it is hard to avoid the use of grids that are coarser than desirable. Therefore, we also perform weighted least squares fits to the error estimators [11] being the selected option (weighted or non weighted) decided by σ. Finally, to complete the "tools" required for the estimation of the uncertainty of a flow quantity $U(\phi)$ we define a data range parameter as

$$\Delta_\phi = \frac{(\phi_i)_{\max} - (\phi_i)_{\min}}{n_g - 1}, \tag{13}$$

where n_g stands for the number of grids available.

The proposed procedure starts by estimating the error from the four alternatives available. The error estimation is considered reliable if the solution is monotonically convergent with $0.5 < p < 2.1$[1] and if $\sigma < \Delta_\phi$. Following the Grid Convergence Index (GCI) procedure [34,36], the safety factor is chosen as $F_s = 1.25$ if the error estimate is deemed reliable, else $F_s = 3$.

The determination of U_ϕ, usually for the finest grid (best) solution, but essentially for any ϕ_i of a data set, is not only dependent on F_s. Different expressions are adopted for "good" and "bad" error estimations, i.e. for $\sigma \leqslant \Delta_\phi$ or $\sigma > \Delta_\phi$.

- For $\sigma \leqslant \Delta_\phi$:

$$U_\phi(\phi_i) = F_s \epsilon_\phi(\phi_i) + \sigma + |\phi_i - \phi_{\text{fit}}|. \tag{14}$$

[1] Although the error estimation does not allow $p > 2$, it is reasonable to assume a tolerance for the change in the safety factor.

- For $\sigma > \Delta_\phi$:

$$U_\phi(\phi_i) = 3\frac{\sigma}{\Delta_\phi}\big(\epsilon_\phi(\phi_i) + \sigma + |\phi_i - \phi_{\text{fit}}|\big). \tag{15}$$

The estimated uncertainty has therefore three components: the absolute value of the estimated discretisation error times a safety factor; the standard deviation of the fit; the difference between the real data point and the value obtained from the fit for the same grid density. Obviously, the latter two quantities are just a consequence of noise in the data. Consequently, for a grid refinement study with data without noise the method reduces to the well-known GCI [34,36].

It could be argued that in the cases with $\sigma > \Delta_\phi$ it is not allowed to base an uncertainty estimation on the present error estimators. However, the estimated uncertainty with Eq. (15) will then be high, indicating clearly that the data quality is bad (due to e.g. grids without similarity, incomplete iterative convergence, bugs in the code ...), thus still conveying a useful message.

5.1. Flat plate flow

The first example of a Solution Verification exercise is the simple (but highly instructive) flow over a flat plate at a Reynolds number based on the undisturbed velocity U_∞ and plate length L of 10^7. In this case, all calculations have been performed with PARNASSOS using the shear-stress transport (SST) version of the two-equation k–ω model [30]. Calculations were made with double-precision (15 digits) and the iterative error has been reduced to machine accuracy. Therefore, the numerical uncertainty is mainly due to the discretisation error.

The calculation domain is a rectangle with the inlet at $x = -0.25L$, the outlet at $x = 1.25L$ and the top boundary at $y = 0.25L$. It has been checked in [7,8] that with these options the results are not affected by the limited domain size. The following boundary conditions have been applied: at the inlet boundary, the velocity is set equal to undisturbed flow conditions ($u_x = U_\infty$ and $u_y = 0$), k and ω are specified to guarantee $(\nu_t)_{\text{inlet}} = 0.01\nu$ and the pressure is extrapolated from the interior of the domain; at the outlet boundary, the derivatives with respect to x of all flow quantities are assumed to be zero; at the top boundary, the horizontal velocity component is set equal to the undisturbed flow velocity, U_∞, zero normal derivatives are assumed for k and ω and the pressure coefficient is set equal to zero; on the two segments of the bottom boundary coinciding with the symmetry line, upstream of the plate and in the wake, we impose $u_y = 0$ and zero normal derivatives for k and ω, whereas u_x and C_p are computed from the momentum equations in the x and y directions, imposing the symmetry conditions at virtual grid nodes; at the plate surface, u_x, u_y and k are set equal to zero, ω is specified at first grid node away from the wall [5] and zero normal derivative is assumed for the pressure.

In this example, we consider three sets of nine geometrically similar Cartesian grids. All grids have the same type of grid node distribution in the longitudinal (x) direction (see [10] for a detailed description) and one-sided stretching functions [43]

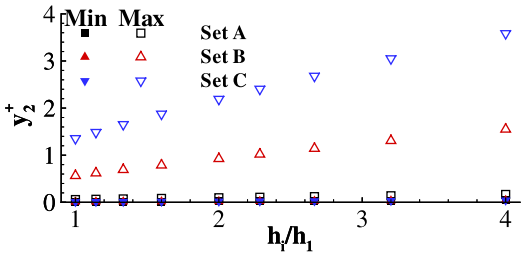

Fig. 11. Maximum and minimum y_2^+ for the three grid sets used in the calculation of the flow over a flat plate at $Rn = 10^7$. (Colors are visible in the online version of the article; http://dx.doi.org/10.3233/ISP-130083.)

are applied in the vertical direction to cluster grid nodes close to the wall. The three sets differ only in the selected stretching parameter with the smallest near-wall spacing for set A and the largest for set C. The minimum and maximum y_2^+ values are depicted in Fig. 11. It must be mentioned that the average value of y_2^+ is below 1 for all grids of the three sets.

We have performed calculations using the so-called "blended scheme" in the discretisation of the convective terms of the momentum equations. $\beta = 0$ stands for first-order upwind, whereas $\beta = 1$ corresponds to third-order upwind. In order to illustrate the convergence properties as a function of h_i/h_1 and β we have selected the friction resistance coefficient C_F and the displacement thickness δ^*/L at $x = 0.5L$.

The results obtained for C_F are presented in Fig. 12. As expected, all grid sets and β values show similar extrapolated values to cell size zero ($h_i/h_1 = 0$). Thus, all estimated uncertainties overlap. However, the estimated uncertainties and the observed order of accuracy are both dependent on the grid line spacing and β. Plausibly, C_F decreases with the reduction of β, but the effect of the near-wall grid spacing on the convergence properties of C_F is stronger than the effect of β. It is also clear that the data obtained on the coarsest grids are not in the "asymptotic range", but due to the simplicity of the computational domain, there is no scatter in the data.

The trends observed in δ^*/L at $x = 0.5L$ are significantly different from those obtained for C_F. The data plotted in Fig. 13 show again a significant influence of the grid line distribution and β on the convergence properties of δ^*/L. However, in this case β has a much stronger effect than the grid line spacing distribution. This can be explained by the fact that δ^* is more dependent on convection than C_F. It is also important to remark that many of the error estimates performed for the data of Fig. 13 can not use Eq. (1) because the convergence is not monotonic. Nevertheless, consistent error estimates are obtained using the alternative power series expansions and in particular Eq. (12).

5.2. *Flow around the KVLCC2 tanker*

The second example of a Solution Verification exercise is the calculation of the flow around the KVLCC2 tanker at model (4.6×10^6) and full (2.03×10^9) scale

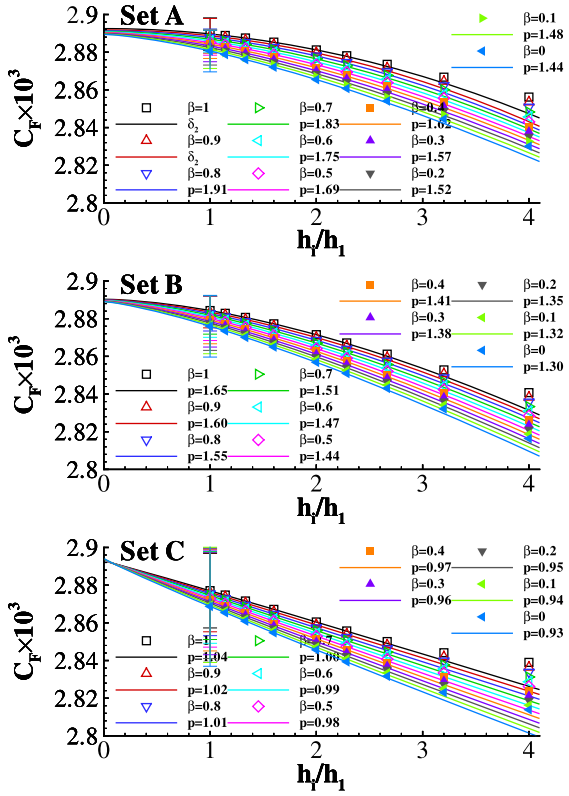

Fig. 12. Friction resistance coefficient C_F as a function of the grid refinement ratio for the flow over a flat plate at $Rn = 10^7$. SST k–ω turbulence model and convective terms of the momentum equations approximated with a blend of first-order ($\beta = 0$) and third-order ($\beta = 1$) upwind. (Colors are visible in the online version of the article; http://dx.doi.org/10.3233/ISP-130083.)

Reynolds number (based on the undisturbed flow velocity, U_∞, and length between perpendiculars, L_{PP}). The results were obtained with PARNASSOS using the SST k–ω model [30] with double-precision and a tolerance for the iterative convergence that guarantees a negligible influence of the iterative error.

The computational domain has a cylindrical shape. The boundaries of the domain are two x = constant planes as the inlet and outlet, the planes $y = 0$ (symmetry plane of the ship) and $z = 0$ (still water plane, no gravity waves were included), the surface of the ship and a cylindrical surface (external boundary) with the axis along the x direction at the intersection of the symmetry plane of the ship and the still water plane. The distance from inlet to the forward perpendicular, from aft perpendicular to outlet and from x-axis to external boundary is equal to L_{PP}.

At the inlet plane, the velocity components are specified from a potential flow solution and the pressure is extrapolated from the interior of the domain. The tur-

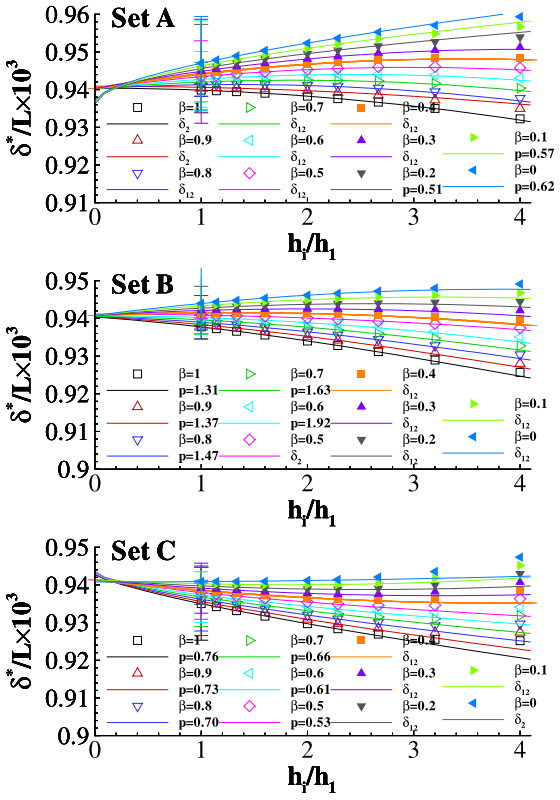

Fig. 13. Displacement thickness δ^*/L at $x = 0.5L$ as a function of the grid refinement ratio for the flow over a flat plate at $Rn = 10^7$. SST k–ω turbulence model and convective terms of the momentum equations approximated with a blend of first-order ($\beta = 0$) and third-order ($\beta = 1$) upwind. (Colors are visible in the online version of the article; http://dx.doi.org/10.3233/ISP-130083.)

bulence quantities are specified to obtain $\nu_t = 0.01\nu$ and $\omega = \frac{10U_\infty}{L_{PP}}$. Zero streamwise derivatives are assumed for all flow variables at the outlet plane. The symmetry plane of the ship coincides with two boundaries of the computational space ((i, j, k) or (ξ, η, ζ)): $j = 1, \eta = 0$ and $k = 1, \zeta = 0$. In the former case, the normal velocity, $V(2)$, is set equal to zero and the tangential velocity components ($V(1)$ and $V(3)$) and pressure are obtained from the solution of the momentum equations with the symmetry conditions applied by referring to virtual nodes. Normal derivatives of all turbulence quantities are set equal to zero. For the other boundary, the girthwise velocity, $V(3)$, is set equal to zero and the girthwise derivative (ζ) is set equal to zero for the tangential velocity components ($V(1)$ and $V(2)$), pressure and turbulence quantities.

For the still water plane, $V(3)$ is set to equal to zero and the tangential velocity components ($V(1)$ and $V(2)$) and pressure are obtained from the solution of the

continuity and ξ and η momentum equations with the symmetry conditions applied at virtual nodes. For the turbulence quantities, the girthwise derivatives are set equal to zero.

At the external boundary, tangential velocity components and pressure are imposed, as derived from a potential flow calculation; whereas zero normal derivatives are assumed for all turbulence quantities.

At the ship surface, the no-slip condition is applied directly at the wall, i.e. velocity components set equal to zero and wall shear-stress computed from the normal velocity derivative at the ship surface. The normal pressure derivative is assumed to be equal to zero. The turbulence kinetic energy k is set equal to zero and ω is specified at the first grid node away from the wall [5] using the near-wall analytical solution for smooth surfaces.

A set of six nearly geometrically similar single-block H-O grids was generated for each Reynolds number. The two sets have identical number of nodes in the streamwise (N_ξ) and girthwise (N_ζ) directions. In the normal (N_η) direction the grids for the full scale calculations have more grid nodes than those used for model scale. The coarsest grids include $280 \times 80 \times 30$ cells for $Rn = 4.6 \times 10^6$ and $280 \times 100 \times 30$ cells for $Rn = 2.03 \times 10^9$, whereas the finest grids include $560 \times 160 \times 60$ and $560 \times 200 \times 60$ cells, respectively. Both sets cover a grid refinement ratio of 2 and the near-wall distances were tuned to obtain the maximum y^+ of the first grid node away from the wall below 0.6. Figure 14 illustrates the coarsest grids in the bow and stern regions. Downstream of the stern, the grids have a singular line along the wake extending from the keel of the ship, whereas upstream of the bow the singular line is located slightly above the keel line.

As for the flat plate flow, we have performed calculations with β ranging from zero (first-order) to one (third-order) in the upwind scheme applied to the convective terms of the momentum equations. We have selected three significant flow quantities of the calculation of viscous flows around ships to illustrate the results of the Solution Verification exercise: the friction resistance coefficient C_F, the pressure resistance coefficient C_P and the wake fraction coefficient W_f calculated at the propeller plane ($x = 0.9825 L_{PP}$) according to the instructions of the 2010 Workshop [26].

The results obtained at model scale Reynolds number ($Rn = 4.6 \times 10^6$) are presented in Fig. 15. The convergence properties obtained for C_F are similar to those observed in the flat plate flow. However, for this flow $p \leqslant 1$ for $\beta \leqslant 0.4$. Although

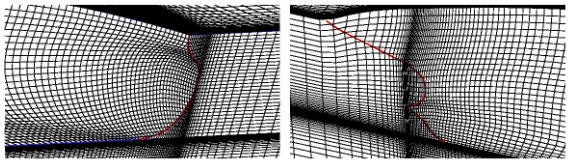

Fig. 14. Illustration of the grid at the bow and stern regions of the KVLCC2 tanker. (Colors are visible in the online version of the article; http://dx.doi.org/10.3233/ISP-130083.)

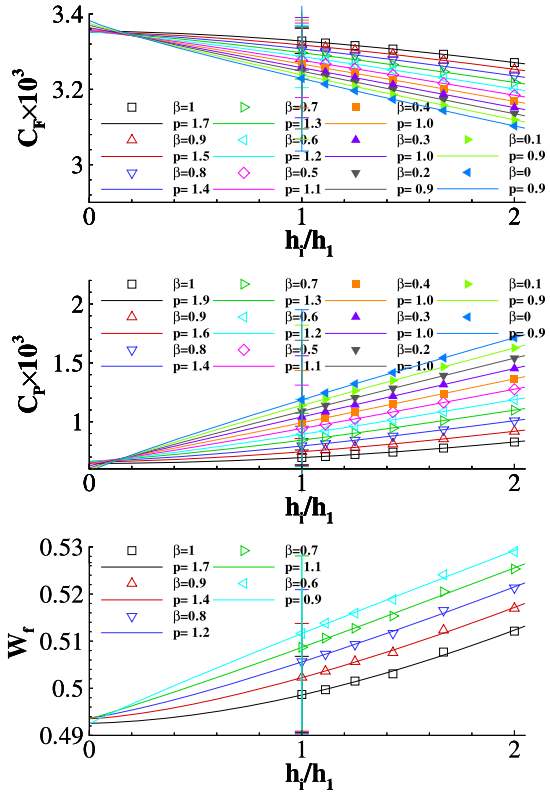

Fig. 15. Friction C_F and pressure C_P resistance coefficients and wake fraction at the propeller plane ($x = 0.9825 L_{PP}$) as a function of the grid refinement ratio. KVLCC2 tanker at model scale Reynolds number, $Rn = 4.6 \times 10^6$. SST k–ω turbulence model and convective terms of the momentum equations approximated with a blend of first-order ($\beta = 0$) and third-order ($\beta = 1$) upwind. (Colors are visible in the online version of the article; http://dx.doi.org/10.3233/ISP-130083.)

qualitatively the C_P results are similar to those obtained for C_F, the impact of β and of the grid refinement ratio on the value of C_P is clearly stronger than on C_F. Just by changing β in the finest grid we obtain a change of more than 70% and if we include h_i/h_1 all the numerically computed values[2] of C_P range from 0.7×10^{-3} to 1.71×10^{-3}! However, all grid refinement studies are naturally converging to the same solution. In the plot of W_f versus h_i/h_1 we have deliberately plotted only the lines that have $p \geqslant 1$. The purpose is to illustrate the existence of scatter in the data that would not be evident with a scale that fits the solutions obtained for all the values

[2]The "careless" CFD practitioner aiming for a match between his calculation and an experiment must be very unlucky to miss his target ....

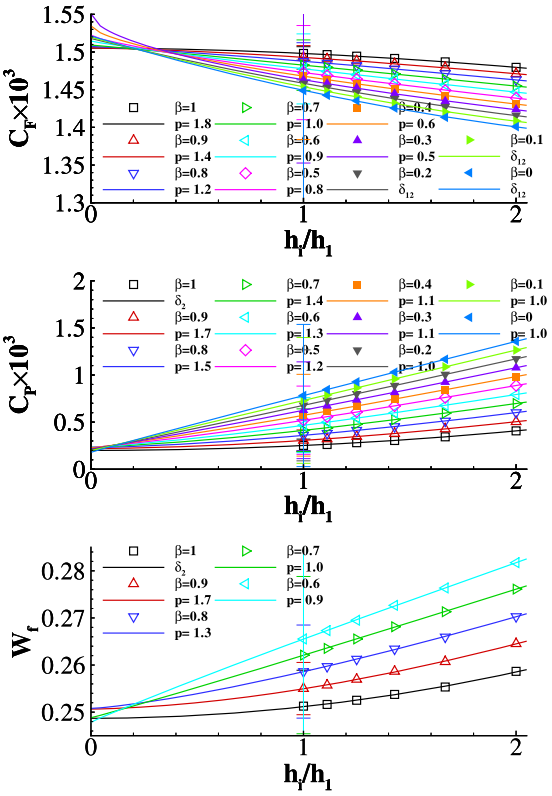

Fig. 16. Friction C_F and pressure C_P resistance coefficients and wake fraction at the propeller plane ($x = 0.9825 L_{PP}$) as a function of the grid refinement ratio. KVLCC2 tanker at full scale Reynolds number, $Rn = 2.03 \times 10^6$. SST k–ω turbulence model and convective terms of the momentum equations approximated with a blend of first-order ($\beta = 0$) and third-order ($\beta = 1$) upwind. (Colors are visible in the online version of the article; http://dx.doi.org/10.3233/ISP-130083.)

of β and h_i/h_1 tested. In such conditions, the least squares approximation is crucial to obtain a reliable error/uncertainty estimation.

Figure 16 presents the results obtained at full scale Reynolds number. The trends are similar to those obtained at model scale Reynolds number. Nevertheless, the effect of β on C_F is stronger than in the model scale data. Interestingly, there is less scatter in the wake fraction data at full scale than at model scale.

The importance of Solution Verification is clearly demonstrated in Fig. 17, presenting the ratio between the viscous resistance coefficient $C_V = C_F + C_P$ of the KVLCC2 tanker at full and model scale Reynolds number as a function of the grid refinement ratio h_i/h_1 and of the blending parameter β of the approximation of the convective terms of the momentum equations. The values of the viscous resistance ratio obtained in the six different grids with β ranging from 0 to 1 exhibit changes

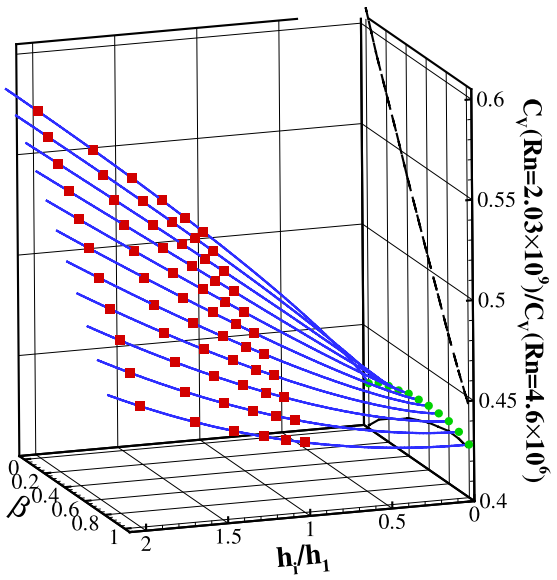

Fig. 17. Ratio of the viscous resistance coefficient $C_V = C_F + C_P$ of the KVLCC2 tanker at full and model scale as a function of the grid refinement ratio h_i/h_1 and blending parameter β of the convective terms of the momentum equations. SST k–ω turbulence model. (Colors are visible in the online version of the article; http://dx.doi.org/10.3233/ISP-130083.)

that reach 33%. Furthermore, if one decides to make a single grid calculation for $h_i/h_1 \simeq 1.43$ (roughly 2×10^6 cells) with $\beta = 0.5$ the difference to the most accurate solution plotted in Fig. 17 ($h_i/h_1 = 1$ and $\beta = 1$) is 14%.

It may be argued that Solution Verification involves "too much work". It is indisputable that it requires more work than obtaining a single-grid solution. But the extra effort will pay off handsomely by giving you a hold on the reliability of your results. Besides, scientifically the amount of work to be done is never a good argument to refrain from a job.

It must be clear that the agreement between simulation and experiment is not enough justification for the quality of a solution. The examples above show clearly that the range of values that can be obtained by "playing" with numerical settings is so large that in many cases it is not hard to find the grid refinement level (h_i/h_1) and/or discretisation scheme (β) that produces a match between simulation and experiment.

6. Validation

The goal of Validation is to quantify the modelling uncertainty, i.e. to quantify how well the mathematical models represent the physical world. Therefore, the first

task in Validation is to define what are the quantities of interest. A given model may be adequate in determining some flow quantities, but fall short to simulate others. Once such decision has been made, a second challenge appears: what is the most appropriate Validation metric? The answer to this question is still a research topic and there are several proposals available in the open literature [1,37,38].

In this paper, we will only illustrate the ASME V&V 20 standard [1], which has been applied in a simplified form in the third Lisbon Workshop [15]. The goal of the procedure is to estimate the interval that contains the modelling error, δ_{model}, with 95% confidence: $[E-U_{\text{val}}, E+U_{\text{val}}]$. E is the validation comparison error E defined as the difference between the simulation value (S) and the experimental data value (D): $E = S-D$. The validation uncertainty U_{val} is composed of three contributions: the estimated numerical uncertainty U_{num}, the estimated experimental uncertainty U_D and a so-called "parameter uncertainty", U_{input}. Although the combination of these contributions is still being discussed [35], the proposal of [1] is

$$U_{\text{val}} = \sqrt{U_{\text{num}}^2 + U_{\text{input}}^2 + U_D^2}. \tag{16}$$

It should be mentioned that U_{input} is supposed to include all uncertainty contributions due to lack of knowledge in the flow conditions, for example Reynolds number, inlet conditions or domain geometry. The determination of U_{input} may be extremely time consuming (typical problems of uncertainty quantification UQ based on Monte Carlo methods) and the simplified version tested in Lisbon [15] just assumed $U_{\text{input}} = 0$.

The outcome of the procedure is the following:

(a) If $|E| \geqslant U_{\text{val}}$ the modelling error δ_{model} is likely to be comparable in magnitude with $|E|$, i.e. $\delta_{\text{model}} \simeq |E|$. If δ_{model} exceeds an acceptably low level, the model must be considered as not good enough to represent the flow physics and the exercise suggests that the model must be improved (e.g. LES instead of RANS, or a Reynolds-stress model instead of a one-equation turbulence model). On the other hand, if the estimated δ_{model} is sufficiently small one may still be justified to accept the model as good enough for practical purposes.

(b) If $|E| \leqslant U_{\text{val}}$ the validation exercise indicates that the modelling error is smaller than the level of U_{val}, but it is not necessarily a quality assignment. If U_{val} is large, attempts should be made to make it smaller, either by better numerical simulations or more accurate experiments, or both. If on the other hand the validation uncertainty is small we have a strong corroboration of the model.

Notice that both cases require the definition of an acceptable threshold, which will always depend on the analyst. Furthermore, this evaluation is done per problem and per selected quantity of interest. Consequently, we cannot speak of a "validated code".

Table 1
Predicted (S) and experimental (D) re-attachment point with the validation comparison error ($|E|$) and the validation uncertainty (U_{val})

Code	TM	S	D	$\|E\|$	U_{val}
FLUENT	SA	5.77	6.26	0.49	0.19
ISIS	SA	6.02	6.26	0.24	0.10
OpenFoam	SA	5.94	6.26	0.32	0.10
CFL3D	SA	6.01	6.26	0.25	0.10
FUN3D	SA	6.06	6.26	0.20	0.11
PARNASSOS	SA	6.08	6.26	0.18	0.12
FLUENT	SKE	5.55	6.26	0.71	1.36
CADYF	SKE	5.47	6.26	0.79	0.10
CHAPMAN	BSL	5.52	6.26	0.74	0.60
PARNASSOS	BSL	5.44	6.26	0.82	0.63
PARNASSOS	SST	6.21	6.26	0.05	0.61

Note: Flow over a backward facing step.

6.1. Example from the Lisbon Workshop

In order to give an example of the ASME V&V 20 validation procedure, we present in Table 1 the results submitted to the third Lisbon Workshop [15] for the flow over a backward facing step with regard to the position of the re-attachment point (divided by the step height). In this table, TM stands for turbulence model: SA for the Spalart & Allmaras model [40], SKE for the standard $k-\epsilon$ model [27], BSL for the baseline version of the k–ω model [30] and SST for the shear-stress transport version of the k–ω model [30].

As mentioned above U_{val} is obtained from U_{num} and U_D only so that the different levels obtained by the participants are a consequence of the numerical uncertainty U_{num} only. All submissions with the SA model exhibit $E > U_{\text{val}}$ and in several cases U_{val} is of the order of 1.5%, which indicates that the Spalart & Allmaras model is not able to predict the re-attachment point within such level of validation uncertainty.

There are only two results that satisfy $E < U_{\text{val}}$: the single submission with the SST model and one of the submissions with the SKE model. However, U_{val} is roughly 10% for the SST model and 22% for the FLUENT SKE solution. On the other hand, the other SKE solution exhibits $E \gg U_{\text{val}}$ because U_{num} of the CADYF solution is significantly smaller than U_{num} in the FLUENT solution. Again a warning that the statement "solution validated" does not necessarily imply a high-quality solution.

6.2. Example from the Gothenburg Workshop

The second example of the use of the ASME V&V 20 validation procedure is the flow around the KVLCC2 tanker for double-body conditions and model scale

Reynolds number, $Rn = 4.6 \times 10^6$, which was part of the 2010 Workshop on Numerical Ship Hydrodynamics [26]. Experimental data are available from wind tunnel (double-body) [28] and towing tank measurements [23]. Wind tunnel experiments included the measurement of mean and turbulence quantities at several cross-stream planes, but no forces were measured. On the other hand, towing tank tests include the total resistance and mean velocity fields at several cross-stream planes for a Froude number of $Fr = 0.142$.

As stated in [28], wind tunnel measurements have no velocity correction for blockage effect and the reported uncertainties of the mean velocity components are 0.5% for the axial component u_x, 0.7% for the horizontal component u_y and 1.0% for the vertical component u_z. For the turbulence kinetic energy k, the reported uncertainty is 12.8%. Although the paper [23] includes a section called "Measurement and uncertainty analysis", there are no uncertainties reported. Therefore, none of the available experimental data sets conforms to the conditions set for the calculations of the flow around the KVLCC2 (double-body) at model scale Reynolds number. The tests reported in [28] have identical geometry, but include an unaccounted blockage effect of the tunnel walls. Even if the Froude number used in [23] is rather small ($Fr = 0.142$), towing tank tests include gravity waves and experimental uncertainties are not reported. Nevertheless, the results presented below illustrate the use of the ASME V&V Validation procedure and indicate that it goes beyond the typical graphical comparison of isolines as made at the CFD Workshops [26] (Fig. 18).

The total resistance coefficient reported in [23] is $C_T = 4.11 \times 10^{-3}$, whereas the calculated value (for the finest grid with $\beta = 1$) is $C_T = 4.025 \times 10^{-3} \pm 0.034 \times 10^{-3}$. The application of the ASME V&V 20 validation procedure is troublesome: the only contribution to U_{val} available is the numerical uncertainty and the

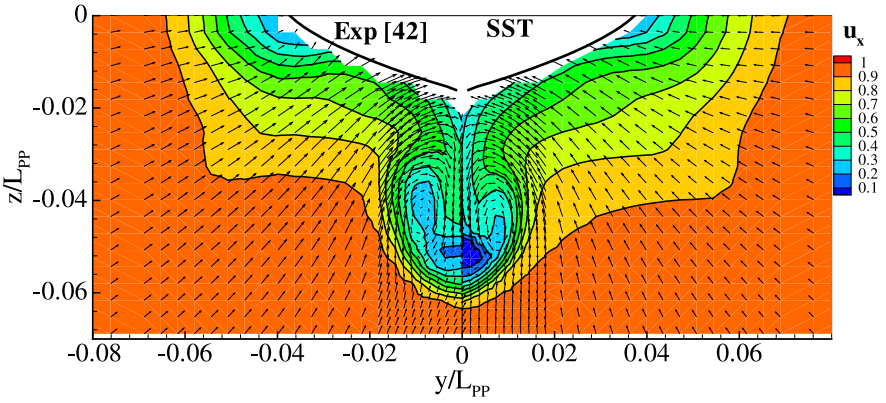

Fig. 18. Velocity field at the propeller plane ($x = 0.9825 L_{PP}$). Comparison of experimental results [28] and calculations performed with the SST k–ω turbulence model. KVLCC2 tanker at model scale Reynolds number, $Rn = 4.6 \times 10^6$. (Colors are visible in the online version of the article; http://dx.doi.org/10.3233/ISP-130083.)

experiment contains a wave resistance contribution that is not accounted for in the calculations. Nevertheless, an experimental uncertainty (plus parameter uncertainty) of 1.9% would make $|E| = U_{\text{val}}$. As already pointed out for the previous example, it is not difficult to produce results with $|E| < U_{\text{val}}$. If we consider the solution obtained with first-order upwind ($\beta = 0$, equivalent to using bad instruments in an experiment) we have $C_T = 4.416 \times 10^{-3} \pm 0.574 \times 10^{-3}$. Thus $|E| = 0.306 \times 10^{-3}$ and U_{val} (only with U_{num}) is 0.574×10^{-3}!

Figure 19 presents the comparison error $|E|$ and the validation uncertainty U_{val} of the axial mean velocity component u_x and of the turbulence kinetic energy k for an horizontal cut at the propeller plane ($x = 0.9825 L_{PP}$, $z = -0.0435 L_{PP}$). The figure also contains a plot with the direct comparison of experiments and simulations

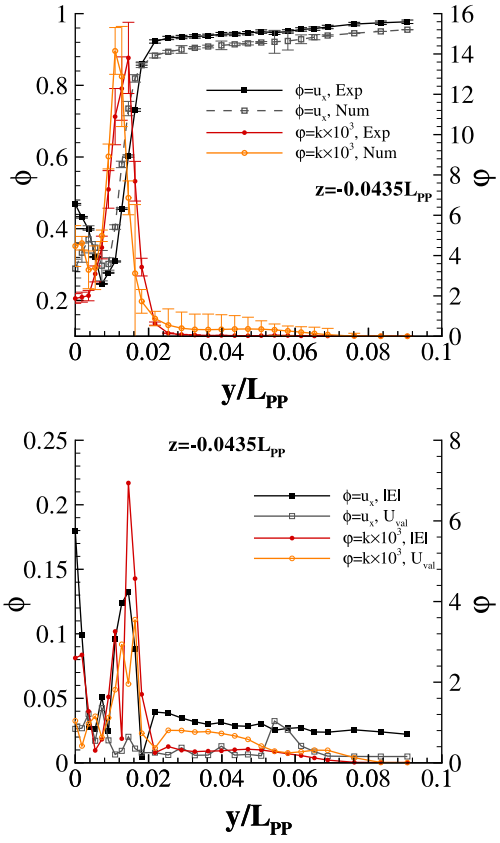

Fig. 19. Comparison error $|E|$ and validation uncertainty U_{val} for the axial mean velocity component u_x and turbulence kinetic energy k along an horizontal cut at the propeller plane ($x = 0.9825 L_{PP}$, $z = -0.0435 L_{PP}$). KVLCC2 tanker at model scale Reynolds number, $Rn = 4.6 \times 10^6$. SST k–ω turbulence model. (Colors are visible in the online version of the article; http://dx.doi.org/10.3233/ISP-130083.)

including the respective uncertainties. The results obtained for u_x show that U_{val} is always below 5% of U_∞ and that $|E|$ is mostly above U_{val}, especially close to the symmetry plane of the ship. However, it is likely that the comparison in the outer region of the flow is affected by the blockage effect of the wind tunnel. On the other hand, the values of U_{val} for k are larger (in percentage) than those obtained for u_x and so we have $|E|$ below or similar to U_{val} for most of the cut.

7. Closure

An experimentalist will develop an appeal for the limitations of his experimental facility and the shortcomings of his measuring devices by redundancy experiments, by confrontation of results from different devices to measure the same quantity, by changing the sensitivity range of his gauges, etc. Similarly, a CFD-practitioner will learn by making once in a while a careful Verification and Validation exercise. It will make him aware of the variability of his results dependent on the numerical settings.

That a thorough grid convergence study requires a big effort and is therefore time-consuming and costly is readily acknowledged. That in many cases an engineering application will continue to be carried out on a single grid without Solution Verification is also to be expected. But one should be realistic then and not come with unfounded reliability claims, even if the results look like the outcome of a corresponding experiment. After all such good agreement may turn out to be coincidental. Only if experience is built up with uncertainty estimation one will be able to appreciate the complexity of a numerical simulation and see the influence of options like grid density, grid lay-out, size of computation domain, solution convergence level, etc. It leads in the end to confidence in the use of numerical tools. And that confidence is needed, particularly in the circumstance that corresponding experimental data are not available and the CFD-results have to stand on their own.

Acknowledgements

The authors gratefully acknowledge the stimulating role of Patrick J. Roache in their work on Verification and Validation over the last ten years. The discussions with him and his comments/advices on our reporting have been of great value. We would also like to thank the CFD group of MARIN and our “companion in arms” Hoyte Raven for their support.

References

[1] ASME Committee PTC 61: ANSI Standard V&V 20: Guide on Verification and Validation in Computational Fluid Dynamics and Heat Transfer, 2009.

[2] F.G. Blottner, Accurate Navier–Stokes results for the hypersonic flow over a spherical nosetip, *Journal of Spacecraft and Rockets* **27**(2) (1990), 113–122.

[3] B.W. Boehm, Software engineering economics, *IEEE Transactions on Software Engineering* **SE-10**(1) (1984), 4–21.
[4] L. Eça and M. Hoekstra, An evaluation of verification procedures for CFD applications, in: *24th Symposium on Naval Hydrodynamics*, Fukuoka, Japan, July 2002.
[5] L. Eça and M. Hoekstra, On the grid sensitivity of the wall boundary condition of the k–ω model, *Journal of Fluids Engineering* **126**(6) (2004), 900–910.
[6] L. Eça and M. Hoekstra, On the influence of the iterative error in the numerical uncertainty of ship viscous flow calculations, in: *26th Symposium on Naval Hydrodynamics*, Rome, Italy, September 2006.
[7] L. Eça and M. Hoekstra, Code verification and verification of calculations with RANS solvers, in: *AVT-147 – Computational Uncertainty in Military Vehicle Design*, Athens, December 2007.
[8] L. Eça and M. Hoekstra, The numerical friction line, *Journal of Marine Science and Technology* **13**(4) (2008), 328–345.
[9] L. Eça and M. Hoekstra, Evaluation of numerical error estimation based on grid refinement studies with the method of the manufactured solutions, *Computers and Fluids* **38** (2009), 1580–1591.
[10] L. Eça and M. Hoekstra, Numerical aspects of including wall roughness effects in the SST k–ω eddy-viscosity turbulence model, *Computers and Fluids* **40** (2011), 299–314.
[11] L. Eça and M. Hoekstra, A procedure for the estimation of the numerical uncertainty of CFD calculations based on grid refinement studies, *Journal of Computational Physics* (2013), to appear.
[12] L. Eça, M. Hoekstra, J. Beja Pedro and J.A.C. Falcão de Campos, On the characterization of grid density in grid refinement studies for discretization error estimation, *International Journal for Numerical Methods in Fluids* (2012), DOI:10.1002/fld.3737.
[13] L. Eça, M. Hoekstra, A. Hay and D. Pelletier, On the construction of manufactured solutions for one and two-equation eddy-viscosity models, *International Journal of Numerical Methods in Fluids* **54** (2007), 119–154.
[14] L. Eça, M. Hoekstra, A. Hay and D. Pelletier, Verification of RANS solvers with manufactured solutions, *Engineering with Computers* **23**(4) (2007), 253–270.
[15] L. Eça, M. Hoekstra, P.J. Roache and H.C. Coleman, Code verification, solution verification and validation: an overview of the 3rd Lisbon Workshop, in: *AIAA Paper 3647, AIAA Computational Fluid Dynamics Conference*, San Antonio, June 2009.
[16] L. Eça, M. Hoekstra and G. Vaz, Manufactured solutions for steady-flow RANS solvers, *International Journal CFD* **26**(5) (2012), 313–332.
[17] Editorial on Experimental Uncertainty, *ASME Journal of Fluids Engineering*.
[18] Editorial Policy Statement, All AIAA Journals, January 2006, available at: https://www.aiaa.org.
[19] Editorial Policy Statement on the Control of Numerical Accuracy, *ASME Journal of Fluids Engineering* **130**(7) (2008).
[20] Facility Bias World Wide Campaign, Resistance Committee Report, in: *Proceedings of 25th ITTC*, Fukuoka, 2008.
[21] J.H. Ferziger and M. Peric, *Computational Methods for Fluid Dynamics*, Springer-Verlag, Berlin, 1996.
[22] M. Hoekstra, Numerical simulation of ship stern flows with a space-marching Navier–Stokes method, PhD thesis, Delft, Netherlands, 1999.
[23] W.J. Kim, S.-H. Van and D.H. Kim, Measurement of flows around modern commercial ship models, *Experiments in Fluids* **31** (2001), 567–578.
[24] P. Knupp and K. Salari, *Verification of Computer Codes in Computational Science and Engineering*, CRC Press, Boca Raton, FL, 2002.
[25] J.C. Kok, Resolving the dependence on free-stream values for the k–ω turbulence model, NLR-TP-99295, 1999.
[26] L. Larsson, M. Visonneau and F. Stern, eds, in: *Gothenburg 2010 – A Workshop on Numerical Ship Hydrodynamics*, Gothenburg, December 2010.

[27] B.E. Launder and D.B. Spalding, The numerical computation of turbulent flows, *Computer Methods in Applied Mechanics and Engineering* **3**(2) (1974), 269–289.
[28] S. Lee, H. Kim, W. Kim and S.-H. Van, Wind tunnel tests on flow characteristics of the KRISO 3,600 TEU containership and 300K VLCC double-deck ship models, *Journal of Ship Research* **47**(1) (2003), 24–38.
[29] B.P. Leonard, Simple high-accuracy resolution program for convective modelling of discontinuities, *International Journal for Numerical Methods in Fluids* **8** (1988), 1291–1318.
[30] F.R. Menter, Two-equation eddy-viscosity turbulence models for engineering applications, *AIAA Journal* **32** (1994), 1598–1605.
[31] F. Menter and T. Esch, Elements of industrial heat transfer predictions, in: *16th Brasilian Congress of Mechanical Engineering*, November 2001.
[32] W.L. Oberkampf and C.J. Roy, *Verification and Validation in Scientific Computing*, Cambridge Univ. Press, Cambridge, UK, 2010.
[33] P.J. Roache, Code verification by the method of the manufactured solutions, *ASME Journal of Fluids Engineering* **114** (2002), 4–10.
[34] P.J. Roache, *Fundamentals of Verification and Validation*, Hermosa Publishers, Albuquerque, NM, 2009.
[35] P.J. Roache, From validation set points to a continuum domain of validation, in: *ASME Verification and Validation Symposium*, Las Vegas, NV, May 2012.
[36] P.J. Roache, *Verification and Validation in Computational Science and Engineering*, Hermosa Publishers, Albuquerque, NM, 1998.
[37] V. Romero, Comparison of a pragmatic and versatile "Real Space" model validation framework against several other frameworks, in: *International Workshop on V&V*, Computational Science Notre Dame University, October 2011.
[38] C.J. Roy and W.L. Oberkampf, A comprehensive framework for verification, validation, and uncertainty quantification in scientific computing, *Computer Methods in Applied Mechanics and Engineering* **200** (2011), 2131–2144.
[39] M.D. Salas, Some observations on grid convergence, *Computers and Fluids* **35** (2006), 688–692.
[40] P.R. Spalart and S.R. Allmaras, A one-equation turbulence model for aerodynamic flows, in: *AIAA 30th Aerospace Sciences Meeting*, Reno, NV, 1992.
[41] A. van der Ploeg, L. Eça and M. Hoekstra, Combining accuracy and efficiency with robustness in ship stern flow computations, in: *23rd Symposium on Naval Hydrodynamics*, Val re Reuil, France, 2000.
[42] G. Vaz, F. Jaouen and M. Hoekstra, Free-surface viscous flow computations. Validation of URANS code FRESCO, in: *Proceedings of OMAE2009*, Honolulu, HI, June 2009.
[43] M. Vinokur, On one-dimensional stretching functions for finite-difference calculations, *Journal of Computational Physics* **50** (1983), 215–234.